User's Manual

32

RZ/A2M Group

User's Manual: Hardware

Renesas Microprocessor
RZ Family / RZ/A Series

arm

Renesas Electronics
www.renesas.com

Rev.4.00 Nov 2020

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.
   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.
6. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.
(Note2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.4.0-1 November 2017)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin
   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Contents

1. Overview

1.1 Features of This LSI

This LSI is a single-chip microcontroller that includes an Arm Cortex®-A9 processor along with the integrated peripheral functions required to configure a system.

This LSI includes a 32-Kbyte L1 instruction cache, a 32-Kbyte L1 data cache, and a 128-Kbyte L2 cache. It also includes the following on-chip peripheral functions that are necessary for system configuration.

- 4-Mbyte large-capacity RAM (128 Kbytes are also used as data-retention RAM)
- HyperBus™*1 controller
- Octa memory*2 controller
- IEEE 1588 PTP compliant Ethernet MAC controller
- USB 2.0 host/function module
- Video display controller 6
- Image renderer engine
- 2D drawing engine
- MIPI CSI-2 interface
- SD/MMC host interface

The features of this LSI are listed in Table 1.1.

Note 1. HyperBus™, HyperFlash™ and HyperRAM™ are trademarks of Cypress Semiconductor Corporation.
Note 2. OctaFlash™ and OctaRAM™ are trademarks of Macronix International Co., Ltd.

Table 1.1 Features of RZ/A2M

Items	Specification
CPU	• Arm Cortex-A9 processor • Maximum operating frequency: 528 MHz • Instruction cache size: 32 Kbytes • Data cache size: 32 Kbytes (write-back algorithm) • TLB entries: 128 entries • Jazelle® architecture extension: Full implementation • Media processing engine with NEON™ technology
Boot modes	• Eight boot modes Boot mode 0: Booting from memory (bus width: 16 bits) connected to the CS0 space Note: This only applies to 324-pin products. Boot mode 1: Booting from a NAND flash memory with SD controller Boot mode 2: Booting from a NAND flash memory with MMC controller (data bus width: 8 bits) Boot mode 3: Booting from a serial flash memory (3.3 V) connected to the SPI multi I/O bus space Boot mode 4: Booting from an Octal-SPI flash memory connected to the SPI multi I/O bus space Boot mode 5: Booting from a HyperFlash connected to the SPI multi I/O bus space Boot mode 6: Booting from an OctaFlash connected to the OctaFlash space Boot mode 7: Booting from a HyperFlash connected to the HyperFlash space
Secondary cache	• Arm CoreLink™ Level 2 Cache Controller L2C-310 • Operating frequency: 132 MHz • Cache size: 128 Kbytes
Clock pulse generator	• Clock mode: Input clock can be selected from external input (EXTAL) or crystal resonator. • Input clock can be multiplied by 44 (max.) by the internal PLL circuit. • Peak values of EMI noise can be reduced by the on-chip SSCG circuit. • Five types of clocks generated: —CPU clock (Iφ): Maximum 528 MHz —Image processing clock (Gφ): Maximum 264 MHz —Internal bus clock (Bφ): Maximum 132 MHz —Peripheral clock 1 (P1φ): Maximum 66 MHz —Peripheral clock 0 (P0φ): Maximum 33 MHz
Interrupt controller	• Arm CoreLink™ Generic Interrupt Controller (GIC-400) • External interrupt pins (NMI, IRQ7 to IRQ0, and TINT31 to TINT0) • On-chip peripheral interrupts: Priority level set for each module • 32 priority levels available
Bus state controller [Only for 324-pin products]	• Address space divided into six areas (0 to 5), each a maximum of 64 Mbytes • The following features settable for each area independently —Bus size (8 or 16 bits): Available sizes depend on the area. —Number of access wait cycles (different wait cycles can be specified for read and write access cycles in some areas) —Idle wait cycle insertion (between the same area access cycles or different area access cycles) —Specifying the memory to be connected to each area enables direct connection to SRAM, SRAM with byte selection, SDRAM, and burst ROM (clocked synchronous or asynchronous). The address/data multiplexed I/O (MPX) interface is also available. • Outputs a chip select signal (CS0# to CS5#) according to the target area (CS assert or negate timing can be selected by software) • SDRAM refresh Auto refresh or self refresh mode selectable • SDRAM burst access
Direct memory access controller	• Two modules, sixteen channels; external requests are available for one channel. • Can be activated by on-chip peripheral modules. • A specific DMA transfer interval can be specified to adjust the bus occupancy. • Link mode (DMA transfer under descriptor control) supported • Transfer information can be automatically reloaded.

Table 1.1 Features of RZ/A2M

Items	Specification
Multi-function timer pulse unit 3	• 9 channels (16 bits × 8 channels, 32 bits × 1 channel) • Maximum 28 lines of pulse inputs/outputs and 3 lines of pulse inputs • 14 types of count clocks (P1φ/1, P1φ/2, P1φ/4, P1φ/8, P1φ/16, P1φ/32, P1φ/64, P1φ/256, P1φ/1024, MTCLKA, MTCLKB, MTCLKC, MTCLKD, MTIOC1A) selectable —14 types of count clocks selectable (channel 0) —12 types of count clocks selectable (channel 2) —11 types of count clocks selectable (channels 1, 3, 4, 6, 7, and 8) —10 types of count clocks selectable (channel 5) • Input capture function • 39 output compare and input capture registers • Counter clear operation (Simultaneous counter clearing by Compare match or Input capture is available.) • Simultaneous writing to multiple timer counters (TCNT) • Synchronous input/output of each register due to synchronous operation of the counter. • Buffered operation • Cascade-connected operation • 43 types of interrupt sources • Automatic transfer of register data • Pulse output modes Toggle, PWM, complementary PWM, and reset-synchronized PWM modes • Synchronization of multiple counters • Complementary PWM output mode —Non-overlapping waveforms output for 3-phase inverter control —Automatic dead time setting —0% to 100% PWM duty value specifiable —A/D converter start request delaying function —Interrupt skipping at crest or trough • Reset-synchronized PWM mode Three-phase PWM waveforms in positive and negative phases can be output with a required duty value. • Phase counting mode: 16-bit mode (channels 1 and 2) / 32-bit mode (channels 1 and 2) • Counter function of dead time compensation • Conversion start trigger of A/D converter can be generated. • Conversion start trigger of A/D converter can be skipped. • Digital filter functions for the input capture and external count clock pin.
Port output enable 3	• Control of the high-impedance state of the MTU3a waveform output pins • Activation with four input pins • Activation on detection of short-circuited outputs (detection of simultaneous PWM output to the active level) • Activation by register write • Additional programming of output control target pins is possible.
General PWM timer	• 32 bits × 8 channels • Counting up or down (sawtooth wave), counting up and down (triangular wave) selectable for all channels. • Independent selectable for each channel. • 2 input/output pins per channel • 2 output compare/input capture registers per channel • For the 2 output compare/input capture registers of each channel, 4 registers are provided as buffer registers and are capable of operating as comparison registers when buffering is not in use. • In output compare operation, buffer switching can be at peaks or troughs, enabling the generation of laterally asymmetrically PWM waveforms. • Registers for setting up frame intervals on each channel (with capability for generating interrupts on overflow or underflow) • Generation of dead times in PWM operation • Synchronous start / stop / clear of counters on arbitrary channels. • Starting, stopping, and clearing up/down counters in response to a maximum of eight events • Starting, stopping, and clearing up/down counters in response to input level comparison • Starting, stopping, and clearing up/down counters in response to a maximum of four external triggers • Output pin invalidation functions due to dead time error or detection of short circuit between output pins. • Generation of triggers for A/D converter conversion • Digital filter functions for the input capture and external trigger pins

Table 1.1 Features of RZ/A2M

Items	Specification
Port output enable for GPT	• Output prohibition control of the GPT waveform output pin • Activation with up to four input pins • Activation by dead time error detection or output short detection • Activation by register write
OS timer	• Three-channel 32-bit counters • Two operating modes: —Interval timer mode —Free-running comparison mode • DMA transfer request or interrupt request can be issued when a compare match occurs.
Watchdog timer	• On-chip one-channel watchdog timer • A counter overflow can reset the LSI. • CPU parity error can reset the LSI.
Realtime clock	• Internal clock, Two types of count mode, alarm function • Interrupts can be generated at intervals of 1/64 s by the 32.768-kHz on-chip crystal oscillator.
Serial communications interface with FIFO	• Five channels • Clock synchronous mode or asynchronous mode selectable • Simultaneous transmission and reception (full-duplex communication) supported • Dedicated baud rate generator • Separate 16-byte FIFO registers for transmission and reception • Modem control function (channels 0, 1, and 2 in asynchronous mode)
Serial communications interface	• Two channels • Clock synchronous mode, asynchronous mode, or smart card interface mode is selectable. • Simultaneous transmission and reception (full-duplex communication) supported • Dedicated baud rate generator • LSB first/MSB first selectable • Modem control function • Encoding and decoding of IrDA communications waveforms in accord with version 1.0 of the IrDA standard (on channel 0)
Renesas serial peripheral interface	• Three channels • SPI operation • Master mode and slave mode selectable • Programmable bit length, clock polarity, and clock phase can be selected. • Consecutive transfers • MSB first/LSB first selectable • Maximum transfer rate: 33.00 Mbps
SPI multi I/O bus controller	• One channel • Up to two serial flash memories with multiple I/O bus sizes (single/quad) can be connected. • Connectable with one Octal-SPI flash memory. • Connectable with one HyperFlash™ memory. • External address space read mode (built-in read cache) • SPI operating mode • Maximum frequency: 132 MHz (QSPI0_SPCLK)
HyperBus controller	• Two channels • Support HyperBus interface • One HyperFlash and HyperRAM can be connected per channel. • Maximum frequency: 132 MHz (HM_CK, HM_CK#)
Octa memory controller	• Two channels • Support Macronix Serial Multi I/O (MXSMIO®) Octa Peripheral Interface (OPI) interface. • One OctaFlash and OctaRAM can be connected per channel. • Support memory-map read/write feature with independent Flash/RAM address space. • OctaFlash supports STR (Single Transfer Rate) mode and DTR (Double Transfer Rate) mode. • SPI operation • Support Read-While-Write (RWW) function • Maximum frequency: 132 MHz (OM_SCLK)
I^2C bus interface	• Four channels • Master mode and slave mode supported • Support for 7-bit and 10-bit slave address formats • Support for multi-master operation • Timeout detection

Table 1.1 Features of RZ/A2M

Items	Specification
Serial sound interface	• Four-channel bidirectional serial transfer • Duplex communication (channels 0, 1, and 3) • Support of I²S, Monaural, and TDM audio formats • Support of master and slave functions • Generation of programmable word clock and bit clock • Multi-channel formats • Support of 8, 16, 18, 20, 22, 24, and 32-bit data formats • Support of 32-stage FIFO for transmission and reception • Support of LR-clock continue function in which the LR-clock signal is not stopped.
CANFD interface	• Two channels • ISO11898-1 (2003) compliant • CAN-FD ISO 11898-1 (2015) compliant • Message buffer: —Up to 64 × 2-channel receive message buffers: shared among all channels. —16 transmit message buffers per channel
Renesas SPDIF interface	• Support of IEC60958 standard (stereo and consumer use modes only) • Sampling frequencies of 32 kHz, 44.1 kHz, and 48 kHz • Audio word sizes of 16 to 24 bits per sample • Biphase mark encoding • Double buffered data • Parity encoded serial data
Ethernet MAC controller	• 176-pin products: 1 channel when the media independent interface (MII) is in use, 2 channels when the reduced media independent interface (RMII) is in use 256-, 272-, or 324-pin products: 2 channels whether the MII or RMII is in use • Conforms with the Ethernet / IEEE802.3 MAC (Media Access Control) layer standard • Supports transfer at 10 and 100 Mbps • Supports full-duplex and half-duplex mode • Supports Media Independent Interface (MII) compliant with the IEEE802.3u standard and Reduced Media Independent Interface (RMII) • Magic Packet™* detection and Wake-On-LAN (WOL) signal output • Built-in PTP controller (EPTPC) for Ethernet controller that uses Time Precision Time Protocol (PTP) defined by IEEE 1588-2008 (Version 2.0) to synchronize time between devices. • E-DMAC (Direct Memory Access Controller for Ethernet controller) function Note: * Magic Packet is a trademark of Advanced Micro Devices, Inc.
A/D converter	• 12-bit resolution • Eight input channels • Minimum conversion time: 1 µs per channel • A/D conversion request by the external trigger or timer trigger
NAND flash controller	• Direct-connected memory interface with NAND-type flash memory • ONFI 1.0 (mode 1 to 2) • ECC: 2, 4, 8, 16, 24, and 32 bits • Supports flash memory requiring Large block (2048+64) • Supports flash memory requiring 5-byte addresses (2 Gbits and more) • Interrupt request • Two types of DMA mode
USB 2.0 host/function module	• Two channels • USB 2.0-compliant host/function module • 480-Mbps, 12-Mbps, and 1.5-Mbps transfer rates provided (host mode) • 480-Mbps and 12-Mbps transfer rates provided (function mode) • On-chip 1-Kbyte RAM as communication buffers (host mode) • On-chip 8-Kbyte RAM as communication buffers (function mode)

Table 1.1 Features of RZ/A2M

Items	Specification
Video display controller 6	• Video input interface: BT601, BT656 format (NTSC/PAL) input: Input clock: 27 MHz/54 MHz Digital pin input (channel 0): YCbCr422, YCbCr444, RGB888, RGB666, RGB565 Digital pin input size: Maximum input video image size to be set*: 1920 pixels × 1080 lines (horizontal × vertical) Note: * Depends on the AC characteristics of the connected device. Examples of input video image size: XGA (1024 × 768) SVGA (800 × 600), WVGA (800 × 480), VGA (640 × 480), WQVGA (480 × 240), QVGA (320 × 240, 240 × 320) • Input video control Horizontal noise reduction (NR), brightness adjustment and contrast adjustment using matrix operation • Scaling control Vertical and horizontal scaling up or down of input video possible at a desired ratio (scaling up of graphics also possible) Scaling up ratio: 1 to 8; scaling down ratio: 1/8 to 1 Interpolation: Hold or linear selectable 2D IP conversion: 2D IP conversion through separately setting the initial phases for the top and bottom fields • Video recording Output pixel format: YCbCr444, YCbCr422, RGB888, RGB565 Output field rate: 1/1, 1/2, 1/4, 1/8 Rotation: Horizontal mirroring and 90/180/270 degree rotation for YCbCr422 and RGB565 Maximum video image size to be stored: ×1 size of input video image • Output video control Black stretch: Black area stretched according to Y signal state Enhancer capability: LTI (transient improvement) and sharpness (contour emphasis) for Y signal • Three graphics layers (one of them also for input video) Available input pixel formats 1 bit/pixel: CLUT1 4 bits/pixel: CLUT4 8 bits/pixel: CLUT8 16 bits/pixel: YCbCr422 (graphics layers 0), RGB565, ARGB1555, RGBA5551, ARGB4444 32 bits/pixel: ARGB8888, RGBA8888, RGB888, YCbCr444 (graphics layers 0) • Superimposition Alpha blending in a rectangular area: Input video, layer 1, and layer 2 blended according to the transparency percentage α (fade-in and fade-out function available) Chroma key function: Mixing based on transparency percentage α using the specified RGB and CLUT value Pixel-base alpha blending: Alpha blending for each pixel based on transparency percentage α • Panel output control Panel output correction: Brightness adjustment and contrast adjustment, gamma correction, panel dithering TCON: Various timing output for LCD panel driving provided by a total of seven vertical and horizontal panel driver signals Panel output pixel format: RGB888, RGB666, RGB565, serial RGB Output video image size: Maximum output video image size to be set*: 1999 pixels × 2035 lines (horizontal × vertical) Note: * Depends on the AC characteristics of the display panel. Examples of output video image size: XGA (1024 × 768) SVGA (800 × 600), WVGA (800 × 480), VGA (640 × 480), WQVGA (480 × 240), QVGA (320 × 240, 240 × 320)

Table 1.1 Features of RZ/A2M

Items	Specification
LVDS output interface	• Four pairs of differential output conforming to the TIA/EIA-644 standard (three pairs for data and one pair for the clock) • When the LVDS output interface is not used, the LVDS pins can be used as CMOS input/output. • The LVDS PLL generates clocks of various frequencies. • LVDS Power-down function
Image renderer (IMR-LS2)	• One channel • Refers to the video captured data as two-dimensional texture data and draws a shape by performing texture mapping for an arbitrary shape divided into triangular objects. • Display list system • Drawing functions Texture mapping, bilinear filtering, automatic coordinate generation (and relative coordinate input) • Instruction system Draw instruction: TRI for drawing a triangle Control instructions: TRAP, INT, NOP, SYNCM, SYNCW, WTL, and WTS • Drawing space Destination coordinates: 0 ≤ X ≤ 2,047, 0 ≤ Y ≤ 2,047 • Source coordinates: 0 ≤ u ≤ 1,439, 0 ≤ v ≤ 1,023
2D drawing engine (DRW)	• Support almost any object geometry, rather than being bound to only a few specific geometries such as lines, triangles, or circles. The edges of every object can be independently blurred or anti-aliased. • Color Formats Frame buffer formats 8-bit: a (8) 16-bit: RGB (565), aRGB (4444), aRGB (1555) 32-bit: aRGB (8888). Texture formats 1-bit: CLUT (1)/I (1) 2-bit: CLUT (2)/I (2) 4-bit: CLUT (4)/I (4) 8-bit: a (8), CLUT (8)/I (8), aCLUT (44) 16-bit: aRGB (4444), aRGB (1555), RGB (565) 24-bit: RGB (888) (run length encoded (RLE) unit) 32-bit: aRGB (8888). CLUT formats use a 256-entry color lookup table.
Sprite engine (SPEA)	• RLE Unit Supports Targa RLE data packet formats • Sprite Unit Two channels Up to 16 separate sprites processed by each channel
JPEG codec unit	• Compression and decompression method conforming to the JPEG baseline standard within the range described in this document. • Operational precision: Conforming to JPEG Part 2, ISO-IEC10918-2 • Pixel format: Compression: YCbCr422 Decompression: YCbCr444, YCbCr422, YCbCr411, YCbCr420 Output pixel format to the buffer: YCbCr422, ARGB8888, RGB565 • Four quantization tables provided • Four Huffman tables provided (two tables for AC coefficients and two tables for DC coefficients) • Markers supported: SOI, SOF0, SOS, DQT, DHT, DRI, RSTm, and EOI • Image data rate: Max. 132 Mbytes/s (at 66-MHz operation)

Table 1.1 Features of RZ/A2M

Items	Specification
Capture engine unit	• Examples of input video image size : 5 megapixels (2,560 × 1,920) 3 megapixels (2,048 × 1,536) 2 megapixels (1,632 × 1,224) UXGA (1,600 × 1,200) SXGA (1) (1,280 × 1,024) SXGA (2) (1,280 × 960) WXGA (1,280 × 768) XGA (1,024 × 768) SVGA (800 × 600) WVGA (800 × 480) VGA (640 × 480) WQVGA (480 × 240) QVGA (320 × 240, 240 × 320) Note: Depends on the AC characteristics of the connected device, frame rate of the connected device, and transfer speed to the destination RAM. • Input format: 8- or 16-bit binary data • Memory output format: YCbCr422, YCbCr420 Note: The captured data cannot be displayed via the video display controller 6 because the Y data and CbCr data are split when written to memory.
MIPI CSI-2 interface [Only for 256-, 272-, or 324-pin products]	• Up to 1.0-Gbps transfer rate of MIPI CSI-2 • ECC 1-bit error correction and 2-bit or more error detection of a packet header • CRC error detection of a payload data part • Generation of VD (vertical sync), HD (horizontal sync), and FLD (field) signals • Lane swapping
Video input module	• Input formats MIPI CSI2 interface: YCbCr422, RGB888, RAW8 Up to 2048 × 2048 pixels capture area • Memory output data formats YCbCr422, RGB565, ARGB1555, RGB888, ARGB8888
SD/MMC host interface	• 176-pin products: 1 channel 256-, 272-, or 324-pin products: 2 channels • SD memory I/O card interface (1-/4-bit SD bus) • SD, SDHC, and SDXC SD memory card access supported • Default, high-speed, UHS-I/SDR50, DDR50, and SDR104 transfer modes supported • Error check function: CRC7 (command), CRC16 (data) • Interrupt request: 2 • Card detection function, write protect supported • MMC interface (1-/4-/8-bit MMC bus) Note: Channel 1 only supports a 1- or 4-bit MMC bus. • e-MMC device access supported • High-speed, HS200 transfer modes supported
On-chip RAM	• 4-Mbyte large capacity memory for video display/recording and work (128 Kbytes are used for data retention) • 128-Kbyte memory for data retention (16 Kbytes × 2, 32 Kbytes × 1, 64 Kbytes × 1)
General I/O ports	• 176-pin products: 47 I/Os, 8 inputs with open-drain outputs, 14 inputs (input only), and 1 output (output only) • 256-pin products: 92 I/Os, 8 inputs with open-drain outputs, 14 inputs (input only), and 1 output (output only) • 272-pin products: 92 I/Os, 8 inputs with open-drain outputs, 14 inputs (input only), and 1 output (output only) • 324-pin products: 128 I/Os, 8 inputs with open-drain outputs, 14 inputs (input only), and 1 output (output only) • Input or output can be selected for each bit.
Power-down modes	• Four power-down modes provided to reduce the power consumption in this LSI —Sleep mode —Software standby mode —Deep standby mode —Module standby mode
Debugger interface	• Arm CoreSight™ architecture • JTAG-standard pin assignment

Table 1.1 Features of RZ/A2M

Items	Specification
Trusted Secure IP [option]	• Security algorithm Common key encryption: AES (compliant with NIST FIPS PUB 197), 3DES, ARC4 Non-common key encryption: RSA • Other features TRNG (true-random number generator) Hash value generation: SHA1, SHA224, SHA256, GHASH Support of unique ID
OTP	• A nonvolatile memory that can be written only once • Security setting, authentication setting and the OTP boot setting are possible.
Dynamic reconfigurable processor (DRP) [option]	• Dynamically reconfigurable intellectual property module • 6 tiles • 32 I/O ports
Power supply voltage	• Vcc: 1.14 to 1.26 V, PVcc: 3.0 to 3.6 V, PVcc_HO: 1.7 to 1.9 V • PVcc_SPI/PVcc_SD0/PVcc_SD1: 1.7 to 1.9 V/3.0 to 3.6 V
Temperature range	-40 to +85°C
Quality level	Industrial usage, etc.
Package	• PLBG0176GA-B 176-pin BGA, 13-mm square, 0.8-mm pitch JEITA Package Code: P-LFBGA176-13×13-0.80 RENESAS Code: PLBG0176GA-B • PLBG0256KA-A 256-pin BGA, 11-mm square, 0.5-mm pitch JEITA Package Code: P-LFBGA256-11×11-0.50 RENESAS Code: PLBG0256KA-A • PRBG0272GA-A 272-pin BGA, 17-mm square, 0.80-mm pitch JEITA Package Code: P-LFBGA272-17×17-0.80 RENESAS Code: PRBG0272GA-A • PRBG0324GA-A 324-pin BGA, 19-mm square, 0.8-mm pitch JEITA Package Code: P-FBGA324-19x19-0.80 RENESAS Code: PRBG0324GA-A

1.2 Product Lineup

Table 1.2 Product Lineup

Group	Part Number	DRP Function	Trusted Secure IP	Package
RZ/A2M	R7S921040VCBG	No DRP	No Trusted Secure IP	PLBG0176GA-B
	R7S921041VCBG			PLBG0256KA-A
	R7S921042VCBG			PRBG0272GA-A
	R7S921043VCBG			PRBG0324GA-A
	R7S921045VCBG		Available	PLBG0176GA-B
	R7S921046VCBG			PLBG0256KA-A
	R7S921047VCBG			PRBG0272GA-A
	R7S921048VCBG			PRBG0324GA-A
	R7S921051VCBG	Available	No Trusted Secure IP	PLBG0256KA-A
	R7S921052VCBG			PRBG0272GA-A
	R7S921053VCBG			PRBG0324GA-A
	R7S921056VCBG		Available	PLBG0256KA-A
	R7S921057VCBG			PRBG0272GA-A
	R7S921058VCBG			PRBG0324GA-A

Note: Usable pins of the modules listed below depend on the type of package in which the given product comes. For details, see section 1.6, List of Pins.
- Interrupt controller
- Direct memory access controller
- Multi-function timer pulse unit 3
- General PWM timer
- HyperBus controller
- Serial sound interface
- CANFD interface
- Renesas SPDIF interface
- NAND flash controller
- Video display controller 6

1.3 Block Diagram

See section 5, LSI Internal Bus.

1.4 Pin Assignment

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
A	Vcc	QSPI1_IO3	QSPI1_SPCLK	RPC_WP#	QSPI0_IO3	PVcc_SPI	Vss	PVcc	PF_4	PE_6	PL_2	PE_5	P8_4	P8_6	PE_4	P9_1	PVcc	Vss	PE_1	PA_4	CKIO	Vss	A
B	PK_1	Vcc	QSPI1_IO1	QSPI1_IO0	RPC_RESET#	QSPI0_IO1	QSPI0_SPCLK	PF_5	P6_3	PH_0	PL_3	PL_1	P8_3	PF_2	P8_7	PE_3	PA_0	PA_3	PA_5	PA_6	Vss	PVcc	B
C	PH_2	P8_2	Vcc	QSPI1_SSL	RPC_INT#	QSPI0_SSL	QSPI0_IO0	P6_1	P6_2	PH_1	PL_4	PL_0	P8_5	PF_1	P9_0	PE_2	PA_2	PG_0	PB_0	Vss	PD_7	PD_3	C
D	PVcc	BSCANP	P8_1	Vcc	QSPI1_IO2	Vss	QSPI0_IO2	PK_0	PF_6	PE_0	PF_3	PVcc	Vss	PF_0	P8_0	PA_1	PA_7	PVcc	Vss	PD_6	PD_4	PD_1	D
E	Vss	PH_3	PK_3	PK_2															PD_5	PD_2	TCK/SWDCLK	JP0_0	E
F	PVcc_HO	HM_CS0#/OM_CS0#	HM_CK/OM_SCLK	PF_7															PD_0	TRST#	TMS/SWDIO	PB_1	F
G	HM_DQ1/OM_SIO1	HM_RWDS/OM_DQS	HM_CS1#/OM_CS1#	HM_CK#															JP0_1	PB_2	PB_3	P9_2	G
H	HM_DQ4/OM_SIO4	HM_DQ2/OM_SIO2	HM_DQ3/OM_SIO3	HM_DQ0/OM_SIO0															PB_4	P9_3	PB_5	P9_5	H
J	HM_RESET#/OM_RESET#	HM_DQ6/OM_SIO6	HM_DQ7/OM_SIO7	HM_DQ5/OM_SIO5					Vcc	Vss	Vss	Vss	Vss	Vcc					P9_4	P7_7	P7_6	P9_6	J
K	Vss	PJ_6	PH_4	PJ_0					Vcc	Vss	Vss	Vss	Vss	Vcc					P9_7	PG_1	P7_5	PG_2	K
L	PVcc	P0_1	P0_0	PJ_7					Vcc	Vss	Vss	Vss	Vss	Vcc					P7_1	P7_4	P7_3	P7_2	L
M	P0_2	P0_5	P0_4	P0_3					Vcc	Vss	Vss	Vss	Vss	Vcc					P6_6	P6_0	P7_0	PVcc	M
N	PJ_3	PJ_1	P0_6	PJ_2					Vcc	Vss	Vss	Vss	Vss	Vcc					P6_5	PG_3	P6_4	Vss	N
P	PH_6	PH_5	PK_5	PVcc					Vcc	Vss	Vss	Vss	Vss	Vcc					SD0_DAT7	SD0_RST#	P6_7	PVcc_SD0	P
R	PVcc	PJ_4	PJ_5	Vss															SD0_DAT2	SD0_DAT5	SD0_DAT4	SD0_DAT6	R
T	AUDIO_X1	AUDIO_X2	P3_5	P3_2															SD0_DAT0	SD0_DAT1	SD0_DAT3	Vss	T
U	Vss	PK_4	P3_1	MIPI AVcc18															SD1_DAT0	SD1_DAT2	SD0_CMD	SD0_CLK	U
V	CSI_CLKP	CSI_CLKN	PG_4	Vss															P5_4	SD1_DAT1	SD1_DAT3	PVcc_SD1	V
W	CSI_DATA0P	CSI_DATA0N	Vss	PG_6	P1_0	P1_2	P2_0	PC_2	P4_3	LVDS APVcc	Vss	LVDS PLLVcc	USB DPVcc0	USBVss	Vss	PVcc	Vss	PLLVcc	P5_2	P5_6	SD1_CMD	Vss	W
Y	CSI_DATA1P	CSI_DATA1N	Vss	P3_3	P1_4	PC_0	P2_2	P4_2	P4_6	NMI	Vss	USBVss	USBVss	USBVss	USBVss	USB DPVcc1	PC_7	PC_6	P5_0	P5_1	P5_7	SD1_CLK	Y
AA	MIPI AVcc18	Vss	P1_1	P3_4	P3_0	PC_1	P4_0	P4_4	P4_7	USB_X2	DP0	USB APVcc0	RREF0	USBVss	DP1	PVcc	PC_5	XTAL	PC_4	RTC_X2	P5_3	P5_5	AA
AB	Vss	PG_5	PG_7	P1_3	P2_1	P2_3	P4_1	P4_5	RES#	USB_X1	DM0	USB APVcc1	RREF1	USBVss	DM1	PVcc	PC_3	EXTAL	Vss	RTC_X1	AVcc	AVss	AB
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	

Figure 1.1 Pin Assignment of the 324-Pin BGA (Top Perspective View)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
A	Vcc	QSPI1_IO3	QSPI1_IO1	QSPI1_SPCLK	QSPI0_SSL	QSPI0_IO0	Vss	PVcc	P6_3	PE_6	PL_4	PL_2	PL_1	PF_1	PE_2	PA_3	PA_6	Vss	CKIO	Vss	A
B	Vss	Vcc	QSPI1_IO2	RPC_INT#	RPC_RESET#	QSPI0_IO3	QSPI0_SPCLK	P6_1	PF_5	PH_1	PF_3	PL_0	PE_5	P8_0	PE_3	PA_2	PA_4	PB_0	Vss	PD_6	B
C	PH_2	PK_1	Vcc	QSPI1_SSL	RPC_WP#	QSPI0_IO1	QSPI0_IO2	PF_6	PE_0	PF_4	PL_3	PF_2	PE_4	PA_0	PE_1	PA_5	PG_0	Vss	PD_7	PD_5	C
D	PH_3	PK_3	PK_2	Vcc	QSPI1_IO0	Vss	PVcc_SPI	PK_0	P6_2	PH_0	PVcc	Vss	PF_0	PA_1	PA_7	PVcc	Vss	PD_4	PD_3	PD_1	D
E	Vss	BSCANP	PF_7	PVcc													PVcc	PD_2	TCK/SWDCLK	JP0_0	E
F	HM_RWDS/OM_DQS	HM_CK#	HM_CK/OM_SCLK	Vss													TMS/SWDIO	PD_0	TRST#	JP0_1	F
G	HM_DQ2/OM_SIO2	HM_DQ1/OM_SIO1	HM_CS0#/OM_CS0#	PVcc_HO													Vss	PB_1	PB_2	PB_3	G
H	HM_DQ3/OM_SIO3	HM_DQ6/OM_SIO6	HM_DQ4/OM_SIO4	HM_CS1#/OM_CS1#													PVcc	PB_5	PB_4	P7_7	H
J	HM_RESET#/OM_RESET#	HM_DQ7/OM_SIO7	HM_DQ5/OM_SIO5	HM_DQ0/OM_SIO0					Vcc	Vss	Vss	Vcc					P7_6	PG_2	PG_1	P7_2	J
K	PJ_7	PJ_6	PH_4	PJ_0					Vcc	Vss	Vss	Vcc					PVcc	PG_3	P6_7	PVcc	K
L	PJ_1	PJ_3	PH_5	PVcc					Vcc	Vss	Vss	Vcc					Vss	P6_0	P6_5	Vss	L
M	PH_6	PJ_2	PK_5	Vss					Vcc	Vss	Vss	Vcc					PVcc_SD0	P6_4	P6_6	SD0_RST#	M
N	PVcc	PJ_5	PJ_4	PVcc													SD0_DAT0	SD0_DAT5	SD0_DAT7	SD0_DAT6	N
P	AUDIO_X1	AUDIO_X2	P3_5	PK_4													PVcc_SD1	SD0_DAT2	SD0_DAT3	SD0_DAT4	P
R	Vss	P3_1	PG_4	MIPI AVcc18													Vss	SD0_CMD	SD0_DAT1	SD0_CLK	R
T	CSI_CLKP	CSI_CLKN	P3_2	Vss													AVss	SD1_DAT0	SD1_CLK	Vss	T
U	CSI_DATA0P	CSI_DATA0N	Vss	PG_6	PC_0	P3_3	P4_6	LVDS PLLVcc	LVDS APVcc	USB DPVcc0	RREF0	RREF1	Vss	PLLVcc	PVcc	PC_6	AVcc	SD1_CMD	SD1_DAT1	SD1_DAT2	U
V	CSI_DATA1P	CSI_DATA1N	Vss	P3_4	PC_2	P4_2	P4_7	Vss	Vss	USBVss	USBVss	USBVss	Vss	PC_3	PC_5	PC_7	P5_0	P5_6	P5_7	SD1_DAT3	V
W	MIPI AVcc18	Vss	PG_5	P3_0	P4_1	P4_3	NMI	USB_X2	USBVss	DP0	USB APVcc0	USB APVcc1	DM1	USB DPVcc1	PC_4	XTAL	RTC_X2	P5_4	P5_3	P5_5	W
Y	Vss	PG_7	PC_1	P4_0	P4_4	P4_5	RES#	USB_X1	USBVss	DM0	USBVss	USBVss	DP1	USBVss	EXTAL	Vss	RTC_X1	P5_2	P5_1	Vss	Y
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	

Figure 1.2 Pin Assignment of the 272-Pin BGA (Top Perspective View)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	
A	Vcc	QSPI1_SSL	QSPI1_IO3	QSPI1_SPCLK	RPC_RESET#	PVcc_SPI	Vss	QSPI0_SPCLK	P6_1	P6_3	PE_6	PL_4	PL_0	PE_5	PE_4	PE_3	PA_2	PE_1	PA_6	CKIO	Vss	A
B	PK_1	Vcc	QSPI1_IO0	QSPI1_IO1	RPC_INT#	QSPI0_SSL	QSPI0_IO3	QSPI0_IO1	PF_5	PF_4	PH_1	PL_1	PL_2	P8_0	PF_1	PA_0	PA_7	PA_4	PB_0	Vss	PD_3	B
C	PK_3	PH_2																		PD_7	PD_1	C
D	HM_CK/OM_SCLK	BSCANP		Vcc	QSPI1_IO2	RPC_WP#	QSPI0_IO2	PK_0	P6_2	PH_0	PF_3	PF_2	PF_0	PE_2	PA_1	PG_0	PA_5	Vss		PD_5	JP0_0	D
E	HM_CK#	HM_CS0#/OM_CS0#		PK_2	Vcc	Vss	PVcc	QSPI0_IO0	PF_6	PE_0	PL_3	Vcc	Vss	PVcc	PA_3	Vcc	Vss	TCK/SWDCLK		PD_0	TRST#	E
F	HM_DQ1/OM_SIO1	HM_RWDS/OM_DQS		PF_7	Vss												PD_6	PD_2		JP0_1	PB_1	F
G	PVcc_HO	HM_DQ2/OM_SIO2		HM_DQ4/OM_SIO4	PVcc												PD_4	TMS/SWDIO		PB_3	PB_5	G
H	Vss	HM_DQ6/OM_SIO6		HM_CS1#/OM_CS1#	PH_3												PB_2	PB_4		P7_7	PG_2	H
J	HM_RESET#/OM_RESET#	PH_4		HM_DQ3/OM_SIO3	HM_DQ0/OM_SIO0												PG_1	P7_6		P7_2	P6_0	J
K	PJ_6	PJ_1		HM_DQ7/OM_SIO7	HM_DQ5/OM_SIO5												PVcc	PG_3		P6_5	P6_4	K
L	PH_5	PJ_3		PJ_0	Vcc												Vss	P6_7		P6_6	SD0_DAT6	L
M	PH_6	PJ_7		PJ_2	Vss												Vcc	SD0_RST#		SD0_DAT7	PVcc_SD0	M
N	P3_5	PJ_5		PJ_4	PK_5												SD0_DAT0	SD0_DAT5		SD0_DAT4	Vss	N
P	AUDIO_X1	AUDIO_X2		PK_4	PVcc												SD0_CLK	SD0_DAT2		SD0_DAT3	SD0_DAT1	P
R	PG_4	P3_1		P3_2	Vss												SD1_DAT2	SD1_DAT0		SD1_DAT3	SD0_CMD	R
T	Vss	Vss		MIPIA Vcc18	Vcc												SD1_CLK	SD1_CMD		SD1_DAT1	PVcc_SD1	T
U	CSI_DATA0P	CSI_DATA0N		Vss	PVcc	Vss	LVDS APVcc	P4_3	P4_7	USBVss	RREF0	RREF1	USBVss	Vss	PLLVcc	PVcc	Vss	P5_4		P5_5	Vss	U
V	CSI_CLKP	CSI_CLKN		PG_6	P3_3	PC_2	Vss	P4_2	P4_6	Vss	USB APVcc0	USB APVcc1	USBVss	PC_4	PC_6	PC_3	PVcc	P5_0		P5_6	P5_7	V
W	CSI_DATA1P	CSI_DATA1N																		P5_2	P5_3	W
Y	Vss	Vss	PG_7	PC_0	PC_1	P4_0	P4_4	LVDS PLLVcc	NMI	USB_X2	USB DPVcc0	DP0	USBVss	DP1	USB DPVcc1	Vss	XTAL	RTC_X2	AVcc	PVcc	P5_1	Y
AA	Vss	PG_5	P3_4	P3_0	Vss	P4_1	P4_5	Vss	RES#	USB_X1	USBVss	DM0	USBVss	DM1	USBVss	PC_5	PC_7	EXTAL	RTC_X1	AVss	PVcc	AA
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	

Figure 1.3 Pin Assignment of the 256-Pin BGA (Top Perspective View)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
A	Vss	PK_3	QSPI1_SPCLK	RPC_WP#	PVcc_SPI	Vss	QSPI0_IO3	P6_1	P6_2	PH_1	PE_6	PL_2	PE_3	CKIO	Vss	A
B	BSCANP	PH_3	QSPI1_SSL	QSPI1_IO1	RPC_RESET#	QSPI0_IO2	QSPI0_IO1	PK_0	P6_3	PL_4	PL_0	PE_5	PE_1	Vss	PD_7	B
C	HM_CK/OM_SCLK	PK_1	QSPI1_IO3	QSPI1_IO2	RPC_INT#	QSPI0_SSL	QSPI0_SPCLK	PE_0	PH_0	PL_1	PE_4	PG_0	Vss	PD_5	PD_3	C
D	HM_CK#	HM_CS0#/OM_CS0#	PK_2	Vss	Vcc	QSPI1_IO0	QSPI0_IO0	PVcc	PL_3	PE_2	PVcc	Vss	PD_6	PD_4	PD_1	D
E	Vss	HM_DQ0/OM_SIO0	HM_RWDS/OM_DQS	PVcc								PD_2	PD_0	TCK/SWDCLK	JP0_0	E
F	PVcc_HO	HM_DQ2/OM_SIO2	HM_DQ1/OM_SIO1	HM_CS1#/OM_CS1#								TMS/SWDIO	JP0_1	TRST#	P6_0	F
G	HM_DQ4/OM_SIO4	HM_DQ7/OM_SIO7	HM_DQ5/OM_SIO5	HM_DQ3/OM_SIO3								Vcc	PG_3	PG_2	PVcc	G
H	HM_DQ6/OM_SIO6	HM_RESET#/OM_RESET#	PH_4	Vcc								PG_1	SD0_DAT5	SD0_RST#	Vss	H
J	PJ_0	PJ_1	PJ_2	Vss								SD0_DAT7	SD0_DAT2	SD0_DAT4	PVcc_SD0	J
K	PJ_5	PJ_3	PJ_4	PVcc								SD0_CLK	SD0_DAT3	SD0_DAT0	SD0_DAT6	K
L	PK_4	P3_5	Vss	Vcc								PVcc	Vss	SD0_DAT1	SD0_CMD	L
M	AUDIO_X1	AUDIO_X2	P3_2	Vss	P3_3	LVDS APVcc	LVDS PLLVcc	NMI	USBVss	PVcc	Vcc	P5_0	P5_4	P5_7	P5_5	M
N	P3_1	PG_4	Vss	PG_7	P4_2	P4_3	P4_6	RES#	RREF0	Vss	PLLVcc	Vcc	P5_2	P5_3	P5_6	N
P	PVcc	Vss	PG_6	P4_0	P4_4	P4_7	USB_X2	USB DPVcc0	USB APVcc0	USBVss	XTAL	RTC_X2	AVcc	Vcc	P5_1	P
R	Vss	PG_5	P3_4	P4_1	P4_5	Vss	USB_X1	DP0	DM0	USBVss	EXTAL	Vss	RTC_X1	AVss	Vcc	R
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	

Figure 1.4 Pin Assignment of the 176-Pin BGA (Top Perspective View)

1.5 Pin Functions

Table 1.3 Pin Function

Classification	Symbol	I/O	Name	Function
Power supply	Vcc	I	Power supply	Power supply pins. All the Vcc pins must be connected to the system power supply. This LSI does not operate correctly if there is a pin left open.
	Vss	I	Ground	Ground pins. All the Vss pins must be connected to the system power supply (0 V). This LSI does not operate correctly if there is a pin left open.
	PVcc	I	Power supply for I/O circuits	Power supply for I/O pins. All the PVcc pins must be connected to the system power supply. This LSI does not operate correctly if there is a pin left open.
	PVcc_SPI, PVcc_HO, PVcc_SD0, PVcc_SD1	I	Power supply for I/O circuits	Power supply for I/O pins. All the PVcc_SPI, PVcc_HO, PVcc_SD0, PVcc_SD1 pins must be connected to the system power supply. This LSI does not operate correctly if there is a pin left open.
	PLLVcc	I	Power supply for PLL	Power supply for the on-chip PLL oscillator.
Clock	EXTAL	I	External clock	Connected to a crystal resonator. An external clock signal may also be input to the EXTAL pin.
	XTAL	O	Crystal	Connected to a crystal resonator.
	CKIO	O	System clock output	Supplies the system clock to external devices.
	AUDIO_CLK	I	External clock for audio	Input pin of external clock for audio. A clock input to the divider is selected from an oscillation clock input on this pin or pins AUDIO_X1 and AUDIO_X2.
	AUDIO_X1	I	Crystal resonator/ external clock for audio	Pins connected to a crystal resonator for audio. An external clock can be input on pin AUDIO_X1. A clock input to the divider is selected from an oscillation clock input on these pins or the AUDIO_CLK pin.
	AUDIO_X2	O		
	AUDIO_XOUT	O	AUDIO_X1 clock output	Output for the on-chip crystal oscillator on AUDIO_X1 or the external clock signal.
Operating mode control	MD_BOOT2, MD_BOOT1, MD_BOOT0	I	Mode set	Sets the operating mode. Do not change the signal levels on these pins while the RES# pin is asserted or until the mode is fixed, after the negation.
	MD_CLK	I	Clock mode set	Sets the clock operating mode. Do not change the signal levels on this pin while the RES# pin is asserted or until the mode is fixed, after the negation.
	MD_CLKS	I	SSCG clock mode set	Switches the SSCG circuit on or off. Do not change the signal levels on this pin while the RES# pin is asserted or until the mode is fixed, after the negation.
	BSCANP	I	Boundary scan set	Boundary scan setting pin. This pin is set to the high level for a boundary scan and to the low level for normal operation.
System control	RES#	I	Power-on reset	This LSI enters the power-on reset state when this signal goes low.
	WDTOVF# / PERROUT#	O	Watchdog timer overflow / CPU Parity error	Outputs an overflow signal from the watchdog timer. Outputs the CPU parity error.

Table 1.3 Pin Function

Classification	Symbol	I/O	Name	Function
Interrupts	NMI	I	Non-maskable interrupt	Non-maskable interrupt request pin. It is handled as an FIQ exception. Fix it high when not in use.
	IRQ7 to IRQ0	I	Interrupt requests 7 to 0	Maskable interrupt request pins. Level-input or edge-input detection can be selected. When the edge-input detection is selected, the rising edge, falling edge, or both edges can also be selected.
Address bus	A25 to A0	O	Address bus	Outputs addresses
Data bus	D15 to D0	I/O	Data bus	Bidirectional data bus
Bus control	CS5# to CS0#	O	Chip select 5 to 0	Chip-select signals for external memory or devices.
	RD#	O	Read	Indicates that data is read from an external device.
	RD/WR#	O	Read/write	Read/write signal
	BS#	O	Bus start	Bus-cycle start signal
	AH#	O	Address hold	Address hold timing signal for the device that uses the address/data-multiplexed bus.
	WAIT#	I	Wait	Inserts a wait cycle into the bus cycles during access to the external space.
	WE0#	O	Byte select	Indicates a write access to bits 7 to 0 of data of external memory or device.
	WE1#	O	Byte select	Indicates a write access to bits 15 to 8 of data of external memory or device.
	DQML	O	Byte select	Selects bits D7 to D0 when SDRAM is connected.
	DQMU	O	Byte select	Selects bits D15 to D8 when SDRAM is connected.
	RAS#	O	RAS	Connected to the RAS# pin when SDRAM is connected.
	CAS#	O	CAS	Connected to the CAS# pin when SDRAM is connected.
	CKE	O	CK enable	Connected to the CKE pin when SDRAM is connected.
Direct memory access controller	DREQ0	I	DMA-transfer request	Input pin to receive external requests for DMA transfer.
	DACK0	O	DMA-transfer request accept	Output pin for signals indicating acceptance of external requests from external devices.
	TEND0	O	DMA-transfer end output	Output pin for DMA transfer end.

Table 1.3 Pin Function

Classification	Symbol	I/O	Name	Function
Multi-function timer pulse unit 3	MTCLKA, MTCLKB, MTCLKC, MTCLKD	I	Timer clock input	Input pins for external clock signals or for phase counting mode clock signals.
	MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D	I/O	Input capture/output compare (channel 0)	TGRA_0 to TGRD_0 input capture input/output compare output/PWM output pins.
	MTIOC1A, MTIOC1B	I/O	Input capture/output compare (channel 1)	TGRA_1 and TGRB_1 input capture input/output compare output/PWM output pins.
	MTIOC2A, MTIOC2B	I/O	Input capture/output compare (channel 2)	TGRA_2 and TGRB_2 input capture input/output compare output/PWM output pins.
	MTIOC3A, MTIOC3B, MTIOC3C, MTIOC3D	I/O	Input capture/output compare (channel 3)	TGRA_3 to TGRD_3 input capture input/output compare output/PWM output pins.
	MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D	I/O	Input capture/output compare (channel 4)	TGRA_4 to TGRD_4 input capture input/output compare output/PWM output pins.
	MTIC5U, MTIC5V, MTIC5W	I/O	Input capture/output compare (channel 5)	TGRU_5, TGRV_5, and TGRW_5 input capture input/dead time compensation input pins.
	MTIOC6A, MTIOC6B, MTIOC6C, MTIOC6D	I/O	Input capture/output compare (channel 6)	TGRA_6 and TGRB_6 input capture input/output compare output/PWM output pins.
	MTIOC7A, MTIOC7B, MTIOC7C, MTIOC7D	I/O	Input capture/output compare (channel 7)	TGRA_7 to TGRD_7 input capture input/output compare output/PWM output pins.
	MTIOC8A, MTIOC8B, MTIOC8C, MTIOC8D	I/O	Input capture/output compare (channel 8)	TGRA_8 to TGRD_8 input capture input/output compare output pins.
Port output enable 3	POE0#, POE4#, POE8#, POE10#	I	High impedance request	Input pins for request signals to place the MTU3a waveform output pins in the high impedance state.
General purpose PWM timer	GTETRGA to GTETRGD	I	External trigger	External trigger input pins.
	GTIOC0A to GTIOC7A, GTIOC0B to GTIOC7B	I/O	Timer I/O	Input capture, output compare, or PWM output pins.
Realtime clock	RTC_X1	I	Crystal resonator for realtime clock/external clock	Connected to 32.768-kHz crystal resonator The RTC_X1 pin can also be used to input an external clock.
	RTC_X2	O		
	EXTAL	I	Crystal resonator for internal clock/external clock	Connected to a crystal resonator.
	XTAL	O		
Serial communication interface with FIFO	TxD4 to TxD0	O	Transmit data	Data output pins.
	RxD4 to RxD0	I	Receive data	Data input pins.
	SCK4 to SCK0	I/O	Serial clock	Clock input/output pins.
	RTS2#, RTS1#, RTS0#	O	Transmit request	Modem control pins.
	CTS2#, CTS1#, CTS0#	I/O	Transmit enable	Modem control pins.
Serial communication interface	SCI_SCK1, SCI_SCK0	I/O	Serial clock	Clock input/output pins.
	SCI_TXD1, SCI_TXD0	O	Transmit data	Data output pins.
	SCI_RXD1, SCI_RXD0	I	Receive data	Data input pins.
	SCI_CTS1# / RTS1#, SCI_CTS0# / RTS0#	I/O	Transmit and receive start control	I/O pins for controlling the start of transmission and reception.
I^2C bus interface	RIIC3SCL to RIIC0SCL	I/O	Serial clock pin	Serial clock I/O pins.
	RIIC3SDA to RIIC0SDA	I/O	Serial data pin	Serial data I/O pins.
Serial sound interface	SSITxD3, SSITxD1, SSITxD0	O	Data output	Serial data output pin.
	SSIRxD3, SSIRxD1, SSIRxD0	I	Data input	Serial data input pin.
	SSIDATA2	I/O	Data I/O	Serial data I/O pins.
	SSIBCK3 to SSISBCK0	I/O	bit clock I/O	I/O pins for bit clocks.
	SSILRCK3 to SSILRCK0	I/O	LR clock I/O	I/O pins for LR clock / frame sync.

Table 1.3 Pin Function

Classification	Symbol	I/O	Name	Function
CANFD interface	CAN_CLK	I	Clock source for CAN communication	Clock source for CAN communication.
	CAN1TX, CAN0TX	O	CAN bus transmit data	Output pins for transmit data on the CAN bus.
	CAN1RX, CAN0RX	I	CAN bus receive data	Input pins for receive data on the CAN bus.
	CAN0RX_DATARATE_EN, CAN1RX_DATARATE_EN	O	CAN Bus receive data phase	This pin indicates the receive data phase of the CAN bus. This pin is set to "1" in the data (high-speed bit rate) area.
	CAN0TX_DATARATE_EN, CAN1TX_DATARATE_EN	O	CAN bus transmit data phase	This pin indicates the transmit data phase of the CAN bus. This pin is set to "1" in the data (high-speed bit rate) area.
Renesas SPDIF interface	SPDIF_OUT	O	Data output	Transmit data output pin.
	SPDIF_IN	I	Data input	Receive data input pin.
Renesas serial peripheral interface	MOSI2 to MOSI0	I/O	Data	Data I/O pins.
	MISO2 to MISO0	I/O	Data	Data I/O pins.
	RSPCK2 to RSPCK0	I/O	Clock	Clock I/O pins.
	SSL20, SSL10, SSL00	I/O	Slave select	Slave select I/O pins.
SPI multi I/O bus controller	QSPI0_SPCLK, QSPI1_SPCLK	O	Clock	Clock output pins.
	QSPI0_SSL	O	Slave select	Slave select output pins.
	QSPI1_SSL	I/O	Slave select or Data strobe	When serial flash memory is connected: This pin serves as a slave select output pin. When Octal-SPI flash or HyperFlash memory is connected: This pin serves as a data strobe input pin.
	QSPI0_IO3 to QSPI0_IO0, QSPI1_IO3 to QSPI1_IO0	I/O	Data	Data I/O pins.
	RPC_RESET#	O	Reset	Reset output pin.
	RPC_WP#	O	Write protect	Write protect output pin.
	RPC_INT#	I	Interrupt	Interrupt input pin.
HyperBus controller	HM_CK	O	Clock	Differential clock output pin.
	HM_CK#	O	Clock	Differential clock output pin.
	HM_CS0#	O	Chip select 0	Chip select signal for the external memory.
	HM_CS1#	O	Chip select 1	Chip select signal for the external memory.
	HM_RWDS	I/O	Read Write Data Strobe	Read: data strobe pin. Write: data Mask pin.
	HM_DQ7 to HM_DQ0	I/O	Data	Data I/O pin.
	HM_RESET#	O	Reset output	Reset signal for the external memory.
	HM_RSTO#	I	Reset input	Reset signal from the external memory.
	HM_INT#	I	Interrupt input	Interrupt signal from the external memory.

Table 1.3 Pin Function

Classification	Symbol	I/O	Name	Function
Octa memory controller	OM_SCLK	O	Clock	clock output pin.
	OM_CS0#	O	Chip select 0	Chip select signal for the external memory.
	OM_CS1#	O	Chip select 1	Chip select signal for the external memory.
	OM_DQS	I/O	Data strobe	Read: data strobe pin. Write: data Mask pin.
	OM_SIO7 to OM_SIO0	I/O	Data	Data I/O pins.
	OM_RESET#	O	Reset output	Reset signal for the external memory.
	OM_ECS#	I	ECC error detection	ECC error detection pin from the external memory.
Ethernet controller	ET1_TXCLK, ET0_TXCLK	I	Transmit clock	MII clock pin for transmission.
	ET1_TXEN, ET0_TXCLK	O	Transmit enable	MII transmit data enable pin
	ET0_TXD3 to ET0_TXD0, ET1_TXD3 to ET1_TXD0	O	Transmit data	MII transmit data pins.
	ET1_COL, ET0_COL	I	Collision detection	MII collision detection pin.
	ET1_TXER, ET0_TXER	O	Transmit error	MII transmit error output pin.
	ET1_RXCLK, ET0_RXCLK	I	Receive clock	MII receive clock pin
	ET1_RXDV, ET0_RXDV	I	Receive enable	MII receive data enable pin
	ET1_RXD3 to ET1_RXD0, ET0_RXD3 to ET0_RXD0	I	Receive data	MII receive data pins.
	ET1_RXER, ET0_RXER	I	Receive error	MII receive error input pin.
	ET1_CRS, ET0_CRS	I	Carrier detection	MII carrier detection pin.
	REF50CK0, REF50CK1	I	Reference clock	RMII reference clock pin.
	RMII0_CRS_DV, RMII1_CRS_DV	I	Carrier detection and Receive enable	RMII carrier detection and Receive enable pin.
	RMII0_TXD1, RMII0_TXD0, RMII1_TXD1, RMII1_TXD0	O	Transmit data	RMII transmit data pin.
	RMII0_TXD_EN, RMII1_TXD_EN	O	Transmit enable	RMII transmit enable pin.
	RMII0_RXER, RMII1_RXER	I	Receive error	RMII receive error input pin.
	RMII0_RXD1, RMII0_RXD0, RMII1_RXD1, RMII1_RXD0	I	Receive data	RMII receive data pin.
	ET1_MDC, ET0_MDC	O	Management data clock	Clock pin for information transfer via MDIO.
	ET1_MDIO, ET0_MDIO	I/O	Management data I/O	Bidirectional pin for exchange of management data.
	ET1_LINKSTA, ET0_LINKSTA	I	Status	Link status pin from the PHY-LSI.
	ET1_EXOUT, ET0_EXOUT	O	General purpose output	General purpose output pin.
	ET1_WOL, ET0_WOL	O	Wake-On-LAN	Indicate the LSI receive a Magic Packet.
	ET0_SCLKIN, ET1_SCLKIN	I	SCLKA clock	Clock signal supplied to the statistical time correction algorithm unit.

Table 1.3 Pin Function

Classification	Symbol	I/O	Name	Function
NAND flash controller	NFALE	O	Flash memory address latch enable	Asserted for address output and negated for data I/O.
	NFRE#	O	Flash memory read enable	Reads data at falling edge.
	NFCE#	O	Flash memory chip enable	Enables the flash memory connected to this LSI.
	NFCLE	O	Flash memory command latch enable	Asserted at command output.
	NFRB#	I	Flash memory ready/busy	High level indicates ready state and low level indicates busy state.
	NFWE#	O	Flash memory write enable	Flash memory latches commands, addresses, and data at falling edge.
	NFDATA[7:0]	I/O	Flash memory data	Data I/O pins.
USB 2.0 host/function module	DP1, DP0	I/O	USB 2.0 host/function module D+ data	D+ data pins for USB 2.0 host/function module bus.
	DM1, DM0	I/O	USB 2.0 host/function module D– data	D– data pins for USB 2.0 host/function module bus.
	VBUSIN1, VBUSIN0	I	VBUS input	A pin for monitoring connection of the USB cable. Step down the VBUS voltage to 3.3 V. Connect this pin to the Vbus pin of the USB bus so that it can detect connection and disconnection of the Vbus pin.
	VBUSEN1, VBUSEN0	O	VBUS output	VBUS power supply enable pin.
	OVRCUR1, OVRCUR0	I	Overcurrent input	Overcurrent input pin.
	OTG_EXICEN1, OTG_EXICEN0	O	OTG Power supply IC control	OTG Power supply IC control pin.
	OTG_ID1, OTG_ID0	I	OTG Power supply IC ID	OTG Power supply IC ID pin.
	CC1_Rd1, CC1_Ra1, CC2_Rd1, CC2_Ra1, CC1_Rd0, CC1_Ra0, CC2_Rd0, CC2_Ra0	I	CC input	These pins are used to monitor the state of the resistance on the CC pins of the USB Type-C connector. Check the state of the resistance of the USB Type-C connector and input the result of checking through the CC pins of the RZ/A2M.
	RREF1, RREF0	I	Reference input	Connected to USBVss via 2.2kΩ ± 1% resistance.
	USB_X1	I	Crystal resonator for USB 2.0 host/function module	Connected to a crystal resonator for USB 2.0 host/function module.
	USB_X2	O		
	USBAPVcc1, USBAPVcc0	I	Power supply for transceiver analog pins	Power supply for pins.
	USBDPVcc1, USBDPVcc0	I	Power supply for transceiver digital pins	Power supply for pins.
	USBVss	I	Ground for transceiver pins	Ground for pins.
Video display controller 6	LCD0_DATA23 to LCD0_DATA0	O	Output data	Data output pins for panel.
	LCD0_TCON6 to LCD0_TCON0	O	Panel timing adjustment output	Output pins for panel timing adjustment.
	LCD0_CLK	O	Panel clock	Panel clock output pins.
	LCD0_EXTCLK	I	Panel clock source	Panel clock source input pins.
	DV0_DATA23 to DV0_DATA0	I	Input data	Data input pins for graphics data.
	DV0_VSYNC	I	VSYNC input	VSYNC input pins.
	DV0_HSYNC	I	HSYNC input	HSYNC input pins.
	DV0_CLK	I	Input clock	Clock input signal pins for graphics data.

Table 1.3 Pin Function

Classification	Symbol	I/O	Name	Function
LVDS output interface	TXCLKOUTP, TXCLKOUTM	O	Output clock	LVDS differential clock output pins.
	TXOUT2P to TXOUT0P, TXOUT2M to TXOUT0M	O	Output data	LVDS differential data output pins.
	LVDSAPVcc	I	LVDS analog power supply	Power supply for LVDS output.
	LVDSPLLVcc	I	LVDS PLL power supply	Power supply for LVDS PLL.
Capture engine unit	VIO_D15 to VIO_D0	I	Input data	Graphics data input pins.
	VIO_CLK	I	Input clock	Graphics data clock input pin.
	VIO_VD	I	VSYNC input	VSYNC input pin.
	VIO_HD	I	HSYNC input	HSYNC input pin.
	VIO_FLD	I	FIELD input	Input pin for field information
MIPI CSI-2 interface	CSI_DATA0P	I	Input data	Positive data receive pin for CSI Lane 0 differential input.
	CSI_DATA0N	I	Input data	Negative data receive pin for CSI Lane 0 differential input.
	CSI_DATA1P	I	Input data	Positive data receive pin for CSI Lane 1 differential input.
	CSI_DATA1N	I	Input data	Negative data receive pin for CSI Lane 1 differential input.
	CSI_CLKP	I	Input clock	Positive receive pin for CSI clock Lane input.
	CSI_CLKN	I	Input clock	Negative receive pin for CSI clock Lane input.
	MIPIAVcc18	I	MIPI analog power supply	Power supply for the MIPI analog circuits.
SD/MMC host interface	SD0_CLK, SD1_CLK	O	SD/MMC clock	Output pins for SD/MMC clock.
	SD0_CMD, SD1_CMD	I/O	SD/MMC command	SD/MMC command output and response input signals.
	SD0_DAT7 to SD0_DAT0, SD1_DAT3 to SD1_DAT0	I/O	SD/MMC data	SD/MMC data bus signals.
	SD0_CD, SD1_CD	I	SD/MMC card detection	SD/MMC card detection.
	SD0_WP, SD1_WP	I	SD/MMC write protection	SD/MMC write protection signals.
	SD0_RST#	O	SD/MMC reset	SD/MMC reset signal.
A/D converter	AN007 to AN000	I	Analog input pins	Analog input pins.
	ADTRG#	I	A/D conversion trigger input	External trigger input pin for starting A/D conversion.
	AVcc	I	Analog power supply and reference voltage	Analog power supply and reference voltage pin for A/D converter.
	AVss	I	Analog ground	Ground pin for A/D converter.
General I/O ports	P0_0 to PK_5	I/O	General port	General purpose I/O port pins 176-pin products: 47 pins 256-pin products: 92 pins 272-pin products: 92 pins 324-pin products: 128 pins
	PD_0 to PD_7	I/O	General port	8 general input port pins with open-drain output.
	P5_0 to P5_7, PL_0 to PL_4, JP0_0	I	General port	14 general input port pins
	JP0_1	O	General port	1 general output port pin

Table 1.3 Pin Function

Classification	Symbol	I/O	Name	Function
Debugger interface	TCK/SWDCLK	I	Test clock	Test-clock input pin. Also used as the input clock pin for serial wire debugging
	TMS/SWDIO	I/O	Test mode select	Test-mode select signal input pin. Also used as the I/O data pin for serial wire debugging
	TDI	I	Test data input	Serial input pin for instructions and data.
	TDO/SWO	O	Test data output	Serial output pin for instructions and data.
	TRST#	I	Test reset	Initialization-signal input pin.
	TRACEDATA3 to TRACEDATA0	O	Data output	Trace data output pins.
	TRACECLK	O	Clock output	Trace clock output pin.
	TRACECTL	O	Enable output	Trace enable output pin.
Dynamic Reconfigurable Processor (DRP)*	DRP31 to DRP00	I/O	DRP I/O pins	32 DRP input/output pins

Note: * Only in products with a DRP

1.6 List of Pins

Table 1.4 List of Pins (324-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
A1	Vcc																		
A2	QSPI1_IO3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A3	QSPI1_SPCLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A4	RPC_WP#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A5	QSPI0_IO3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A6	PVcc_SPI																		
A7	Vss																		
A8	PVcc																		
A9	PF_4	I(s)/O	-	-	RxD2	I(s)	DV0_DATA19	I(s)	LCD0_DATA4	O	MTIOC6A	I(s)/O	SSIBCK0	I(s)/O	IRQ1	I(s)	-	-	(1)
A10	PE_6	I(s)/O	-	-	ET0_MDIO	I(s)/O	VIO_D2	I(s)	SSIRxD0	I(s)	MTIOC0D	I(s)/O	CC2_Rd1	I(s)	-	-	-	-	(1)
A11	PL_2	I(s)	MD_BOOT2	I(s)	-	-	-	-	-	-	-	-	IRQ6	I(s)	-	-	-	-	(5)
A12	PE_5	I(s)/O	-	-	ET0_MDC	O	VIO_D3	I(s)	SSITxD0	O	MTIOC0C	I(s)/O	CC1_Rd1	I(s)	-	-	-	-	(1)
A13	P8_4	I(s)/O	-	-	A4	O	DRP20*	I(s)/O	DV0_DATA13	I(s)	SSL00	I(s)/O	SSIRxD3	I(s)	-	-	-	-	(1)
A14	P8_6	I(s)/O	-	-	A6	O	DRP18*	I(s)/O	DV0_DATA11	I(s)	MOSI0	I(s)/O	SSILRCK3	I(s)/O	-	-	-	-	(1)
A15	PE_4	I(s)/O	-	-	ET0_CRS	I(s)	VIO_D4	I(s)	SSILRCK0	I(s)/O	MTIOC0B	I(s)/O	-	-	-	-	RMII0_CRS_DV	I(s)	(1)
A16	P9_1	I(s)/O	-	-	A9	O	DRP15*	I(s)/O	DV0_DATA8	I(s)	RxD4	I(s)	SSILRCK2	I(s)/O	-	-	-	-	(1)
A17	PVcc																		
A18	Vss																		
A19	PE_1	I(s)/O	-	-	ET0_RXD0	I(s)	VIO_D7	I(s)	RxD2	I(s)	POE8#	I(s)	VBUSIN1	I(s)	IRQ1	I(s)	RMII0_RXD0	I(s)	(1)
A20	PA_4	I(s)/O	-	-	A20	O	DV0_DATA9	I(s)	LCD0_DATA14	O	SCI_TXD0	I(s)/O	MTIOC0C	I(s)/O	-	-	-	-	(1)
A21	CKIO	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(11)
A22	Vss																-	-	
B1	PK_1	I(s)/O	-	-	ET1_TXD0	O	NFDATA4	I(s)/O	CC1_Ra0	I(s)	CAN_CLK	I(s)	SSIDATA2	I(s)/O	-	-	RMII1_TXD0	O	(1)
B2	Vcc																		
B3	QSPI1_IO1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B4	QSPI1_IO0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B5	RPC_RESET#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B6	QSPI0_IO1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B7	QSPI0_SPCLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B8	PF_5	I(s)/O	-	-	TxD2	O	DV0_DATA20	I(s)	LCD0_DATA3	O	MTIOC6B	I(s)/O	SSILRCK0	I(s)/O	-	-	-	-	(1)
B9	P6_3	I(s)/O	-	-	ET0_TXD1	O	VIO_HD	I(s)	TxD3	O	POE0#	I(s)	-	-	-	-	RMII0_TXD1	O	(1)
B10	PH_0	I(s)/O	-	-	AUDIO_CLK	I(s)	VIO_D1	I(s)	GTIOC4A	I(s)/O	MTIOC1A	I(s)/O	CC1_Rd0	I(s)	IRQ3	I(s)	-	-	(1)
B11	PL_3	I(s)	MD_BOOT1	I(s)	-	-	-	-	-	-	-	-	IRQ7	I(s)	-	-	-	-	(5)
B12	PL_1	I(s)	MD_CLK	I(s)	-	-	-	-	-	-	-	-	IRQ5	I(s)	-	-	-	-	(5)
B13	P8_3	I(s)/O	-	-	A3	O	DRP21*	I(s)/O	DV0_DATA14	I(s)	MTIOC6A	I(s)/O	GTIOC3A	I(s)/O	-	-	-	-	(1)
B14	PF_2	I(s)/O	-	-	TxD3	O	DV0_DATA17	I(s)	LCD0_DATA6	O	MTIOC7C	I(s)/O	MISO1	I(s)/O	-	-	-	-	(1)
B15	P8_7	I(s)/O	-	-	A7	O	DRP17*	I(s)/O	DV0_DATA10	I(s)	RSPCK0	I(s)/O	SSIBCK3	I(s)/O	-	-	-	-	(1)
B16	PE_3	I(s)/O	-	-	ET0_RXER	I(s)	VIO_D5	I(s)	SSIBCK0	I(s)/O	MTIOC0A	I(s)/O	-	-	-	-	RMII0_RXER	I(s)	(1)
B17	PA_0	I(s)/O	-	-	A16	O	DV0_DATA13	I(s)	LCD0_DATA10	O	SCI_TXD1	I(s)/O	MTIOC8C	I(s)/O	-	-	-	-	(1)
B18	PA_3	I(s)/O	-	-	A19	O	DV0_DATA10	I(s)	LCD0_DATA13	O	SCI_CTS0#/RTS0#	I(s)/O	MTIOC0D	I(s)/O	-	-	-	-	(1)

Table 1.4 List of Pins (324-Pin BGA)

Ball Number	Port Function/ Dedicated Function Symbol	I/O	Mode Function Symbol	I/O	Function 1 Symbol	I/O	Function 2 Symbol	I/O	Function 3 Symbol	I/O	Function 4 Symbol	I/O	Function 5 Symbol	I/O	Function 6 Symbol	I/O	Function 7 Symbol	I/O	Simplified Circuit Diagram Figure 51.1 to Figure 51.15
B19	PA_5	I(s)/O	-	-	A21	O	DV0_DATA8	I(s)	LCD0_DATA15	O	SCI_RXD0	I(s)/O	MTIOC0B	I(s)/O	IRQ5	I(s)	-	-	(1)
B20	PA_6	I(s)/O	-	-	A22	O	DV0_DATA7	I(s)	LCD0_DATA16	O	SCI_SCK0	I(s)/O	MTIOC0A	I(s)/O	-	-	-	-	(1)
B21	Vss																		
B22	PVcc																		
C1	PH_2	I(s)/O	-	-	CTS2#	I(s)/O	DV0_DATA22	I(s)	LCD0_DATA1	O	MTIOC6D	I(s)/O	SSIRxD0	I(s)	-	-	-	-	(1)
C2	P8_2	I(s)/O	-	-	A2	O	DRP22*	I(s)/O	DV0_DATA15	I(s)	GTIOC5A	I(s)/O	IRQ2	I(s)	-	-	-	-	(1)
C3	Vcc																		
C4	QSPI1_SSL	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C5	RPC_INT#	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C6	QSPI0_SSL	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C7	QSPI0_IO0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C8	P6_1	I(s)/O	-	-	ET0_TXEN	O	VIO_CLK	I(s)	SCK3	I(s)/O	MTIOC2A	I(s)/O	-	-	-	-	RMII0_TXD_EN	O	(1)
C9	P6_2	I(s)/O	-	-	ET0_TXD0	O	VIO_VD	I(s)	RxD3	I(s)	MTIOC2B	I(s)/O	OTG_EXICEN1	O	IRQ0	I(s)	RMII0_TXD0	O	(1)
C10	PH_1	I(s)/O	-	-	AUDIO_XOUT	O	VIO_D0	I(s)	GTIOC4B	I(s)/O	MTIOC1B	I(s)/O	CC2_Rd0	I(s)	IRQ2	I(s)	-	-	(1)
C11	PL_4	I(s)	MD_BOOT0	I(s)	-	-	-	-	-	-	-	-	IRQ0	I(s)	-	-	-	-	(5)
C12	PL_0	I(s)	MD_CLKS	I(s)	-	-	-	-	-	-	-	-	IRQ4	I(s)	-	-	-	-	(5)
C13	P8_5	I(s)/O	-	-	A5	O	DRP19*	I(s)/O	DV0_DATA12	I(s)	MISO0	I(s)/O	SSITxD3	O	-	-	-	-	(1)
C14	PF_1	I(s)/O	-	-	RxD3	I(s)	DV0_DATA16	I(s)	LCD0_DATA7	O	MTIOC7B	I(s)/O	MOSI1	I(s)/O	IRQ4	I(s)	-	-	(1)
C15	P9_0	I(s)/O	-	-	A8	O	DRP16*	I(s)/O	DV0_DATA9	I(s)	TxD4	O	SSIDATA2	I(s)/O	-	-	-	-	(1)
C16	PE_2	I(s)/O	-	-	ET0_RXD1	I(s)	VIO_D6	I(s)	TxD2	O	POE10#	I(s)	-	-	-	-	RMII0_RXD1	I(s)	(1)
C17	PA_2	I(s)/O	-	-	A18	O	DV0_DATA11	I(s)	LCD0_DATA12	O	SCI_SCK1	I(s)/O	MTIOC8A	I(s)/O	-	-	-	-	(1)
C18	PG_0	I(s)/O	-	-	ET0_TXCLK	I(s)	VIO_D8	I(s)	RSPCK0	I(s)/O	MTIOC3A	I(s)/O	HM_RSTO#	I(s)	-	-	-	-	(1)
C19	PB_0	I(s)/O	-	-	A24	O	DV0_DATA5	I(s)	LCD0_DATA18	O	SSITxD1	O	POE8#	I(s)	-	-	-	-	(1)
C20	Vss																		
C21	PD_7	I(s)/O(o)	-	-	RIIC3SDA	I(s)/O(o)	IRQ7	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
C22	PD_3	I(s)/O(o)	-	-	RIIC1SDA	I(s)/O(o)	IRQ3	I(s)	MTCLKD	I(s)	GTETRGD	I(s)	-	-	-	-	-	-	(4)
D1	PVcc																		
D2	BSCANP	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(12)
D3	P8_1	I(s)/O	-	-	A1	O	DRP23*	I(s)/O	DV0_DATA16	I(s)	GTIOC5B	I(s)/O	IRQ3	I(s)	-	-	-	-	(1)
D4	Vcc																		
D5	QSPI1_IO2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
D6	Vss																		
D7	QSPI0_IO2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
D8	PK_0	I(s)/O	-	-	ET1_TXEN	O	NFDATA3	I(s)/O	CC1_Rd0	I(s)	MTIOC1B	I(s)/O	SSIBCK2	I(s)/O	-	-	RMII1_TXD_EN	O	(1)
D9	PF_6	I(s)/O	-	-	RTS2#	I(s)/O	DV0_DATA21	I(s)	LCD0_DATA2	O	MTIOC6C	I(s)/O	SSITxD0	O	-	-	-	-	(1)
D10	PE_0	I(s)/O	-	-	ET0_RXCLK	I(s)	VIO_FLD	I(s)	SCK2	I(s)/O	POE4#	I(s)	-	-	-	-	REF50CK0	I(s)	(1)
D11	PF_3	I(s)/O	-	-	SCK2	I(s)/O	DV0_DATA18	I(s)	LCD0_DATA5	O	MTIOC7D	I(s)/O	SSL10	I(s)/O	-	-	-	-	(1)
D12	PVcc																		
D13	Vss																		
D14	PF_0	I(s)/O	-	-	SCK3	I(s)/O	DV0_DATA15	I(s)	LCD0_DATA8	O	MTIOC7A	I(s)/O	RSPCK1	I(s)/O	-	-	-	-	(1)
D15	P8_0	I(s)/O	-	-	A0	O	DV0_DATA14	I(s)	LCD0_DATA9	O	SCI_CTS1#/RTS1#	I(s)/O	MTIOC8D	I(s)/O	-	-	-	-	(1)

Table 1.4 List of Pins (324-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
D16	PA_1	I(s)/ O	-	-	A17	O	DV0_ DATA12	I(s)	LCD0_ DATA11	O	SCI_ RXD1	I(s)/ O	MTIOC8B	I(s)/ O	IRQ6	I(s)	-	-	(1)
D17	PA_7	I(s)/ O	-	-	A23	O	DV0_ DATA6	I(s)	LCD0_ DATA17	O	SSIRxD1	I(s)	POE10#	I(s)	-	-	-	-	(1)
D18	PVcc																		
D19	Vss																		
D20	PD_6	I(s)/ O(o)	-	-	RIIC3SCL	I(s)/ O(o)	IRQ6	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
D21	PD_4	I(s)/ O(o)	-	-	RIIC2SCL	I(s)/ O(o)	IRQ4	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
D22	PD_1	I(s)/ O(o)	-	-	RIIC0SDA	I(s)/ O(o)	IRQ1	I(s)	MTCLKB	I(s)	GTETRGB	I(s)	-	-	-	-	-	-	(4)
E1	Vss																		
E2	PH_3	I(s)/ O	-	-	HM_RSTO#	I(s)	RTS2#	I(s)/ O	GTIOC6A	I(s)/ O	MTIOC2A	I(s)/ O	SD0_CD	I(s)	IRQ3	I(s)	-	-	(1)
E3	PK_3	I(s)/ O	-	-	ET1_RXCLK	I(s)	NFDATA6	I(s)/ O	CC2_Rd0	I(s)	CAN0RX_ DATARATE_ EN	O	MOSI0	I(s)/ O	-	-	REF50CK1	I(s)	(1)
E4	PK_2	I(s)/ O	-	-	ET1_TXD1	O	NFDATA5	I(s)/ O	VBUSEN1	O	CAN0RX	I(s)	RSPCK0	I(s)/ O	IRQ5	I(s)	RMII1_TXD1	O	(1)
E19	PD_5	I(s)/ O(o)	-	-	RIIC2SDA	I(s)/ O(o)	IRQ5	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
E20	PD_2	I(s)/ O(o)	-	-	RIIC1SCL	I(s)/ O(o)	IRQ2	I(s)	MTCLKC	I(s)	GTETRGC	I(s)	-	-	-	-	-	-	(4)
E21	TCK/ SWDCLK	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(7)
E22	JP0_0	I(s)	-	-	TDI	I	-	-	-	-	-	-	-	-	-	-	-	-	(6)
F1	PVcc_HO																		
F2	HM_CS0#/ OM_CS0#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
F3	HM_CK/ OM_SCLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
F4	PF_7	I(s)/ O	-	-	GTETRGD	I(s)	DV0_ DATA23	I(s)	LCD0_ DATA0	O	MTCLKD	I(s)	IRQ1	I(s)	-	-	-	-	(1)
F19	PD_0	I(s)/ O(o)	-	-	RIIC0SCL	I(s)/ O(o)	IRQ0	I(s)	MTCLKA	I(s)	GTETRGA	I(s)	-	-	-	-	-	-	(4)
F20	TRST#	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(10)
F21	TMS/SWDIO	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(9)
F22	PB_1	I(s)/ O	-	-	A25	O	DV0_ DATA4	I(s)	LCD0_ DATA19	O	SSILRCK1	I(s)/ O	POE4#	I(s)	-	-	-	-	(1)
G1	HM_DQ1/ OM_SIO1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G2	HM_RWDS/ OM_DQS	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G3	HM_CS1#/ OM_CS1#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G4	HM_CK#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G19	JP0_1	O	-	-	TDO/SWO	O	-	-	-	-	-	-	-	-	-	-	-	-	(8)
G20	PB_2	I(s)/ O	-	-	BS#	O	DV0_ DATA3	I(s)	LCD0_ DATA20	O	SSIBCK1	I(s)/ O	POE0#	I(s)	-	-	-	-	(1)
G21	PB_3	I(s)/ O	-	-	CS0#	O	DV0_ DATA2	I(s)	LCD0_ DATA21	O	SSIDATA2	I(s)/ O	CTS0#	I(s)/ O	-	-	-	-	(1)
G22	P9_2	I(s)/ O	-	-	A10	O	DRP14*	I(s)/ O	DV0_ DATA7	I(s)	SCK4	I(s)/ O	SSIBCK2	I(s)/ O	-	-	-	-	(1)
H1	HM_DQ4/ OM_SIO4	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H2	HM_DQ2/ OM_SIO2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H3	HM_DQ3/ OM_SIO3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H4	HM_DQ0/ OM_SIO0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H19	PB_4	I(s)/ O	-	-	CS1#	O	DV0_ DATA1	I(s)	LCD0_ DATA22	O	SSILRCK2	I(s)/ O	RTS0#	I(s)/ O	-	-	-	-	(1)
H20	P9_3	I(s)/ O	-	-	A11	O	DRP13*	I(s)/ O	DV0_ DATA6	I(s)	-	-	SSIRxD0	I(s)	-	-	-	-	(1)
H21	PB_5	I(s)/ O	-	-	WAIT#	I(s)	DV0_ DATA0	I(s)	LCD0_ DATA23	O	SSIBCK2	I(s)/ O	TxD0	O	-	-	-	-	(1)

Table 1.4 List of Pins (324-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
H22	P9_5	I(s)/O	-	-	A13	O	DRP11*	I(s)/O	DV0_DATA4	I(s)	-	-	SSILRCK0	I(s)/O	-	-	-	-	(1)
J1	HM_RESET#/OM_RESET#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J2	HM_DQ6/OM_SIO6	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J3	HM_DQ7/OM_SIO7	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J4	HM_DQ5/OM_SIO5	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J9	Vcc																		
J10	Vss																		
J11	Vss																		
J12	Vss																		
J13	Vss																		
J14	Vcc																		
J19	P9_4	I(s)/O	-	-	A12	O	DRP12*	I(s)/O	DV0_DATA5	I(s)	-	-	SSITxD0	O	-	-	-	-	(1)
J20	P7_7	I(s)/O	-	-	RD#	O	DV0_HSYNC	I(s)	LCD0_TCON0	O	GTIOC3B	I(s)/O	RxD0	I(s)	-	-	-	-	(1)
J21	P7_6	I(s)/O	-	-	AH#	O	DV0_VSYNC	I(s)	LCD0_TCON1	O	GTIOC3A	I(s)/O	SCK0	I(s)/O	-	-	-	-	(1)
J22	P9_6	I(s)/O	-	-	A14	O	DRP10*	I(s)/O	DV0_DATA3	I(s)	-	-	SSIBCK0	I(s)/O	-	-	-	-	(1)
K1	Vss																		
K2	PJ_6	I(s)/O	-	-	GTETRGC	I(s)	NFCE#	O	LCD0_CLK	O	MTCLKC	I(s)	IRQ0	I(s)	-	-	-	-	(1)
K3	PH_4	I(s)/O	-	-	HM_INT#/OM_ECS#	I(s)	CTS2#	I(s)/O	GTIOC6B	I(s)/O	MTIOC2B	I(s)/O	SD0_WP	I(s)	IRQ2	I(s)	-	-	(1)
K4	PJ_0	I(s)/O	-	-	TRACECLK	O	SPDIF_OUT	O	-	-	SCK1	I(s)/O	SSIRxD3	I(s)	-	-	-	-	(1)
K9	Vcc																		
K10	Vss																		
K11	Vss																		
K12	Vss																		
K13	Vss																		
K14	Vcc																		
K19	P9_7	I(s)/O	-	-	A15	O	DRP09*	I(s)/O	DV0_DATA2	I(s)	-	-	SD1_WP	I(s)	-	-	-	-	(1)
K20	PG_1	I(s)/O	-	-	ET0_TXD2	O	VIO_D9	I(s)	MOSI0	I(s)/O	MTIOC3C	I(s)/O	HM_INT#/OM_ECS#	I(s)	-	-	-	-	(1)
K21	P7_5	I(s)/O	-	-	CKE	O	DRP08*	I(s)/O	DV0_DATA1	I(s)	CTS1#	I(s)/O	OVRCUR1	I(s)	-	-	-	-	(1)
K22	PG_2	I(s)/O	-	-	ET0_TXD3	O	VIO_D10	I(s)	MISO0	I(s)/O	MTIOC3B	I(s)/O	GTIOC0A	I(s)/O	IRQ4	I(s)	-	-	(1)
L1	PVcc																		
L2	P0_1	I(s)/O	-	-	D1	I/O	DRP25*	I(s)/O	DV0_DATA18	I(s)	MTIOC6C	I(s)/O	GTIOC4A	I(s)/O	-	-	-	-	(2)
L3	P0_0	I(s)/O	-	-	D0	I/O	DRP24*	I(s)/O	DV0_DATA17	I(s)	MTIOC6B	I(s)/O	GTIOC3B	I(s)/O	-	-	-	-	(2)
L4	PJ_7	I(s)/O	-	-	GTETRGB	I(s)	NFDATA0	I(s)/O	LCD0_EXTCLK	I(s)	MTCLKB	I(s)	-	-	-	-	-	-	(1)
L9	Vcc																		
L10	Vss																		
L11	Vss																		
L12	Vss																		
L13	Vss																		
L14	Vcc																		
L19	P7_1	I(s)/O	-	-	RD/WR#	O	DRP05*	I(s)/O	DV0_VSYNC	I(s)	RxD1	I(s)	CC1_Ra1	I(s)	-	-	-	-	(1)
L20	P7_4	I(s)/O	-	-	CAS#	O	DRP07*	I(s)/O	DV0_DATA0	I(s)	RTS1#	I(s)/O	CC2_Ra1	I(s)	-	-	-	-	(1)
L21	P7_3	I(s)/O	-	-	RAS#	O	DRP06*	I(s)/O	DV0_HSYNC	I(s)	TxD1	O	CC2_Rd1	I(s)	-	-	-	-	(1)

Table 1.4 List of Pins (324-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
L22	P7_2	I(s)/O	-	-	CS4#	O	DV0_CLK	I(s)	LCD0_TCON2	O	TEND0	O	CC2_Ra0	I(s)	-	-	-	-	(1)
M1	P0_2	I(s)/O	-	-	D2	I/O	DRP26*	I(s)/O	DV0_DATA19	I(s)	MTIOC6D	I(s)/O	GTIOC4B	I(s)/O	-	-	-	-	(2)
M2	P0_5	I(s)/O	-	-	D5	I/O	DRP29*	I(s)/O	DV0_DATA22	I(s)	MTIOC7C	I(s)/O	GTIOC7A	I(s)/O	-	-	-	-	(2)
M3	P0_4	I(s)/O	-	-	D4	I/O	DRP28*	I(s)/O	DV0_DATA21	I(s)	MTIOC7B	I(s)/O	GTIOC6B	I(s)/O	-	-	-	-	(2)
M4	P0_3	I(s)/O	-	-	D3	I/O	DRP27*	I(s)/O	DV0_DATA20	I(s)	MTIOC7A	I(s)/O	GTIOC6A	I(s)/O	-	-	-	-	(2)
M9	Vcc																		
M10	Vss																		
M11	Vss																		
M12	Vss																		
M13	Vss																		
M14	Vcc																		
M19	P6_6	I(s)/O	-	-	CS2#	O	DRP02*	I(s)/O	LCD0_TCON4	O	DREQ0	I(s)	CC1_Ra0	I(s)	-	-	-	-	(1)
M20	P6_0	I(s)/O	-	-	ADTRG#	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	(1)
M21	P7_0	I(s)/O	-	-	WE1#/DQMU	O	DRP04*	I(s)/O	DV0_CLK	I(s)	SCK1	I(s)/O	CC1_Rd1	I(s)	-	-	-	-	(1)
M22	PVcc																		
N1	PJ_3	I(s)/O	-	-	TRACEDATA1	O	NFDATA0	I(s)/O	-	-	RTS1#	I(s)/O	SSILRCK3	I(s)/O	-	-	-	-	(1)
N2	PJ_1	I(s)/O	-	-	TRACECTL	O	SPDIF_IN	I(s)	-	-	RxD1	I(s)	VBUSIN0	I(s)	IRQ0	I(s)	-	-	(1)
N3	P0_6	I(s)/O	-	-	D6	I/O	DRP30*	I(s)/O	DV0_DATA23	I(s)	MTIOC7D	I(s)/O	GTIOC7B	I(s)/O	-	-	-	-	(2)
N4	PJ_2	I(s)/O	-	-	TRACEDATA0	O	NFCE#	O	-	-	TxD1	O	SSITxD3	O	-	-	-	-	(1)
N9	Vcc																		
N10	Vss																		
N11	Vss																		
N12	Vss																		
N13	Vss																		
N14	Vcc																		
N19	P6_5	I(s)/O	-	-	CS3#	O	DRP01*	I(s)/O	LCD0_TCON5	O	AUDIO_XOUT	O	CC1_Rd0	I(s)	-	-	-	-	(1)
N20	PG_3	I(s)/O	-	-	ET0_COL	I(s)	VIO_D11	I(s)	SSL00	I(s)/O	MTIOC3D	I(s)/O	GTIOC0B	I(s)/O	-	-	-	-	(1)
N21	P6_4	I(s)/O	-	-	CS5#	O	DRP00*	I(s)/O	LCD0_TCON6	O	AUDIO_CLK	I(s)	SD1_CD	I(s)	-	-	-	-	(1)
N22	Vss																		
P1	PH_6	I(s)/O	-	-	HM_INT#/OM_ECS#	I(s)	NFDATA3	I(s)/O	ET1_WOL	O	MTIC5V	I(s)	IRQ4	I(s)	-	-	-	-	(1)
P2	PH_5	I(s)/O	-	-	HM_RSTO#	I(s)	NFDATA2	I(s)/O	ET1_EXOUT/ET1_SCLKIN	I(s)/O	MTIC5U	I(s)	IRQ5	I(s)	-	-	-	-	(1)
P3	PK_5	I(s)/O	-	-	GTETRGA	I(s)	NFDATA1	I(s)/O	WDTOVF#/PERROUT#	O	MTCLKA	I(s)	-	-	-	-	-	-	(1)
P4	PVcc																		
P9	Vcc																		
P10	Vss																		
P11	Vss																		
P12	Vss																		
P13	Vss																		
P14	Vcc																		
P19	SD0_DAT7	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
P20	SD0_RST#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
P21	P6_7	I(s)/O	-	-	WE0#/DQML	O	DRP03*	I(s)/O	LCD0_TCON3	O	DACK0	O	CC2_Rd0	I(s)	-	-	-	-	(1)
P22	PVcc_SD0																		
R1	PVcc																		

Table 1.4 List of Pins (324-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
R2	PJ_4	I(s)/O	-	-	TRACE DATA2	O	NFDATA1	I(s)/O	-	-	CTS1#	I(s)/O	SSIBCK3	I(s)/O	-	-	-	-	(1)
R3	PJ_5	I(s)/O	-	-	TRACE DATA3	O	NFDATA2	I(s)/O	OVRCUR0	I(s)	MTIOC1A	I(s)/O	SSILRCK2	I(s)/O	IRQ4	I(s)	-	-	(1)
R4	Vss																		
R19	SD0_DAT2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
R20	SD0_DAT5	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
R21	SD0_DAT4	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
R22	SD0_DAT6	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
T1	AUDIO_X1	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
T2	AUDIO_X2	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
T3	P3_5	I(s)/O	-	-	ET1_RXD1	I(s)	NFCLE	O	CC2_Ra0	I(s)	CAN0TX_ DATARATE_ EN	O	SSL00	I(s)/O	-	-	RMII1_ RXD1	I(s)	(1)
T4	P3_2	I(s)/O	-	-	ET1_CRS	I(s)	NFRE#	O	CC1_Ra1	I(s)	CAN1RX_ DATARATE_ EN	O	MOSI2	I(s)/O	-	-	RMII1_ CRS_DV	I(s)	(1)
T19	SD0_DAT0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
T20	SD0_DAT1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
T21	SD0_DAT3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
T22	Vss																		
U1	Vss																		
U2	PK_4	I(s)/O	-	-	ET1_RXD0	I(s)	NFDATA7	I(s)/O	OVRCUR1	I(s)	CAN0TX	O	MISO0	I(s)/O	IRQ6	I(s)	RMII1_RXD0	I(s)	(1)
U3	P3_1	I(s)/O	-	-	ET1_RXER	I(s)	NFALE	O	VBUSEN0	I(s)	CAN1RX	I(s)	RSPCK2	I(s)/O	IRQ6	I(s)	RMII1_RXER	I(s)	(1)
U4	MIPIAVcc18																		
U19	SD1_DAT0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
U20	SD1_DAT2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
U21	SD0_CMD	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
U22	SD0_CLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
V1	CSI_CLKP	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
V2	CSI_CLKN	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
V3	PG_4	I(s)/O	-	-	ET0_TXER	O	VIO_D15	I(s)	RSPCK1	I(s)/O	MTIOC4A	I(s)/O	GTIOC1A	I(s)/O	-	-	-	-	(1)
V4	Vss																		
V19	P5_4	I(s)	-	-	AN004	I(a)	IRQ0	I(s)	SD1_CD	I(s)	-	-	-	-	-	-	-	-	(3)
V20	SD1_DAT1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
V21	SD1_DAT3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
V22	PVcc_SD1																		
W1	CSI_DATA0 P	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
W2	CSI_DATA0 N	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
W3	Vss																		
W4	PG_6	I(s)/O	-	-	ET0_RXD2	I(s)	VIO_D13	I(s)	MISO1	I(s)/O	MTIOC4C	I(s)/O	GTIOC2A	I(s)/O	IRQ5	I(s)	-	-	(1)
W5	P1_0	I(s)/O	-	-	D7	I/O	DRP31*	I(s)/O	IRQ0	I(s)	CAN_CLK	I(s)	VBUSEN0	O	-	-	-	-	(2)
W6	P1_2	I(s)/O	-	-	D9	I/O	MTIOC8B	I(s)/O	IRQ2	I(s)	CAN0RX_ DATARATE_ EN	O	VBUSEN1	O	-	-	-	-	(2)
W7	P2_0	I(s)/O	-	-	D12	I/O	GTIOC6A	I(s)/O	IRQ5	I(s)	CAN1RX	I(s)	OTG_ EXICEN0	O	-	-	-	-	(2)
W8	PC_2	I(s)/O	-	-	OTG_ EXICEN0	O	NFDATA7	I(s)/O	ET1_TXD3	O	MISO2	I(s)/O	LCD0_ TCON5	O	-	-	-	-	(1)
W9	P4_3	I(s)/O	-	-	RTS0#	I(s)/O	TXOUT1M	O	SCI_CTS1#/ RTS1#	I(s)/O	SSILRCK1	I(s)/O	MTIOC8D	I(s)/O	IRQ3	I(s)	-	-	(1)
W10	LVDSAPVcc																		
W11	Vss																		
W12	LVDSPLLVcc																		
W13	USBDPVcc0																		

Table 1.4 List of Pins (324-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
W14	USBVss																		
W15	Vss																		
W16	PVcc																		
W17	Vss																		
W18	PLLVcc																		
W19	P5_2	I(s)	-	-	AN002	I(a)	IRQ6	I(s)	VBUSIN0	I(s)	-	-	-	-	-	-	-	-	(3)
W20	P5_6	I(s)	-	-	AN006	I(a)	IRQ2	I(s)	-	-	-	-	-	-	-	-	-	-	(3)
W21	SD1_CMD	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
W22	Vss																		
Y1	CSI_DATA1P	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
Y2	CSI_DATA1N	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
Y3	Vss																		
Y4	P3_3	I(s)/O	-	-	ET1_MDC	O	NFWE#	O	OTG_EXICEN0	O	CAN1TX	O	MISO2	I(s)/O	IRQ7	I(s)	-	-	(1)
Y5	P1_4	I(s)/O	-	-	D11	I/O	MTIOC8D	I(s)/O	IRQ4	I(s)	CAN0TX_DATARATE_EN	O	VBUSIN0	I(s)	-	-	-	-	(2)
Y6	PC_0	I(s)/O	-	-	VBUSIN1	I(s)	NFDATA5	I(s)/O	ET1_TXCLK	I(s)	RSPCK2	I(s)/O	IRQ2	I(s)	-	-	-	-	(1)
Y7	P2_2	I(s)/O	-	-	D14	I/O	GTIOC7A	I(s)/O	IRQ7	I(s)	CAN1TX	O	VBUSIN1	I(s)	-	-	-	-	(2)
Y8	P4_2	I(s)/O	-	-	TxD0	O	TXOUT1P	O	SCI_TXD1	I(s)/O	SSITxD1	O	MTIOC8C	I(s)/O	IRQ2	I(s)	-	-	(1)
Y9	P4_6	I(s)/O	-	-	ET0_EXOUT/ET0_SCLKIN	I(s)/O	TXCLKOUTP	O	SCI_TXD0	I(s)/O	TxD4	O	DACK0	O	-	-	-	-	(1)
Y10	NMI	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(10)
Y11	Vss																		
Y12	USBVss																		
Y13	USBVss																		
Y14	USBVss																		
Y15	USBVss																		
Y16	USBDPVcc1																		
Y17	PC_7	I(s)/O	-	-	OVRCUR0	I(s)	NFRB#	I(s)	ET1_RXD3	I(s)	SD1_WP	I(s)	LCD0_TCON0	O	IRQ6	I(s)	-	-	(1)
Y18	PC_6	I(s)/O	-	-	VBUSEN0	O	NFWE#	O	ET1_RXD2	I(s)	SD1_CD	I(s)	LCD0_TCON1	O	IRQ7	I(s)	-	-	(1)
Y19	P5_0	I(s)	-	-	AN000	I(a)	IRQ4	I(s)	SD0_CD	I(s)	SD1_CD	I(s)	-	-	-	-	-	-	(3)
Y20	P5_1	I(s)	-	-	AN001	I(a)	IRQ5	I(s)	SD0_WP	I(s)	SD1_WP	I(s)	-	-	-	-	-	-	(3)
Y21	P5_7	I(s)	-	-	AN007	I(a)	IRQ3	I(s)	-	-	-	-	-	-	-	-	-	-	(3)
Y22	SD1_CLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
AA1	MIPIAVcc18																		
AA2	Vss																		
AA3	P1_1	I(s)/O	-	-	D8	I/O	MTIOC8A	I(s)/O	IRQ1	I(s)	CAN0RX	I(s)	OVRCUR0	I(s)	-	-	-	-	(2)
AA4	P3_4	I(s)/O	-	-	ET1_MDIO	I(s)/O	NFRB#	I(s)	CC2_Ra1	I(s)	CAN1TX_DATARATE_EN	O	SSL20	I(s)/O	-	-	-	-	(1)
AA5	P3_0	I(s)/O	-	-	OTG_EXICEN1	O	NFDATA4	I(s)/O	ET1_LINKSTA	I(s)	MTIC5W	I(s)	IRQ3	I(s)	-	-	-	-	(1)
AA6	PC_1	I(s)/O	-	-	VBUSIN0	I(s)	NFDATA6	I(s)/O	ET1_TXD2	O	MOSI2	I(s)/O	LCD0_TCON6	O	-	-	-	-	(1)
AA7	P4_0	I(s)/O	-	-	SCK0	I(s)/O	TXOUT0P	O	SCI_SCK1	I(s)/O	SSIBCK1	I(s)/O	MTIOC8A	I(s)/O	IRQ0	I(s)	-	-	(1)
AA8	P4_4	I(s)/O	-	-	CTS0#	I(s)/O	TXOUT2P	O	SCI_CTS0#/RTS0#	I(s)/O	WDTOVF#/PERROUT#	O	OTG_EXICEN0	O	-	-	-	-	(1)
AA9	P4_7	I(s)/O	-	-	ET0_WOL	O	TXCLKOUTM	O	SCI_SCK0	I(s)/O	SCK4	I(s)/O	TEND0	O	-	-	-	-	(1)
AA10	USB_X2	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
AA11	DP0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
AA12	USBAPVcc0																		
AA13	RREF0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	

Table 1.4 List of Pins (324-Pin BGA)

Ball Number	Port Function/ Dedicated Function Symbol	I/O	Mode Function Symbol	I/O	Function 1 Symbol	I/O	Function 2 Symbol	I/O	Function 3 Symbol	I/O	Function 4 Symbol	I/O	Function 5 Symbol	I/O	Function 6 Symbol	I/O	Function 7 Symbol	I/O	Simplified Circuit Diagram Figure 51.1 to Figure 51.15
AA14	USBVss																		
AA15	DP1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
AA16	PVcc																		
AA17	PC_5	I(s)/O	-	-	VBUSEN1	O	NFRE#	O	ET1_RXDV	I(s)	SPDIF_OUT	O	LCD0_TCON2	O	IRQ0	I(s)	-	-	(1)
AA18	XTAL	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
AA19	PC_4	I(s)/O	-	-	OTG_ID1	I(s)	NFALE	O	ET1_TXER	O	SPDIF_IN	I(s)	LCD0_TCON3	O	IRQ1	I(s)	-	-	(1)
AA20	RTC_X2	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(14)
AA21	P5_3	I(s)	-	-	AN003	I(a)	IRQ7	I(s)	OTG_ID0	I(s)	-	-	-	-	-	-	-	-	(3)
AA22	P5_5	I(s)	-	-	AN005	I(a)	IRQ1	I(s)	SD1_WP	I(s)	-	-	-	-	-	-	-	-	(3)
AB1	Vss																		
AB2	PG_5	I(s)/O	-	-	ET0_RXDV	I(s)	VIO_D14	I(s)	MOSI1	I(s)/O	MTIOC4B	I(s)/O	GTIOC1B	I(s)/O	-	-	-	-	(1)
AB3	PG_7	I(s)/O	-	-	ET0_RXD3	I(s)	VIO_D12	I(s)	SSL10	I(s)/O	MTIOC4D	I(s)/O	GTIOC2B	I(s)/O	-	-	-	-	(1)
AB4	P1_3	I(s)/O	-	-	D10	I/O	MTIOC8C	I(s)/O	IRQ3	I(s)	CAN0TX	O	OTG_ID1	I(s)	-	-	-	-	(2)
AB5	P2_1	I(s)/O	-	-	D13	I/O	GTIOC6B	I(s)/O	IRQ6	I(s)	CAN1RX_DATARATE_EN	O	OTG_ID0	I(s)	-	-	-	-	(2)
AB6	P2_3	I(s)/O	-	-	D15	I/O	GTIOC7B	I(s)/O	WDTOVF#/PERROUT#	O	CAN1TX_DATARATE_EN	O	OTG_EXICEN1	O	-	-	-	-	(2)
AB7	P4_1	I(s)/O	-	-	RxD0	I(s)	TXOUT0M	O	SCI_RXD1	I(s)/O	SSIRxD1	I(s)	MTIOC8B	I(s)/O	IRQ1	I(s)	-	-	(1)
AB8	P4_5	I(s)/O	-	-	ET0_LINKSTA	I(s)	TXOUT2M	O	SCI_RXD0	I(s)/O	RxD4	I(s)	DREQ0	I(s)	-	-	-	-	(1)
AB9	RES#	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(12)
AB10	USB_X1	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
AB11	DM0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
AB12	USBA PVcc1																		
AB13	RREF1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
AB14	USBVss																		
AB15	DM1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
AB16	PVcc																		
AB17	PC_3	I(s)/O	-	-	OTG_ID0	I(s)	NFCLE	O	ET1_COL	I(s)	SSL20	I(s)/O	LCD0_TCON4	O	-	-	-	-	(1)
AB18	EXTAL	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
AB19	Vss																		
AB20	RTC_X1	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(14)
AB21	AVcc																		
AB22	AVss																		

[Legend]
(s): Schmitt
(a): Analog
(o): Open drain

Note: * Only in products with a DRP

Table 1.5 List of Pins (272-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
A1	Vcc																		
A2	QSPI1_IO3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A3	QSPI1_IO1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A4	QSPI1_ SPCLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A5	QSPI0_SSL	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A6	QSPI0_IO0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A7	Vss																		
A8	PVcc																		
A9	P6_3	I(s)/ O	-	-	ET0_TXD1	O	VIO_HD	I(s)	TxD3	O	POE0#	I(s)	-	-	-	-	RMII0_TXD1	O	(1)
A10	PE_6	I(s)/ O	-	-	ET0_MDIO	I(s)/ O	VIO_D2	I(s)	SSIRxD0	I(s)	MTIOC0D	I(s)/ O	CC2_Rd1	I(s)	-	-	-	-	(1)
A11	PL_4	I(s)	MD_BOOT0	I(s)	-	-	-	-	-	-	-	-	IRQ0	I(s)	-	-	-	-	(5)
A12	PL_2	I(s)	MD_BOOT2	I(s)	-	-	-	-	-	-	-	-	IRQ6	I(s)	-	-	-	-	(5)
A13	PL_1	I(s)	MD_CLK	I(s)	-	-	-	-	-	-	-	-	IRQ5	I(s)	-	-	-	-	(5)
A14	PF_1	I(s)/ O	-	-	RxD3	I(s)	DV0_DATA16	I(s)	LCD0_DATA7	O	MTIOC7B	I(s)/ O	MOSI1	I(s)/ O	IRQ4	I(s)	-	-	(1)
A15	PE_2	I(s)/ O	-	-	ET0_RXD1	I(s)	VIO_D6	I(s)	TxD2	O	POE10#	I(s)	-	-	-	-	RMII0_RXD1	I(s)	(1)
A16	PA_3	I(s)/ O	-	-	-	-	DV0_DATA10	I(s)	LCD0_ DATA13	O	SCI_CTS0#/ RTS0#	I(s)/ O	MTIOC0D	I(s)/ O	-	-	-	-	(1)
A17	PA_6	I(s)/ O	-	-	-	-	DV0_DATA7	I(s)	LCD0_ DATA16	O	SCI_SCK0	I(s)/ O	MTIOC0A	I(s)/ O	-	-	-	-	(1)
A18	Vss																		
A19	CKIO	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(11)
A20	Vss																		
B1	Vss																		
B2	Vcc																		
B3	QSPI1_IO2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B4	RPC_INT#	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B5	RPC_ RESET#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B6	QSPI0_IO3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B7	QSPI0_ SPCLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B8	P6_1	I(s)/ O	-	-	ET0_TXEN	O	VIO_CLK	I(s)	SCK3	I(s)/ O	MTIOC2A	I(s)/ O	-	-	-	-	RMII0_ TXD_EN	O	(1)
B9	PF_5	I(s)/ O	-	-	TxD2	O	DV0_DATA20	I(s)	LCD0_DATA3	O	MTIOC6B	I(s)/ O	SSILRCK0	I(s)/ O	-	-	-	-	(1)
B10	PH_1	I(s)/ O	-	-	AUDIO_XOU T	O	VIO_D0	I(s)	GTIOC4B	I(s)/ O	MTIOC1B	I(s)/ O	CC2_Rd0	I(s)	IRQ2	I(s)	-	-	(1)
B11	PF_3	I(s)/ O	-	-	SCK2	I(s)/ O	DV0_DATA18	I(s)	LCD0_DATA5	O	MTIOC7D	I(s)/ O	SSL10	I(s)/ O	-	-	-	-	(1)
B12	PL_0	I(s)	MD_CLKS	I(s)	-	-	-	-	-	-	-	-	IRQ4	I(s)	-	-	-	-	(5)
B13	PE_5	I(s)/ O	-	-	ET0_MDC	O	VIO_D3	I(s)	SSITxD0	O	MTIOC0C	I(s)/ O	CC1_Rd1	I(s)	-	-	-	-	(1)
B14	P8_0	I(s)/ O	-	-	-	-	DV0_DATA14	I(s)	LCD0_DATA9	O	SCI_CTS1#/ RTS1#	I(s)/ O	MTIOC8D	I(s)/ O	-	-	-	-	(1)
B15	PE_3	I(s)/ O	-	-	ET0_RXER	I(s)	VIO_D5	I(s)	SSIBCK0	I(s)/ O	MTIOC0A	I(s)/ O	-	-	-	-	RMII0_RXER	I(s)	(1)
B16	PA_2	I(s)/ O	-	-	-	-	DV0_DATA11	I(s)	LCD0_ DATA12	O	SCI_SCK1	I(s)/ O	MTIOC8A	I(s)/ O	-	-	-	-	(1)
B17	PA_4	I(s)/ O	-	-	-	-	DV0_DATA9	I(s)	LCD0_ DATA14	O	SCI_TXD0	I(s)/ O	MTIOC0C	I(s)/ O	-	-	-	-	(1)
B18	PB_0	I(s)/ O	-	-	-	-	DV0_DATA5	I(s)	LCD0_ DATA18	O	SSITxD1	O	POE8#	I(s)	-	-	-	-	(1)
B19	Vss																		
B20	PD_6	I(s)/ O(o)	-	-	RIIC3SCL	I(s)/ O(o)	IRQ6	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
C1	PH_2	I(s)/ O	-	-	CTS2#	I(s)/ O	DV0_DATA22	I(s)	LCD0_DATA1	O	MTIOC6D	I(s)/ O	SSIRxD0	I(s)	-	-	-	-	(1)
C2	PK_1	I(s)/ O	-	-	ET1_TXD0	O	NFDATA4	I(s)/ O	CC1_Ra0	I(s)	CAN_CLK	I(s)	SSIDATA2	I(s)/ O	-	-	RMII1_TXD0	O	(1)
C3	Vcc																		

Table 1.5 List of Pins (272-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
C4	QSPI1_SSL	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C5	RPC_WP#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C6	QSPI0_IO1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C7	QSPI0_IO2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C8	PF_6	I(s)/ O	-	-	RTS2#	I(s)/ O	DV0_DATA21	I(s)	LCD0_DATA2	O	MTIOC6C	I(s)/ O	SSITxD0	O	-	-	-	-	(1)
C9	PE_0	I(s)/ O	-	-	ET0_RXCLK	I(s)	VIO_FLD	I(s)	SCK2	I(s)/ O	POE4#	I(s)	-	-	-	-	REF50CK0	I(s)	(1)
C10	PF_4	I(s)/ O	-	-	RxD2	I(s)	DV0_DATA19	I(s)	LCD0_DATA4	O	MTIOC6A	I(s)/ O	SSIBCK0	I(s)/ O	IRQ1	I(s)	-	-	(1)
C11	PL_3	I(s)	MD_BOOT1	I(s)	-	-	-	-	-	-	-	-	IRQ7	I(s)	-	-	-	-	(5)
C12	PF_2	I(s)/ O	-	-	TxD3	O	DV0_DATA17	I(s)	LCD0_DATA6	O	MTIOC7C	I(s)/ O	MISO1	I(s)/ O	-	-	-	-	(1)
C13	PE_4	I(s)/ O	-	-	ET0_CRS	I(s)	VIO_D4	I(s)	SSILRCK0	I(s)/ O	MTIOC0B	I(s)/ O	-	-	-	-	RMII0_ CRS_DV	I(s)	(1)
C14	PA_0	I(s)/ O	-	-	-	-	DV0_DATA13	I(s)	LCD0_ DATA10	O	SCI_TXD1	I(s)/ O	MTIOC8C	I(s)/ O	-	-	-	-	(1)
C15	PE_1	I(s)/ O	-	-	ET0_RXD0	I(s)	VIO_D7	I(s)	RxD2	I(s)	POE8#	I(s)	VBUSIN1	I(s)	IRQ1	I(s)	RMII0_RXD0	I(s)	(1)
C16	PA_5	I(s)/ O	-	-	-	-	DV0_DATA8	I(s)	LCD0_ DATA15	O	SCI_RXD0	I(s)/ O	MTIOC0B	I(s)/ O	IRQ5	I(s)	-	-	(1)
C17	PG_0	I(s)/ O	-	-	ET0_TXCLK	I(s)	VIO_D8	I(s)	RSPCK0	I(s)/ O	MTIOC3A	I(s)/ O	HM_RSTO#	I(s)	-	-	-	-	(1)
C18	Vss																		
C19	PD_7	I(s)/ O(o)	-	-	RIIC3SDA	I(s)/ O(o)	IRQ7	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
C20	PD_5	I(s)/ O(o)	-	-	RIIC2SDA	I(s)/ O(o)	IRQ5	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
D1	PH_3	I(s)/ O	-	-	HM_RSTO#	I(s)	RTS2#	I(s)/ O	GTIOC6A	I(s)/ O	MTIOC2A	I(s)/ O	SD0_CD	I(s)	IRQ3	I(s)	-	-	(1)
D2	PK_3	I(s)/ O	-	-	ET1_RXCLK	I(s)	NFDATA6	I(s)/ O	CC2_Rd0	I(s)	CAN0RX_ DATARATE_ EN	O	MOSI0	I(s)/ O	-	-	REF50CK1	I(s)	(1)
D3	PK_2	I(s)/ O	-	-	ET1_TXD1	O	NFDATA5	I(s)/ O	VBUSEN1	O	CAN0RX	I(s)	RSPCK0	I(s)/ O	IRQ5	I(s)	RMII1_TXD1	O	(1)
D4	Vcc																		
D5	QSPI1_IO0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
D6	Vss																		
D7	PVcc_SPI																		
D8	PK_0	I(s)/ O	-	-	ET1_TXEN	O	NFDATA3	I(s)/ O	CC1_Rd0	I(s)	MTIOC1B	I(s)/ O	SSIBCK2	I(s)/ O	-	-	RMII1_ TXD_EN	O	(1)
D9	P6_2	I(s)/ O	-	-	ET0_TXD0	O	VIO_VD	I(s)	RxD3	I(s)	MTIOC2B	I(s)/ O	OTG_ EXICEN1	O	IRQ0	I(s)	RMII0_TXD0	O	(1)
D10	PH_0	I(s)/ O	-	-	AUDIO_CLK	I(s)	VIO_D1	I(s)	GTIOC4A	I(s)/ O	MTIOC1A	I(s)/ O	CC1_Rd0	I(s)	IRQ3	I(s)	-	-	(1)
D11	PVcc																		
D12	Vss																		
D13	PF_0	I(s)/ O	-	-	SCK3	I(s)/ O	DV0_DATA15	I(s)	LCD0_DATA8	O	MTIOC7A	I(s)/ O	RSPCK1	I(s)/ O	-	-	-	-	(1)
D14	PA_1	I(s)/ O	-	-	-	-	DV0_DATA12	I(s)	LCD0_ DATA11	O	SCI_RXD1	I(s)/ O	MTIOC8B	I(s)/ O	IRQ6	I(s)	-	-	(1)
D15	PA_7	I(s)/ O	-	-	-	-	DV0_DATA6	I(s)	LCD0_ DATA17	O	SSIRxD1	I(s)	POE10#	I(s)	-	-	-	-	(1)
D16	PVcc																		
D17	Vss																		
D18	PD_4	I(s)/ O(o)	-	-	RIIC2SCL	I(s)/ O(o)	IRQ4	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
D19	PD_3	I(s)/ O(o)	-	-	RIIC1SDA	I(s)/ O(o)	IRQ3	I(s)	MTCLKD	I(s)	GTETRGD	I(s)	-	-	-	-	-	-	(4)
D20	PD_1	I(s)/ O(o)	-	-	RIIC0SDA	I(s)/ O(o)	IRQ1	I(s)	MTCLKB	I(s)	GTETRGB	I(s)	-	-	-	-	-	-	(4)
E1	Vss																		
E2	BSCANP	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(12)
E3	PF_7	I(s)/ O	-	-	GTETRGD	I(s)	DV0_DATA23	I(s)	LCD0_DATA0	O	MTCLKD	I(s)	IRQ1	I(s)	-	-	-	-	(1)
E4	PVcc																		

Table 1.5 List of Pins (272-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
E17	PVcc																		
E18	PD_2	I(s)/ O(o)	-	-	RIIC1SCL	I(s)/ O(o)	IRQ2	I(s)	MTCLKC	I(s)	GTETRGC	I(s)	-	-	-	-	-	-	(4)
E19	TCK/ SWDCLK	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(7)
E20	JP0_0	I	-	-	TDI	I	-	-	-	-	-	-	-	-	-	-	-	-	(6)
F1	HM_RWDS/ OM_DQS	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
F2	HM_CK#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
F3	HM_CK/ OM_SCLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
F4	Vss																		
F17	TMS/SWDIO	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(9)
F18	PD_0	I(s)/ O(o)	-	-	RIIC0SCL	I(s)/ O(o)	IRQ0	I(s)	MTCLKA	I(s)	GTETRGA	I(s)	-	-	-	-	-	-	(4)
F19	TRST#	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(10)
F20	JP0_1	O	-	-	TDO/SWO	O	-	-	-	-	-	-	-	-	-	-	-	-	(8)
G1	HM_DQ2/ OM_SIO2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G2	HM_DQ1/ OM_SIO1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G3	HM_CS0#/ OM_CS0#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G4	PVcc_HO																		
G17	Vss																		
G18	PB_1	I(s)/ O	-	-	-	-	DV0_DATA4	I(s)	LCD0_ DATA19	O	SSILRCK1	I(s)/ O	POE4#	I(s)	-	-	-	-	(1)
G19	PB_2	I(s)/ O	-	-	-	-	DV0_DATA3	I(s)	LCD0_ DATA20	O	SSIBCK1	I(s)/ O	POE0#	I(s)	-	-	-	-	(1)
G20	PB_3	I(s)/ O	-	-	-	-	DV0_DATA2	I(s)	LCD0_ DATA21	O	SSIDATA2	I(s)/ O	CTS0#	I(s)/ O	-	-	-	-	(1)
H1	HM_DQ3/ OM_SIO3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H2	HM_DQ6/ OM_SIO6	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H3	HM_DQ4/ OM_SIO4	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H4	HM_CS1#/ OM_CS1#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H17	PVcc																		
H18	PB_5	I(s)/ O	-	-	-	-	DV0_DATA0	I(s)	LCD0_ DATA23	O	SSIBCK2	I(s)/ O	TxD0	O	-	-	-	-	(1)
H19	PB_4	I(s)/ O	-	-	-	-	DV0_DATA1	I(s)	LCD0_ DATA22	O	SSILRCK2	I(s)/ O	RTS0#	I(s)/ O	-	-	-	-	(1)
H20	P7_7	I(s)/ O	-	-	-	-	DV0_HSYNC	I(s)	LCD0_ TCON0	O	GTIOC3B	I(s)/ O	RxD0	I(s)	-	-	-	-	(1)
J1	HM_RESET#/ OM_RESET#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J2	HM_DQ7/ OM_SIO7	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J3	HM_DQ5/ OM_SIO5	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J4	HM_DQ0/ OM_SIO0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J9	Vcc																		
J10	Vss																		
J11	Vss																		
J12	Vcc																		
J17	P7_6	I(s)/ O	-	-	-	-	DV0_VSYNC	I(s)	LCD0_TCON 1	O	GTIOC3A	I(s)/ O	SCK0	I(s)/ O	-	-	-	-	(1)
J18	PG_2	I(s)/ O	-	-	ET0_TXD3	O	VIO_D10	I(s)	MISO0	I(s)/ O	MTIOC3B	I(s)/ O	GTIOC0A	I(s)/ O	IRQ4	I(s)	-	-	(1)
J19	PG_1	I(s)/ O	-	-	ET0_TXD2	O	VIO_D9	I(s)	MOSI0	I(s)/ O	MTIOC3C	I(s)/ O	HM_INT#/ OM_ECS#	I(s)	-	-	-	-	(1)
J20	P7_2	I(s)/ O	-	-	-	-	DV0_CLK	I(s)	LCD0_TCON 2	O	TEND0	O	CC2_Ra0	I(s)	-	-	-	-	(1)

Table 1.5 List of Pins (272-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
K1	PJ_7	I(s)/O	-	-	GTETRGB	I(s)	NFDATA0	I(s)/O	LCD0_EXTCLK	I(s)	MTCLKB	I(s)	-	-	-	-	-	-	(1)
K2	PJ_6	I(s)/O	-	-	GTETRGC	I(s)	NFCE#	O	LCD0_CLK	O	MTCLKC	I(s)	IRQ0	I(s)	-	-	-	-	(1)
K3	PH_4	I(s)/O	-	-	HM_INT#/OM_ECS#	I(s)	CTS2#	I(s)/O	GTIOC6B	I(s)/O	MTIOC2B	I(s)/O	SD0_WP	I(s)	IRQ2	I(s)	-	-	(1)
K4	PJ_0	I(s)/O	-	-	TRACECLK	O	SPDIF_OUT	O	-	-	SCK1	I(s)/O	SSIRxD3	I(s)	-	-	-	-	(1)
K9	Vcc																		
K10	Vss																		
K11	Vss																		
K12	Vcc																		
K17	PVcc																		
K18	PG_3	I(s)/O	-	-	ET0_COL	I(s)	VIO_D11	I(s)	SSL00	I(s)/O	MTIOC3D	I(s)/O	GTIOC0B	I(s)/O	-	-	-	-	(1)
K19	P6_7	I(s)/O	-	-	-	-	DRP03*	I(s)/O	LCD0_TCON3	O	DACK0	O	CC2_Rd0	I(s)	-	-	-	-	(1)
K20	PVcc																		
L1	PJ_1	I(s)/O	-	-	TRACECTL	O	SPDIF_IN	I(s)	-	-	RxD1	I(s)	VBUSIN0	I(s)	IRQ0	I(s)	-	-	(1)
L2	PJ_3	I(s)/O	-	-	TRACEDATA1	O	NFDATA0	I(s)/O	-	-	RTS1#	I(s)/O	SSILRCK3	I(s)/O	-	-	-	-	(1)
L3	PH_5	I(s)/O	-	-	HM_RSTO#	I(s)	NFDATA2	I(s)/O	ET1_EXOUT/ET1_SCLKIN	I(s)/O	MTIC5U	I(s)	IRQ5	I(s)	-	-	-	-	(1)
L4	PVcc																		
L9	Vcc																		
L10	Vss																		
L11	Vss																		
L12	Vcc																		
L17	Vss																		
L18	P6_0	I(s)/O	-	-	ADTRG#	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	(1)
L19	P6_5	I(s)/O	-	-	-	-	DRP01*	I(s)/O	LCD0_TCON5	O	AUDIO_XOUT	O	CC1_Rd0	I(s)	-	-	-	-	(1)
L20	Vss																		
M1	PH_6	I(s)/O	-	-	HM_INT#/OM_ECS#	I(s)	NFDATA3	I(s)/O	ET1_WOL	O	MTIC5V	I(s)	IRQ4	I(s)	-	-	-	-	(1)
M2	PJ_2	I(s)/O	-	-	TRACEDATA0	O	NFCE#	O	-	-	TxD1	O	SSITxD3	O	-	-	-	-	(1)
M3	PK_5	I(s)/O	-	-	GTETRGA	I(s)	NFDATA1	I(s)/O	WDTOVF#/PERROUT#	O	MTCLKA	I(s)	-	-	-	-	-	-	(1)
M4	Vss																		
M9	Vcc																		
M10	Vss																		
M11	Vss																		
M12	Vcc																		
M17	PVcc_SD0																		
M18	P6_4	I(s)/O	-	-	-	-	DRP00*	I(s)/O	LCD0_TCON6	O	AUDIO_CLK	I(s)	SD1_CD	I(s)	-	-	-	-	(1)
M19	P6_6	I(s)/O	-	-	-	-	DRP02*	I(s)/O	LCD0_TCON4	O	DREQ0	I(s)	CC1_Ra0	I(s)	-	-	-	-	(1)
M20	SD0_RST#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
N1	PVcc																		
N2	PJ_5	I(s)/O	-	-	TRACEDATA3	O	NFDATA2	I(s)/O	OVRCUR0	I(s)	MTIOC1A	I(s)/O	SSILRCK2	I(s)/O	IRQ4	I(s)	-	-	(1)
N3	PJ_4	I(s)/O	-	-	TRACEDATA2	O	NFDATA1	I(s)/O	-	-	CTS1#	I(s)/O	SSIBCK3	I(s)/O	-	-	-	-	(1)
N4	PVcc																		
N17	SD0_DAT0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
N18	SD0_DAT5	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
N19	SD0_DAT7	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
N20	SD0_DAT6	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
P1	AUDIO_X1	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)

Table 1.5 List of Pins (272-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
P2	AUDIO_X2	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
P3	P3_5	I(s)/ O	-	-	ET1_RXD1	I(s)	NFCLE	O	CC2_Ra0	I(s)	CAN0TX_ DATARATE_ EN	O	SSL00	I(s)/ O	-	-	RMII1_RXD1	I(s)	(1)
P4	PK_4	I(s)/ O	-	-	ET1_RXD0	I(s)	NFDATA7	I(s)/ O	OVRCUR1	I(s)	CAN0TX	O	MISO0	I(s)/ O	IRQ6	I(s)	RMII1_RXD0	I(s)	(1)
P17	PVcc_SD1																		
P18	SD0_DAT2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
P19	SD0_DAT3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
P20	SD0_DAT4	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
R1	Vss																		
R2	P3_1	I(s)/ O	-	-	ET1_RXER	I(s)	NFALE	O	VBUSEN0	I(s)	CAN1RX	I(s)	RSPCK2	I(s)/ O	IRQ6	I(s)	RMII1_RXER	I(s)	(1)
R3	PG_4	I(s)/ O	-	-	ET0_TXER	O	VIO_D15	I(s)	RSPCK1	I(s)/ O	MTIOC4A	I(s)/ O	GTIOC1A	I(s)/ O	-	-	-	-	(1)
R4	MIPIAVcc18																		
R17	Vss																		
R18	SD0_CMD	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
R19	SD0_DAT1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
R20	SD0_CLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
T1	CSI_CLKP	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
T2	CSI_CLKN	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
T3	P3_2	I(s)/ O	-	-	ET1_CRS	I(s)	NFRE#	O	CC1_Ra1	I(s)	CAN1RX_ DATARATE_ EN	O	MOSI2	I(s)/ O	-	-	RMII1_ CRS_DV	I(s)	(1)
T4	Vss																		
T17	AVss																		
T18	SD1_DAT0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
T19	SD1_CLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
T20	Vss																		
U1	CSI_DATA0P	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
U2	CSI_DATA0N	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
U3	Vss																		
U4	PG_6	I(s)/ O	-	-	ET0_RXD2	I(s)	VIO_D13	I(s)	MISO1	I(s)/ O	MTIOC4C	I(s)/ O	GTIOC2A	I(s)/ O	IRQ5	I(s)	-	-	(1)
U5	PC_0	I(s)/ O	-	-	VBUSIN1	I(s)	NFDATA5	I(s)/ O	ET1_TXCLK	I(s)	RSPCK2	I(s)/ O	IRQ2	I(s)	-	-	-	-	(1)
U6	P3_3	I(s)/ O	-	-	ET1_MDC	O	NFWE#	O	OTG_EXICE N0	O	CAN1TX	O	MISO2	I(s)/ O	IRQ7	I(s)	-	-	(1)
U7	P4_6	I(s)/ O	-	-	ET0_EXOUT/ ET0_SCLKIN	I(s)/ O	TXCLKOUTP	O	SCI_TXD0	I(s)/ O	TxD4	O	DACK0	O	-	-	-	-	(1)
U8	LVDSPLLVcc																		
U9	LVDSAPVcc																		
U10	USBDPVcc0																		
U11	RREF0		-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
U12	RREF1		-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
U13	Vss																		
U14	PLLVcc																		
U15	PVcc																		
U16	PC_6	I(s)/ O	-	-	VBUSEN0	O	NFWE#	O	ET1_RXD2	I(s)	SD1_CD	I(s)	LCD0_ TCON1	O	IRQ7	I(s)	-	-	(1)
U17	AVcc																		
U18	SD1_CMD	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
U19	SD1_DAT1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
U20	SD1_DAT2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
V1	CSI_DATA1P	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
V2	CSI_DATA1N	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
V3	Vss																		
V4	P3_4	I(s)/ O	-	-	ET1_MDIO	I(s)/ O	NFRB#	I(s)	CC2_Ra1	I(s)	CAN1TX_ DATARATE_ EN	O	SSL20	I(s)/ O	-	-	-	-	(1)

Table 1.5 List of Pins (272-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
V5	PC_2	I(s)/O	-	-	OTG_EXICEN0	O	NFDATA7	I(s)/O	ET1_TXD3	O	MISO2	I(s)/O	LCD0_TCON5	O	-	-	-	-	(1)
V6	P4_2	I(s)/O	-	-	TxD0	O	TXOUT1P	O	SCI_TXD1	I(s)/O	SSITxD1	O	MTIOC8C	I(s)/O	IRQ2	I(s)	-	-	(1)
V7	P4_7	I(s)/O	-	-	ET0_WOL	O	TXCLKOUTM	O	SCI_SCK0	I(s)/O	SCK4	I(s)/O	TEND0	O	-	-	-	-	(1)
V8	Vss																		
V9	Vss																		
V10	USBVss																		
V11	USBVss																		
V12	USBVss																		
V13	Vss																		
V14	PC_3	I(s)/O	-	-	OTG_ID0	I(s)	NFCLE	O	ET1_COL	I(s)	SSL20	I(s)/O	LCD0_TCON4	O	-	-	-	-	(1)
V15	PC_5	I(s)/O	-	-	VBUSEN1	O	NFRE#	O	ET1_RXDV	I(s)	SPDIF_OUT	O	LCD0_TCON2	O	IRQ0	I(s)	-	-	(1)
V16	PC_7	I(s)/O	-	-	OVRCUR0	I(s)	NFRB#	I(s)	ET1_RXD3	I(s)	SD1_WP	I(s)	LCD0_TCON0	O	IRQ6	I(s)	-	-	(1)
V17	P5_0	I(s)	-	-	AN000	I(a)	IRQ4	I(s)	SD0_CD	I(s)	SD1_CD	I(s)	-	-	-	-	-	-	(3)
V18	P5_6	I(s)	-	-	AN006	I(a)	IRQ2	I(s)	-	-	-	-	-	-	-	-	-	-	(3)
V19	P5_7	I(s)	-	-	AN007	I(a)	IRQ3	I(s)	-	-	-	-	-	-	-	-	-	-	(3)
V20	SD1_DAT3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
W1	MIPIAVcc18																		
W2	Vss																		
W3	PG_5	I(s)/O	-	-	ET0_RXDV	I(s)	VIO_D14	I(s)	MOSI1	I(s)/O	MTIOC4B	I(s)/O	GTIOC1B	I(s)/O	-	-	-	-	(1)
W4	P3_0	I(s)/O	-	-	OTG_EXICEN1	O	NFDATA4	I(s)/O	ET1_LINKSTA	I(s)	MTIC5W	I(s)	IRQ3	I(s)	-	-	-	-	(1)
W5	P4_1	I(s)/O	-	-	RxD0	I(s)	TXOUT0M	O	SCI_RXD1	I(s)/O	SSIRxD1	I(s)	MTIOC8B	I(s)/O	IRQ1	I(s)	-	-	(1)
W6	P4_3	I(s)/O	-	-	RTS0#	I(s)/O	TXOUT1M	O	SCI_CTS1#/RTS1#	I(s)/O	SSILRCK1	I(s)/O	MTIOC8D	I(s)/O	IRQ3	I(s)	-	-	(1)
W7	NMI	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(10)
W8	USB_X2	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
W9	USBVss																		
W10	DP0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
W11	USBAPVcc0																		
W12	USBAPVcc1																		
W13	DM1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
W14	USBDPVcc1																		
W15	PC_4	I(s)/O	-	-	OTG_ID1	I(s)	NFALE	O	ET1_TXER	O	SPDIF_IN	I(s)	LCD0_TCON3	O	IRQ1	I(s)	-	-	(1)
W16	XTAL	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
W17	RTC_X2	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(14)
W18	P5_4	I(s)	-	-	AN004	I(a)	IRQ0	I(s)	SD1_CD	I(s)	-	-	-	-	-	-	-	-	(3)
W19	P5_3	I(s)	-	-	AN003	I(a)	IRQ7	I(s)	OTG_ID0	I(s)	-	-	-	-	-	-	-	-	(3)
W20	P5_5	I(s)	-	-	AN005	I(a)	IRQ1	I(s)	SD1_WP	I(s)	-	-	-	-	-	-	-	-	(3)
Y1	Vss																		
Y2	PG_7	I(s)/O	-	-	ET0_RXD3	I(s)	VIO_D12	I(s)	SSL10	I(s)/O	MTIOC4D	I(s)/O	GTIOC2B	I(s)/O	-	-	-	-	(1)
Y3	PC_1	I(s)/O	-	-	VBUSIN0	I(s)	NFDATA6	I(s)/O	ET1_TXD2	O	MOSI2	I(s)/O	LCD0_TCON6	O	-	-	-	-	(1)
Y4	P4_0	I(s)/O	-	-	SCK0	I(s)/O	TXOUT0P	O	SCI_SCK1	I(s)/O	SSIBCK1	I(s)/O	MTIOC8A	I(s)/O	IRQ0	I(s)	-	-	(1)
Y5	P4_4	I(s)/O	-	-	CTS0#	I(s)/O	TXOUT2P	O	SCI_CTS0#/RTS0#	I(s)/O	WDTOVF#/PERROUT#	O	OTG_EXICEN0	O	-	-	-	-	(1)
Y6	P4_5	I(s)/O	-	-	ET0_LINKSTA	I(s)	TXOUT2M	O	SCI_RXD0	I(s)/O	RxD4	I(s)	DREQ0	I(s)	-	-	-	-	(1)
Y7	RES#	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(12)
Y8	USB_X1	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
Y9	USBVss																		
Y10	DM0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	

Table 1.5 List of Pins (272-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
Y11	USBVss																		
Y12	USBVss																		
Y13	DP1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
Y14	USBVss																		
Y15	EXTAL	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
Y16	Vss																		
Y17	RTC_X1	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(14)
Y18	P5_2	I(s)	-	-	AN002	I(a)	IRQ6	I(s)	VBUSIN0	I(s)	-	-	-	-	-	-	-	-	(3)
Y19	P5_1	I(s)	-	-	AN001	I(a)	IRQ5	I(s)	SD0_WP	I(s)	SD1_WP	I(s)	-	-	-	-	-	-	(3)
Y20	Vss																		

[Legend]
(s): Schmitt
(a): Analog
(o): Open drain

Note: * Only in products with a DRP

Table 1.6 List of Pins (256-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
A1	Vcc																		
A2	QSPI1_SSL	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A3	QSPI1_IO3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A4	QSPI1_ SPCLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A5	RPC_ RESET#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A6	PVcc_SPI																		
A7	Vss																		
A8	QSPI0_ SPCLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A9	P6_1	I(s)/ O	-	-	ET0_TXEN	O	VIO_CLK	I(s)	SCK3	I(s)/ O	MTIOC2A	I(s)/ O	-	-	-	-	RMII0_ TXD_EN	O	(1)
A10	P6_3	I(s)/ O	-	-	ET0_TXD1	O	VIO_HD	I(s)	TxD3	O	POE0#	I(s)	-	-	-	-	RMII0_TXD1	O	(1)
A11	PE_6	I(s)/ O	-	-	ET0_MDIO	I(s)/ O	VIO_D2	I(s)	SSIRxD0	I(s)	MTIOC0D	I(s)/ O	CC2_Rd1	I(s)	-	-	-	-	(1)
A12	PL_4	I(s)	MD_BOOT0	I(s)	-	-	-	-	-	-	-	-	IRQ0	I(s)	-	-	-	-	(5)
A13	PL_0	I(s)	MD_CLKS	I(s)	-	-	-	-	-	-	-	-	IRQ4	I(s)	-	-	-	-	(5)
A14	PE_5	I(s)/ O	-	-	ET0_MDC	O	VIO_D3	I(s)	SSITxD0	O	MTIOC0C	I(s)/ O	CC1_Rd1	I(s)	-	-	-	-	(1)
A15	PE_4	I(s)/ O	-	-	ET0_CRS	I(s)	VIO_D4	I(s)	SSILRCK0	I(s)/ O	MTIOC0B	I(s)/ O	-	-	-	-	RMII0_ CRS_DV	I(s)	(1)
A16	PE_3	I(s)/ O	-	-	ET0_RXER	I(s)	VIO_D5	I(s)	SSIBCK0	I(s)/ O	MTIOC0A	I(s)/ O	-	-	-	-	RMII0_RXER	I(s)	(1)
A17	PA_2	I(s)/ O	-	-	-	-	DV0_DATA11	I(s)	LCD0_ DATA12	O	SCI_SCK1	I(s)/ O	MTIOC8A	I(s)/ O	-	-	-	-	(1)
A18	PE_1	I(s)/ O	-	-	ET0_RXD0	I(s)	VIO_D7	I(s)	RxD2	I(s)	POE8#	I(s)	VBUSIN1	I(s)	IRQ1	I(s)	RMII0_RXD0	I(s)	(1)
A19	PA_6	I(s)/ O	-	-	-	-	DV0_DATA7	I(s)	LCD0_ DATA16	O	SCI_SCK0	I(s)/ O	MTIOC0A	I(s)/ O	-	-	-	-	(1)
A20	CKIO	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(11)
A21	Vss																		
B1	PK_1	I(s)/ O	-	-	ET1_TXD0	O	NFDATA4	I(s)/ O	CC1_Ra0	I(s)	CAN_CLK	I(s)	SSIDATA2	I(s)/ O	-	-	RMII1_TXD0	O	(1)
B2	Vcc																		
B3	QSPI1_IO0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B4	QSPI1_IO1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B5	RPC_INT#	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B6	QSPI0_SSL	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B7	QSPI0_IO3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B8	QSPI0_IO1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B9	PF_5	I(s)/ O	-	-	TxD2	O	DV0_DATA20	I(s)	LCD0_DATA3	O	MTIOC6B	I(s)/ O	SSILRCK0	I(s)/ O	-	-	-	-	(1)
B10	PF_4	I(s)/ O	-	-	RxD2	I(s)	DV0_DATA19	I(s)	LCD0_DATA4	O	MTIOC6A	I(s)/ O	SSIBCK0	I(s)/ O	IRQ1	I(s)	-	-	(1)
B11	PH_1	I(s)/ O	-	-	AUDIO_XOU T	O	VIO_D0	I(s)	GTIOC4B	I(s)/ O	MTIOC1B	I(s)/ O	CC2_Rd0	I(s)	IRQ2	I(s)	-	-	(1)
B12	PL_1	I(s)	MD_CLK	I(s)	-	-	-	-	-	-	-	-	IRQ5	I(s)	-	-	-	-	(5)
B13	PL_2	I(s)	MD_BOOT2	I(s)	-	-	-	-	-	-	-	-	IRQ6	I(s)	-	-	-	-	(5)
B14	P8_0	I(s)/ O	-	-	-	-	DV0_DATA14	I(s)	LCD0_DATA9	O	SCI_CTS1#/ RTS1#	I(s)/ O	MTIOC8D	I(s)/ O	-	-	-	-	(1)
B15	PF_1	I(s)/ O	-	-	RxD3	I(s)	DV0_DATA16	I(s)	LCD0_DATA7	O	MTIOC7B	I(s)/ O	MOSI1	I(s)/ O	IRQ4	I(s)	-	-	(1)
B16	PA_0	I(s)/ O	-	-	-	-	DV0_DATA13	I(s)	LCD0_ DATA10	O	SCI_TXD1	I(s)/ O	MTIOC8C	I(s)/ O	-	-	-	-	(1)
B17	PA_7	I(s)/ O	-	-	-	-	DV0_DATA6	I(s)	LCD0_ DATA17	O	SSIRxD1	I(s)	POE10#	I(s)	-	-	-	-	(1)
B18	PA_4	I(s)/ O	-	-	-	-	DV0_DATA9	I(s)	LCD0_ DATA14	O	SCI_TXD0	I(s)/ O	MTIOC0C	I(s)/ O	-	-	-	-	(1)
B19	PB_0	I(s)/ O	-	-	-	-	DV0_DATA5	I(s)	LCD0_ DATA18	O	SSITxD1	O	POE8#	I(s)	-	-	-	-	(1)
B20	Vss																		
B21	PD_3	I(s)/ O(o)	-	-	RIIC1SDA	I(s)/ O(o)	IRQ3	I(s)	MTCLKD	I(s)	GTETRGD	I(s)	-	-	-	-	-	-	(4)

Table 1.6 List of Pins (256-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
C1	PK_3	I(s)/ O	-	-	ET1_RXCLK	I(s)	NFDATA6	I(s)/ O	CC2_Rd0	I(s)	CAN0RX_ DATARATE_ EN	O	MOSI0	I(s)/ O	-	-	REF50CK1	I(s)	(1)
C2	PH_2	I(s)/ O	-	-	CTS2#	I(s)/ O	DV0_DATA22	I(s)	LCD0_DATA1	O	MTIOC6D	I(s)/ O	SSIRxD0	I(s)	-	-	-	-	(1)
C20	PD_7	I(s)/ O(o)	-	-	RIIC3SDA	I(s)/ O(o)	IRQ7	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
C21	PD_1	I(s)/ O(o)	-	-	RIIC0SDA	I(s)/ O(o)	IRQ1	I(s)	MTCLKB	I(s)	GTETRGB	I(s)	-	-	-	-	-	-	(4)
D1	HM_CK/ OM_SCLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
D2	BSCANP	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(12)
D4	Vcc																		
D5	QSPI1_IO2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
D6	RPC_WP#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
D7	QSPI0_IO2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
D8	PK_0	I(s)/ O	-	-	ET1_TXEN	O	NFDATA3	I(s)/ O	CC1_Rd0	I(s)	MTIOC1B	I(s)/ O	SSIBCK2	I(s)/ O	-	-	RMII1_ TXD_EN	O	(1)
D9	P6_2	I(s)/ O	-	-	ET0_TXD0	O	VIO_VD	I(s)	RxD3	I(s)	MTIOC2B	I(s)/ O	OTG_ EXICEN1	O	IRQ0	I(s)	RMII0_TXD0	O	(1)
D10	PH_0	I(s)/ O	-	-	AUDIO_CLK	I(s)	VIO_D1	I(s)	GTIOC4A	I(s)/ O	MTIOC1A	I(s)/ O	CC1_Rd0	I(s)	IRQ3	I(s)	-	-	(1)
D11	PF_3	I(s)/ O	-	-	SCK2	I(s)/ O	DV0_DATA18	I(s)	LCD0_DATA5	O	MTIOC7D	I(s)/ O	SSL10	I(s)/ O	-	-	-	-	(1)
D12	PF_2	I(s)/ O	-	-	TxD3	O	DV0_DATA17	I(s)	LCD0_DATA6	O	MTIOC7C	I(s)/ O	MISO1	I(s)/ O	-	-	-	-	(1)
D13	PF_0	I(s)/ O	-	-	SCK3	I(s)/ O	DV0_DATA15	I(s)	LCD0_DATA8	O	MTIOC7A	I(s)/ O	RSPCK1	I(s)/ O	-	-	-	-	(1)
D14	PE_2	I(s)/ O	-	-	ET0_RXD1	I(s)	VIO_D6	I(s)	TxD2	O	POE10#	I(s)	-	-	-	-	RMII0_RXD1	I(s)	(1)
D15	PA_1	I(s)/ O	-	-	-	-	DV0_DATA12	I(s)	LCD0_ DATA11	O	SCI_RXD1	I(s)/ O	MTIOC8B	I(s)/ O	IRQ6	I(s)	-	-	(1)
D16	PG_0	I(s)/ O	-	-	ET0_TXCLK	I(s)	VIO_D8	I(s)	RSPCK0	I(s)/ O	MTIOC3A	I(s)/ O	HM_RSTO#	I(s)	-	-	-	-	(1)
D17	PA_5	I(s)/ O	-	-	-	-	DV0_DATA8	I(s)	LCD0_ DATA15	O	SCI_RXD0	I(s)/ O	MTIOC0B	I(s)/ O	IRQ5	I(s)	-	-	(1)
D18	Vss																		
D20	PD_5	I(s)/ O(o)	-	-	RIIC2SDA	I(s)/ O(o)	IRQ5	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
D21	JP0_0	I	-	-	TDI	I	-	-	-	-	-	-	-	-	-	-	-	-	(6)
E1	HM_CK#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
E2	HM_CS0#/ OM_CS0#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
E4	PK_2	I(s)/ O	-	-	ET1_TXD1	O	NFDATA5	I(s)/ O	VBUSEN1	O	CAN0RX	I(s)	RSPCK0	I(s)/ O	IRQ5	I(s)	RMII1_TXD1	O	(1)
E5	Vcc																		
E6	Vss																		
E7	PVcc																		
E8	QSPI0_IO0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
E9	PF_6	I(s)/ O	-	-	RTS2#	I(s)/ O	DV0_DATA21	I(s)	LCD0_DATA2	O	MTIOC6C	I(s)/ O	SSITxD0	O	-	-	-	-	(1)
E10	PE_0	I(s)/ O	-	-	ET0_RXCLK	I(s)	VIO_FLD	I(s)	SCK2	I(s)/ O	POE4#	I(s)	-	-	-	-	REF50CK0	I(s)	(1)
E11	PL_3	I(s)	MD_BOOT1	I(s)	-	-	-	-	-	-	-	-	IRQ7	I(s)	-	-	-	-	(5)
E12	Vcc																		
E13	Vss																		
E14	PVcc																		
E15	PA_3	I(s)/ O	-	-	-	-	DV0_DATA10	I(s)	LCD0_ DATA13	O	SCI_CTS0#/ RTS0#	I(s)/ O	MTIOC0D	I(s)/ O	-	-	-	-	(1)
E16	Vcc																		
E17	Vss																		
E18	TCK/ SWDCLK	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(7)
E20	PD_0	I(s)/ O(o)	-	-	RIIC0SCL	I(s)/ O(o)	IRQ0	I(s)	MTCLKA	I(s)	GTETRGA	I(s)	-	-	-	-	-	-	(4)
E21	TRST#	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(10)

Table 1.6 List of Pins (256-Pin BGA)

Ball Number	Port Function/ Dedicated Function	I/O	Mode Function	I/O	Function 1	I/O	Function 2	I/O	Function 3	I/O	Function 4	I/O	Function 5	I/O	Function 6	I/O	Function 7	I/O	Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol		Symbol		Symbol		Symbol		Symbol		Symbol		Symbol		Symbol		Symbol		
F1	HM_DQ1/ OM_SIO1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
F2	HM_RWDS/ OM_DQS	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
F4	PF_7	I(s)/ O	-	-	GTETRGD	I(s)	DV0_DATA23	I(s)	LCD0_DATA0	O	MTCLKD	I(s)	IRQ1	I(s)	-	-	-	-	(1)
F5	Vss																		
F17	PD_6	I(s)/ O(o)	-	-	RIIC3SCL	I(s)/ O(o)	IRQ6	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
F18	PD_2	I(s)/ O(o)	-	-	RIIC1SCL	I(s)/ O(o)	IRQ2	I(s)	MTCLKC	I(s)	GTETRGC	I(s)	-	-	-	-	-	-	(4)
F20	JP0_1	O	-	-	TDO/SWO	O	-	-	-	-	-	-	-	-	-	-	-	-	(8)
F21	PB_1	I(s)/ O	-	-	-	-	DV0_DATA4	I(s)	LCD0_ DATA19	O	SSILRCK1	I(s)/ O	POE4#	I(s)	-	-	-	-	(1)
G1	PVcc_HO																		
G2	HM_DQ2/ OM_SIO2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G4	HM_DQ4/ OM_SIO4	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G5	PVcc																		
G17	PD_4	I(s)/ O(o)	-	-	RIIC2SCL	I(s)/ O(o)	IRQ4	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
G18	TMS/SWDIO	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(9)
G20	PB_3	I(s)/ O	-	-	-	-	DV0_DATA2	I(s)	LCD0_ DATA21	O	SSIDATA2	I(s)/ O	CTS0#	I(s)/ O	-	-	-	-	(1)
G21	PB_5	I(s)/ O	-	-	-	-	DV0_DATA0	I(s)	LCD0_ DATA23	O	SSIBCK2	I(s)/ O	TxD0	O	-	-	-	-	(1)
H1	Vss																		
H2	HM_DQ6/ OM_SIO6	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H4	HM_CS1#/ OM_CS1#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H5	PH_3	I(s)/ O	-	-	HM_RSTO#	I(s)	RTS2#	I(s)/ O	GTIOC6A	I(s)/ O	MTIOC2A	I(s)/ O	SD0_CD	I(s)	IRQ3	I(s)	-	-	(1)
H17	PB_2	I(s)/ O	-	-	-	-	DV0_DATA3	I(s)	LCD0_ DATA20	O	SSIBCK1	I(s)/ O	POE0#	I(s)	-	-	-	-	(1)
H18	PB_4	I(s)/ O	-	-	-	-	DV0_DATA1	I(s)	LCD0_ DATA22	O	SSILRCK2	I(s)/ O	RTS0#	I(s)/ O	-	-	-	-	(1)
H20	P7_7	I(s)/ O	-	-	-	-	DV0_HSYNC	I(s)	LCD0_TCON 0	O	GTIOC3B	I(s)/ O	RxD0	I(s)	-	-	-	-	(1)
H21	PG_2	I(s)/ O	-	-	ET0_TXD3	O	VIO_D10	I(s)	MISO0	I(s)/ O	MTIOC3B	I(s)/ O	GTIOC0A	I(s)/ O	IRQ4	I(s)	-	-	(1)
J1	HM_RESET#/ OM_RESET#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J2	PH_4	I(s)/ O	-	-	HM_INT#/ OM_ECS#	I(s)	CTS2#	I(s)/ O	GTIOC6B	I(s)/ O	MTIOC2B	I(s)/ O	SD0_WP	I(s)	IRQ2	I(s)	-	-	(1)
J4	HM_DQ3/ OM_SIO3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J5	HM_DQ0/ OM_SIO0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J17	PG_1	I(s)/ O	-	-	ET0_TXD2	O	VIO_D9	I(s)	MOSI0	I(s)/ O	MTIOC3C	I(s)/ O	HM_INT#/ OM_ECS#	I(s)	-	-	-	-	(1)
J18	P7_6	I(s)/ O	-	-	-	-	DV0_VSYNC	I(s)	LCD0_TCON 1	O	GTIOC3A	I(s)/ O	SCK0	I(s)/ O	-	-	-	-	(1)
J20	P7_2	I(s)/ O	-	-	-	-	DV0_CLK	I(s)	LCD0_TCON 2	O	TEND0	O	CC2_Ra0	I(s)	-	-	-	-	(1)
J21	P6_0	I(s)/ O	-	-	ADTRG#	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	(1)
K1	PJ_6	I(s)/ O	-	-	GTETRGC	I(s)	NFCE#	O	LCD0_CLK	O	MTCLKC	I(s)	IRQ0	I(s)	-	-	-	-	(1)
K2	PJ_1	I(s)/ O	-	-	TRACECTL	O	SPDIF_IN	I(s)	-	-	RxD1	I(s)	VBUSIN0	I(s)	IRQ0	I(s)	-	-	(1)
K4	HM_DQ7/ OM_SIO7	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
K5	HM_DQ5/ OM_SIO5	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
K17	PVcc																		
K18	PG_3	I(s)/ O	-	-	ET0_COL	I(s)	VIO_D11	I(s)	SSL00	I(s)/ O	MTIOC3D	I(s)/ O	GTIOC0B	I(s)/ O	-	-	-	-	(1)

Table 1.6 List of Pins (256-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
K20	P6_5	I(s)/O	-	-	-	-	DRP01*	I(s)/O	LCD0_TCON5	O	AUDIO_XOUT	O	CC1_Rd0	I(s)	-	-	-	-	(1)
K21	P6_4	I(s)/O	-	-	-	-	DRP00*	I(s)/O	LCD0_TCON6	O	AUDIO_CLK	I(s)	SD1_CD	I(s)	-	-	-	-	(1)
L1	PH_5	I(s)/O	-	-	HM_RSTO#	I(s)	NFDATA2	I(s)/O	ET1_EXOUT/ ET1_SCLKIN	I(s)/O	MTIC5U	I(s)	IRQ5	I(s)	-	-	-	-	(1)
L2	PJ_3	I(s)/O	-	-	TRACE DATA1	O	NFDATA0	I(s)/O	-	-	RTS1#	I(s)/O	SSILRCK3	I(s)/O	-	-	-	-	(1)
L4	PJ_0	I(s)/O	-	-	TRACECLK	O	SPDIF_OUT	O	-	-	SCK1	I(s)/O	SSIRxD3	I(s)	-	-	-	-	(1)
L5	Vcc																		
L17	Vss																		
L18	P6_7	I(s)/O	-	-	-	-	DRP03*	I(s)/O	LCD0_TCON3	O	DACK0	O	CC2_Rd0	I(s)	-	-	-	-	(1)
L20	P6_6	I(s)/O	-	-	-	-	DRP02*	I(s)/O	LCD0_TCON4	O	DREQ0	I(s)	CC1_Ra0	I(s)	-	-	-	-	(1)
L21	SD0_DAT6	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
M1	PH_6	I(s)/O	-	-	HM_INT#/ OM_ECS#	I(s)	NFDATA3	I(s)/O	ET1_WOL	O	MTIC5V	I(s)	IRQ4	I(s)	-	-	-	-	(1)
M2	PJ_7	I(s)/O	-	-	GTETRGB	I(s)	NFDATA0	I(s)/O	LCD0_ EXTCLK	I(s)	MTCLKB	I(s)	-	-	-	-	-	-	(1)
M4	PJ_2	I(s)/O	-	-	TRACE DATA0	O	NFCE#	O	-	-	TxD1	O	SSITxD3	O	-	-	-	-	(1)
M5	Vss																		
M17	Vcc																		
M18	SD0_RST#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
M20	SD0_DAT7	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
M21	PVcc_SD0																		
N1	P3_5	I(s)/O	-	-	ET1_RXD1	I(s)	NFCLE	O	CC2_Ra0	I(s)	CAN0TX_ DATARATE_ EN	O	SSL00	I(s)/O	-	-	RMII1_RXD1	I(s)	(1)
N2	PJ_5	I(s)/O	-	-	TRACE DATA3	O	NFDATA2	I(s)/O	OVRCUR0	I(s)	MTIOC1A	I(s)/O	SSILRCK2	I(s)/O	IRQ4	I(s)	-	-	(1)
N4	PJ_4	I(s)/O	-	-	TRACE DATA2	O	NFDATA1	I(s)/O	-	-	CTS1#	I(s)/O	SSIBCK3	I(s)/O	-	-	-	-	(1)
N5	PK_5	I(s)/O	-	-	GTETRGA	I(s)	NFDATA1	I(s)/O	WDTOVF#/ PERROUT#	O	MTCLKA	I(s)	-	-	-	-	-	-	(1)
N17	SD0_DAT0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
N18	SD0_DAT5	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
N20	SD0_DAT4	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
N21	Vss																		
P1	AUDIO_X1	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
P2	AUDIO_X2	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
P4	PK_4	I(s)/O	-	-	ET1_RXD0	I(s)	NFDATA7	I(s)/O	OVRCUR1	I(s)	CAN0TX	O	MISO0	I(s)/O	IRQ6	I(s)	RMII1_RXD0	I(s)	(1)
P5	PVcc																		
P17	SD0_CLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
P18	SD0_DAT2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
P20	SD0_DAT3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
P21	SD0_DAT1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
R1	PG_4	I(s)/O	-	-	ET0_TXER	O	VIO_D15	I(s)	RSPCK1	I(s)/O	MTIOC4A	I(s)/O	GTIOC1A	I(s)/O	-	-	-	-	(1)
R2	P3_1	I(s)/O	-	-	ET1_RXER	I(s)	NFALE	O	VBUSEN0	I(s)	CAN1RX	I(s)	RSPCK2	I(s)/O	IRQ6	I(s)	RMII1_RXER	I(s)	(1)
R4	P3_2	I(s)/O	-	-	ET1_CRS	I(s)	NFRE#	O	CC1_Ra1	I(s)	CAN1RX_ DATARATE_ EN	O	MOSI2	I(s)/O	-	-	RMII1_ CRS_DV	I(s)	(1)
R5	Vss																		
R17	SD1_DAT2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
R18	SD1_DAT0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
R20	SD1_DAT3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
R21	SD0_CMD	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
T1	Vss																		
T2	Vss																		

Table 1.6 List of Pins (256-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
T4	MIPIAVcc18																		
T5	Vcc																		
T17	SD1_CLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
T18	SD1_CMD	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
T20	SD1_DAT1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
T21	PVcc_SD1																		
U1	CSI_DATA0P	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
U2	CSI_DATA0N	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
U4	Vss																		
U5	PVcc																		
U6	Vss																		
U7	LVDSAPVcc																		
U8	P4_3	I(s)/ O	-	-	RTS0#	I(s)/ O	TXOUT1M	O	SCI_CTS1#/ RTS1#	I(s)/ O	SSILRCK1	I(s)/ O	MTIOC8D	I(s)/ O	IRQ3	I(s)	-	-	(1)
U9	P4_7	I(s)/ O	-	-	ET0_WOL	O	TXCLKOUTM	O	SCI_SCK0	I(s)/ O	SCK4	I(s)/ O	TEND0	O	-	-	-	-	(1)
U10	USBVss																		
U11	RREF0		-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
U12	RREF1		-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
U13	USBVss																		
U14	Vss																		
U15	PLLVcc																		
U16	PVcc																		
U17	Vss																		
U18	P5_4	I(s)	-	-	AN004	I(a)	IRQ0	I(s)	SD1_CD	I(s)	-	-	-	-	-	-	-	-	(3)
U20	P5_5	I(s)	-	-	AN005	I(a)	IRQ1	I(s)	SD1_WP	I(s)	-	-	-	-	-	-	-	-	(3)
U21	Vss																		
V1	CSI_CLKP	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
V2	CSI_CLKN	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
V4	PG_6	I(s)/ O	-	-	ET0_RXD2	I(s)	VIO_D13	I(s)	MISO1	I(s)/ O	MTIOC4C	I(s)/ O	GTIOC2A	I(s)/ O	IRQ5	I(s)	-	-	(1)
V5	P3_3	I(s)/ O	-	-	ET1_MDC	O	NFWE#	O	OTG_ EXICEN0	O	CAN1TX	O	MISO2	I(s)/ O	IRQ7	I(s)	-	-	(1)
V6	PC_2	I(s)/ O	-	-	OTG_EXICE N0	O	NFDATA7	I(s)/ O	ET1_TXD3	O	MISO2	I(s)/ O	LCD0_TCON 5	O	-	-	-	-	(1)
V7	Vss																		
V8	P4_2	I(s)/ O	-	-	TxD0	O	TXOUT1P	O	SCI_TXD1	I(s)/ O	SSITxD1	O	MTIOC8C	I(s)/ O	IRQ2	I(s)	-	-	(1)
V9	P4_6	I(s)/ O	-	-	ET0_EXOUT/ ET0_SCLKIN	I(s)/ O	TXCLKOUTP	O	SCI_TXD0	I(s)/ O	TxD4	O	DACK0	O	-	-	-	-	(1)
V10	Vss																		
V11	USBAPVcc0																		
V12	USBAPVcc1																		
V13	USBVss																		
V14	PC_4	I(s)/ O	-	-	OTG_ID1	I(s)	NFALE	O	ET1_TXER	O	SPDIF_IN	I(s)	LCD0_TCON 3	O	IRQ1	I(s)	-	-	(1)
V15	PC_6	I(s)/ O	-	-	VBUSEN0	O	NFWE#	O	ET1_RXD2	I(s)	SD1_CD	I(s)	LCD0_TCON 1	O	IRQ7	I(s)	-	-	(1)
V16	PC_3	I(s)/ O	-	-	OTG_ID0	I(s)	NFCLE	O	ET1_COL	I(s)	SSL20	I(s)/ O	LCD0_TCON 4	O	-	-	-	-	(1)
V17	PVcc																		
V18	P5_0	I(s)	-	-	AN000	I(a)	IRQ4	I(s)	SD0_CD	I(s)	SD1_CD	I(s)	-	-	-	-	-	-	(3)
V20	P5_6	I(s)	-	-	AN006	I(a)	IRQ2	I(s)	-	-	-	-	-	-	-	-	-	-	(3)
V21	P5_7	I(s)	-	-	AN007	I(a)	IRQ3	I(s)	-	-	-	-	-	-	-	-	-	-	(3)
W1	CSI_DATA1P	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
W2	CSI_DATA1N	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
W20	P5_2	I(s)	-	-	AN002	I(a)	IRQ6	I(s)	VBUSIN0	I(s)	-	-	-	-	-	-	-	-	(3)
W21	P5_3	I(s)	-	-	AN003	I(a)	IRQ7	I(s)	OTG_ID0	I(s)	-	-	-	-	-	-	-	-	(3)
Y1	Vss																		

Table 1.6 List of Pins (256-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
Y2	Vss																		
Y3	PG_7	I(s)/O	-	-	ET0_RXD3	I(s)	VIO_D12	I(s)	SSL10	I(s)/O	MTIOC4D	I(s)/O	GTIOC2B	I(s)/O	-	-	-	-	(1)
Y4	PC_0	I(s)/O	-	-	VBUSIN1	I(s)	NFDATA5	I(s)/O	ET1_TXCLK	I(s)	RSPCK2	I(s)/O	IRQ2	I(s)	-	-	-	-	(1)
Y5	PC_1	I(s)/O	-	-	VBUSIN0	I(s)	NFDATA6	I(s)/O	ET1_TXD2	O	MOSI2	I(s)/O	LCD0_TCON6	O	-	-	-	-	(1)
Y6	P4_0	I(s)/O	-	-	SCK0	I(s)/O	TXOUT0P	O	SCI_SCK1	I(s)/O	SSIBCK1	I(s)/O	MTIOC8A	I(s)/O	IRQ0	I(s)	-	-	(1)
Y7	P4_4	I(s)/O	-	-	CTS0#	I(s)/O	TXOUT2P	O	SCI_CTS0#/RTS0#	I(s)/O	WDTOVF#/PERROUT#	O	OTG_EXICEN0	O	-	-	-	-	(1)
Y8	LVDSPLLVcc																		
Y9	NMI	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(10)
Y10	USB_X2	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
Y11	USBDPVcc0																		
Y12	DP0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
Y13	USBVss																		
Y14	DP1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
Y15	USBDPVcc1																		
Y16	Vss																		
Y17	XTAL	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
Y18	RTC_X2	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(14)
Y19	AVcc																		
Y20	PVcc																		
Y21	P5_1	I(s)	-	-	AN001	I(a)	IRQ5	I(s)	SD0_WP	I(s)	SD1_WP	I(s)	-	-	-	-	-	-	(3)
AA1	Vss																		
AA2	PG_5	I(s)/O	-	-	ET0_RXDV	I(s)	VIO_D14	I(s)	MOSI1	I(s)/O	MTIOC4B	I(s)/O	GTIOC1B	I(s)/O	-	-	-	-	(1)
AA3	P3_4	I(s)/O	-	-	ET1_MDIO	I(s)/O	NFRB#	I(s)	CC2_Ra1	I(s)	CAN1TX_DATARATE_EN	O	SSL20	I(s)/O	-	-	-	-	(1)
AA4	P3_0	I(s)/O	-	-	OTG_EXICEN1	O	NFDATA4	I(s)/O	ET1_LINKSTA	I(s)	MTIC5W	I(s)	IRQ3	I(s)	-	-	-	-	(1)
AA5	Vss																		
AA6	P4_1	I(s)/O	-	-	RxD0	I(s)	TXOUT0M	O	SCI_RXD1	I(s)/O	SSIRxD1	I(s)	MTIOC8B	I(s)/O	IRQ1	I(s)	-	-	(1)
AA7	P4_5	I(s)/O	-	-	ET0_LINKSTA	I(s)	TXOUT2M	O	SCI_RXD0	I(s)/O	RxD4	I(s)	DREQ0	I(s)	-	-	-	-	(1)
AA8	Vss																		
AA9	RES#	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(12)
AA10	USB_X1	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
AA11	USBVss																		
AA12	DM0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
AA13	USBVss																		
AA14	DM1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
AA15	USBVss																		
AA16	PC_5	I(s)/O	-	-	VBUSEN1	O	NFRE#	O	ET1_RXDV	I(s)	SPDIF_OUT	O	LCD0_TCON2	O	IRQ0	I(s)	-	-	(1)
AA17	PC_7	I(s)/O	-	-	OVRCUR0	I(s)	NFRB#	I(s)	ET1_RXD3	I(s)	SD1_WP	I(s)	LCD0_TCON0	O	IRQ6	I(s)	-	-	(1)
AA18	EXTAL	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
AA19	RTC_X1	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(14)
AA20	AVss																		
AA21	PVcc																		

[Legend]
(s): Schmitt
(a): Analog
(o): Open drain
Note: * Only in products with a DRP

Table 1.7 List of Pins (176-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
A1	Vss																		
A2	PK_3	I(s)/ O	-	-	-	-	NFDATA6	I(s)/ O	CC2_Rd0	I(s)	CAN0RX_ DATARATE_ EN	O	MOSI0	I(s)/ O	-	-	REF50CK1	I(s)	(1)
A3	QSPI1_ SPCLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A4	RPC_WP#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A5	PVcc_SPI																		
A6	Vss																		
A7	QSPI0_IO3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
A8	P6_1	I(s)/ O	-	-	ET0_TXEN	O	VIO_CLK	I(s)	SCK3	I(s)/ O	MTIOC2A	I(s)/ O	-	-	-	-	RMII0_ TXD_EN	O	(1)
A9	P6_2	I(s)/ O	-	-	ET0_TXD0	O	VIO_VD	I(s)	RxD3	I(s)	MTIOC2B	I(s)/ O	OTG_ EXICEN1	O	IRQ0	I(s)	RMII0_TXD0	O	(1)
A10	PH_1	I(s)/ O	-	-	AUDIO_XOUT	O	VIO_D0	I(s)	GTIOC4B	I(s)/ O	MTIOC1B	I(s)/ O	CC2_Rd0	I(s)	IRQ2	I(s)	-	-	(1)
A11	PE_6	I(s)/ O	-	-	ET0_MDIO	I(s)/ O	VIO_D2	I(s)	SSIRxD0	I(s)	MTIOC0D	I(s)/ O	CC2_Rd1	I(s)	-	-	-	-	(1)
A12	PL_2	I(s)	MD_BOOT2	I(s)	-	-	-	-	-	-	-	-	IRQ6	I(s)	-	-	-	-	(5)
A13	PE_3	I(s)/ O	-	-	ET0_RXER	I(s)	VIO_D5	I(s)	SSIBCK0	I(s)/ O	MTIOC0A	I(s)/ O	-	-	-	-	RMII0_RXER	I(s)	(1)
A14	CKIO	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(11)
A15	Vss																		
B1	BSCANP	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(12)
B2	PH_3	I(s)/ O	-	-	HM_RSTO#	I(s)	RTS2#	I(s)/ O	GTIOC6A	I(s)/ O	MTIOC2A	I(s)/ O	SD0_CD	I(s)	IRQ3	I(s)	-	-	(1)
B3	QSPI1_SSL	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B4	QSPI1_IO1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B5	RPC_RESET #	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B6	QSPI0_IO2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B7	QSPI0_IO1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
B8	PK_0	I(s)/ O	-	-	-	-	NFDATA3	I(s)/ O	CC1_Rd0	I(s)	MTIOC1B	I(s)/ O	SSIBCK2	I(s)/ O	-	-	RMII1_ TXD_EN	O	(1)
B9	P6_3	I(s)/ O	-	-	ET0_TXD1	O	VIO_HD	I(s)	TxD3	O	POE0#	I(s)	-	-	-	-	RMII0_TXD1	O	(1)
B10	PL_4	I(s)	MD_BOOT0	I(s)	-	-	-	-	-	-	-	-	IRQ0	I(s)	-	-	-	-	(5)
B11	PL_0	I(s)	MD_CLKS	I(s)	-	-	-	-	-	-	-	-	IRQ4	I(s)	-	-	-	-	(5)
B12	PE_5	I(s)/ O	-	-	ET0_MDC	O	VIO_D3	I(s)	SSITxD0	O	MTIOC0C	I(s)/ O	CC1_Rd1	I(s)	-	-	-	-	(1)
B13	PE_1	I(s)/ O	-	-	ET0_RXD0	I(s)	VIO_D7	I(s)	RxD2	I(s)	POE8#	I(s)	VBUSIN1	I(s)	IRQ1	I(s)	RMII0_RXD0	I(s)	(1)
B14	Vss																		
B15	PD_7	I(s)/ O(o)	-	-	RIIC3SDA	I(s)/ O(o)	IRQ7	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
C1	HM_CK/OM_ SCLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C2	PK_1	I(s)/ O	-	-	-	-	NFDATA4	I(s)/ O	CC1_Ra0	I(s)	CAN_CLK	I(s)	SSIDATA2	I(s)/ O	-	-	RMII1_TXD0	O	(1)
C3	QSPI1_IO3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C4	QSPI1_IO2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C5	RPC_INT#	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C6	QSPI0_SSL	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C7	QSPI0_ SPCLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
C8	PE_0	I(s)/ O	-	-	ET0_RXCLK	I(s)	VIO_FLD	I(s)	SCK2	I(s)/ O	POE4#	I(s)	-	-	-	-	REF50CK0	I(s)	(1)
C9	PH_0	I(s)/ O	-	-	AUDIO_CLK	I(s)	VIO_D1	I(s)	GTIOC4A	I(s)/ O	MTIOC1A	I(s)/ O	CC1_Rd0	I(s)	IRQ3	I(s)	-	-	(1)
C10	PL_1	I(s)	MD_CLK	I(s)	-	-	-	-	-	-	-	-	IRQ5	I(s)	-	-	-	-	(5)
C11	PE_4	I(s)/ O	-	-	ET0_CRS	I(s)	VIO_D4	I(s)	SSILRCK0	I(s)/ O	MTIOC0B	I(s)/ O	-	-	-	-	RMII0_ CRS_DV	I(s)	(1)
C12	PG_0	I(s)/ O	-	-	ET0_TXCLK	I(s)	VIO_D8	I(s)	RSPCK0	I(s)/ O	MTIOC3A	I(s)/ O	HM_RSTO#	I(s)	-	-	-	-	(1)
C13	Vss																		
C14	PD_5	I(s)/ O(o)	-	-	RIIC2SDA	I(s)/ O(o)	IRQ5	I(s)	-	-	-	-	-	-	-	-	-	-	(4)

Table 1.7 List of Pins (176-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
C15	PD_3	I(s)/ O(o)	-	-	RIIC1SDA	I(s)/ O(o)	IRQ3	I(s)	MTCLKD	I(s)	GTETRGD	I(s)	-	-	-	-	-	-	(4)
D1	HM_CK#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
D2	HM_CS0#/ OM_CS0#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
D3	PK_2	I(s)/ O	-	-	-	-	NFDATA5	I(s)/ O	VBUSEN1	O	CAN0RX	I(s)	RSPCK0	I(s)/ O	IRQ5	I(s)	RMII1_TXD1	O	(1)
D4	Vss																		
D5	Vcc																		
D6	QSPI1_IO0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
D7	QSPI0_IO0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
D8	PVcc																		
D9	PL_3	I(s)	MD_BOOT1	I(s)	-	-	-	-	-	-	-	-	IRQ7	I(s)	-	-	-	-	(5)
D10	PE_2	I(s)/ O	-	-	ET0_RXD1	I(s)	VIO_D6	I(s)	TxD2	O	POE10#	I(s)	-	-	-	-	RMII0_RXD1	I(s)	(1)
D11	PVcc																		
D12	Vss																		
D13	PD_6	I(s)/ O(o)	-	-	RIIC3SCL	I(s)/ O(o)	IRQ6	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
D14	PD_4	I(s)/ O(o)	-	-	RIIC2SCL	I(s)/ O(o)	IRQ4	I(s)	-	-	-	-	-	-	-	-	-	-	(4)
D15	PD_1	I(s)/ O(o)	-	-	RIIC0SDA	I(s)/ O(o)	IRQ1	I(s)	MTCLKB	I(s)	GTETRGB	I(s)	-	-	-	-	-	-	(4)
E1	Vss																		
E2	HM_DQ0/ OM_SIO0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
E3	HM_RWDS/ OM_DQS	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
E4	PVcc																		
E12	PD_2	I(s)/ O(o)	-	-	RIIC1SCL	I(s)/ O(o)	IRQ2	I(s)	MTCLKC	I(s)	GTETRGC	I(s)	-	-	-	-	-	-	(4)
E13	PD_0	I(s)/ O(o)	-	-	RIIC0SCL	I(s)/ O(o)	IRQ0	I(s)	MTCLKA	I(s)	GTETRGA	I(s)	-	-	-	-	-	-	(4)
E14	TCK/ SWDCLK	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(7)
E15	JP0_0	I(s)	-	-	TDI	I	-	-	-	-	-	-	-	-	-	-	-	-	(6)
F1	PVcc_HO																		
F2	HM_DQ2/ OM_SIO2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
F3	HM_DQ1/ OM_SIO1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
F4	HM_CS1#/ OM_CS1#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
F12	TMS/SWDIO	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(9)
F13	JP0_1	O	-	-	TDO/SWO	O	-	-	-	-	-	-	-	-	-	-	-	-	(8)
F14	TRST#	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(10)
F15	P6_0	I(s)/ O	-	-	ADTRG#	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	(1)
G1	HM_DQ4/ OM_SIO4	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G2	HM_DQ7/ OM_SIO7	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G3	HM_DQ5/ OM_SIO5	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G4	HM_DQ3/ OM_SIO3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
G12	Vcc																		
G13	PG_3	I(s)/ O	-	-	ET0_COL	I(s)	VIO_D11	I(s)	SSL00	I(s)/ O	MTIOC3D	I(s)/ O	GTIOC0B	I(s)/ O	-	-	-	-	(1)
G14	PG_2	I(s)/ O	-	-	ET0_TXD3	O	VIO_D10	I(s)	MISO0	I(s)/ O	MTIOC3B	I(s)/ O	GTIOC0A	I(s)/ O	IRQ4	I(s)	-	-	(1)
G15	PVcc																		
H1	HM_DQ6/ OM_SIO6	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H2	HM_RESET#/ OM_RESET#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)

Table 1.7 List of Pins (176-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
H3	PH_4	I(s)/O	-	-	HM_INT#/OM_ECS#	I(s)	CTS2#	I(s)/O	GTIOC6B	I(s)/O	MTIOC2B	I(s)/O	SD0_WP	I(s)	IRQ2	I(s)	-	-	(1)
H4	Vcc																		
H12	PG_1	I(s)/O	-	-	ET0_TXD2	O	VIO_D9	I(s)	MOSI0	I(s)/O	MTIOC3C	I(s)/O	HM_INT#/OM_ECS#	I(s)	-	-	-	-	(1)
H13	SD0_DAT5	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H14	SD0_RST#	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
H15	Vss																		
J1	PJ_0	I(s)/O	-	-	TRACECLK	O	SPDIF_OUT	O	-	-	SCK1	I(s)/O	SSIRxD3	I(s)	-	-	-	-	(1)
J2	PJ_1	I(s)/O	-	-	TRACECTL	O	SPDIF_IN	I(s)	-	-	RxD1	I(s)	VBUSIN0	I(s)	IRQ0	I(s)	-	-	(1)
J3	PJ_2	I(s)/O	-	-	TRACEDATA0	O	NFCE#	O	-	-	TxD1	O	SSITxD3	O	-	-	-	-	(1)
J4	Vss																		
J12	SD0_DAT7	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J13	SD0_DAT2	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J14	SD0_DAT4	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
J15	PVcc_SD0																		
K1	PJ_5	I(s)/O	-	-	TRACEDATA3	O	NFDATA2	I(s)/O	OVRCUR0	I(s)	MTIOC1A	I(s)/O	SSILRCK2	I(s)/O	IRQ4	I(s)	-	-	(1)
K2	PJ_3	I(s)/O	-	-	TRACEDATA1	O	NFDATA0	I(s)/O	-	-	RTS1#	I(s)/O	SSILRCK3	I(s)/O	-	-	-	-	(1)
K3	PJ_4	I(s)/O	-	-	TRACEDATA2	O	NFDATA1	I(s)/O	-	-	CTS1#	I(s)/O	SSIBCK3	I(s)/O	-	-	-	-	(1)
K4	PVcc																		
K12	SD0_CLK	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
K13	SD0_DAT3	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
K14	SD0_DAT0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
K15	SD0_DAT6	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
L1	PK_4	I(s)/O	-	-	-	-	NFDATA7	I(s)/O	OVRCUR1	I(s)	CAN0TX	O	MISO0	I(s)/O	IRQ6	I(s)	RMII1_RXD0	I(s)	(1)
L2	P3_5	I(s)/O	-	-	-	-	NFCLE	O	CC2_Ra0	I(s)	CAN0TX_DATARATE_EN	O	SSL00	I(s)/O	-	-	RMII1_RXD1	I(s)	(1)
L3	Vss																		
L4	Vcc																		
L12	PVcc																		
L13	Vss																		
L14	SD0_DAT1	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
L15	SD0_CMD	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(15)
M1	AUDIO_X1	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
M2	AUDIO_X2	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
M3	P3_2	I(s)/O	-	-	-	-	NFRE#	O	CC1_Ra1	I(s)	CAN1RX_DATARATE_EN	O	MOSI2	I(s)/O	-	-	RMII1_CRS_DV	I(s)	(1)
M4	Vss																		
M5	P3_3	I(s)/O	-	-	ET1_MDC	O	NFWE#	O	OTG_EXICEN0	O	CAN1TX	O	MISO2	I(s)/O	IRQ7	I(s)	-	-	(1)
M6	LVDSAPVcc																		
M7	LVDSPLLVcc																		
M8	NMI	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(10)
M9	USBVss																		
M10	PVcc																		
M11	Vcc																		
M12	P5_0	I(s)	-	-	AN000	I(a)	IRQ4	I(s)	SD0_CD	I(s)	SD1_CD	I(s)	-	-	-	-	-	-	(3)
M13	P5_4	I(s)	-	-	AN004	I(a)	IRQ0	I(s)	SD1_CD	I(s)	-	-	-	-	-	-	-	-	(3)
M14	P5_7	I(s)	-	-	AN007	I(a)	IRQ3	I(s)	-	-	-	-	-	-	-	-	-	-	(3)
M15	P5_5	I(s)	-	-	AN005	I(a)	IRQ1	I(s)	SD1_WP	I(s)	-	-	-	-	-	-	-	-	(3)
N1	P3_1	I(s)/O	-	-	-	-	NFALE	O	VBUSEN0	I(s)	CAN1RX	I(s)	RSPCK2	I(s)/O	IRQ6	I(s)	RMII1_RXER	I(s)	(1)
N2	PG_4	I(s)/O	-	-	ET0_TXER	O	VIO_D15	I(s)	RSPCK1	I(s)/O	MTIOC4A	I(s)/O	GTIOC1A	I(s)/O	-	-	-	-	(1)

Table 1.7 List of Pins (176-Pin BGA)

Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Simplified Circuit Diagram Figure 51.1 to Figure 51.15
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
N3	Vss																		
N4	PG_7	I(s)/O	-	-	ET0_RXD3	I(s)	VIO_D12	I(s)	SSL10	I(s)/O	MTIOC4D	I(s)/O	GTIOC2B	I(s)/O	-	-	-	-	(1)
N5	P4_2	I(s)/O	-	-	TxD0	O	TXOUT1P	O	SCI_TXD1	I(s)/O	SSITxD1	O	MTIOC8C	I(s)/O	IRQ2	I(s)	-	-	(1)
N6	P4_3	I(s)/O	-	-	RTS0#	I(s)/O	TXOUT1M	O	SCI_CTS1#/RTS1#	I(s)/O	SSILRCK1	I(s)/O	MTIOC8D	I(s)/O	IRQ3	I(s)	-	-	(1)
N7	P4_6	I(s)/O	-	-	ET0_EXOUT/ET0_SCLKIN	I(s)/O	TXCLKOUTP	O	SCI_TXD0	I(s)/O	TxD4	O	DACK0	O	-	-	-	-	(1)
N8	RES#	I(s)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(12)
N9	RREF0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
N10	Vss																		
N11	PLLVcc																		
N12	Vcc																		
N13	P5_2	I(s)	-	-	AN002	I(a)	IRQ6	I(s)	VBUSIN0	I(s)	-	-	-	-	-	-	-	-	(3)
N14	P5_3	I(s)	-	-	AN003	I(a)	IRQ7	I(s)	OTG_ID0	I(s)	-	-	-	-	-	-	-	-	(3)
N15	P5_6	I(s)	-	-	AN006	I(a)	IRQ2	I(s)	-	-	-	-	-	-	-	-	-	-	(3)
P1	PVcc																		
P2	Vss																		
P3	PG_6	I(s)/O	-	-	ET0_RXD2	I(s)	VIO_D13	I(s)	MISO1	I(s)/O	MTIOC4C	I(s)/O	GTIOC2A	I(s)/O	IRQ5	I(s)	-	-	(1)
P4	P4_0	I(s)/O	-	-	SCK0	I(s)/O	TXOUT0P	O	SCI_SCK1	I(s)/O	SSIBCK1	I(s)/O	MTIOC8A	I(s)/O	IRQ0	I(s)	-	-	(1)
P5	P4_4	I(s)/O	-	-	CTS0#	I(s)/O	TXOUT2P	O	SCI_CTS0#/RTS0#	I(s)/O	WDTOVF#/PERROUT#	O	OTG_EXICEN0	O	-	-	-	-	(1)
P6	P4_7	I(s)/O	-	-	ET0_WOL	O	TXCLKOUTM	O	SCI_SCK0	I(s)/O	SCK4	I(s)/O	TEND0	O	-	-	-	-	(1)
P7	USB_X2	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
P8	USBDPVcc0																		
P9	USBAPVcc0																		
P10	USBVss																		
P11	XTAL	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
P12	RTC_X2	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(14)
P13	AVcc																		
P14	Vcc																		
P15	P5_1	I(s)	-	-	AN001	I(a)	IRQ5	I(s)	SD0_WP	I(s)	SD1_WP	I(s)	-	-	-	-	-	-	(3)
R1	Vss																-	-	
R2	PG_5	I(s)/O	-	-	ET0_RXDV	I(s)	VIO_D14	I(s)	MOSI1	I(s)/O	MTIOC4B	I(s)/O	GTIOC1B	I(s)/O	-	-	-	-	(1)
R3	P3_4	I(s)/O	-	-	ET1_MDIO	I(s)/O	NFRB#	I(s)	CC2_Ra1	I(s)	CAN1TX_DATARATE_EN	O	SSL20	I(s)/O	-	-	-	-	(1)
R4	P4_1	I(s)/O	-	-	RxD0	I(s)	TXOUT0M	O	SCI_RXD1	I(s)/O	SSIRxD1	I(s)	MTIOC8B	I(s)/O	IRQ1	I(s)	-	-	(1)
R5	P4_5	I(s)/O	-	-	ET0_LINKSTA	I(s)	TXOUT2M	O	SCI_RXD0	I(s)/O	RxD4	I(s)	DREQ0	I(s)	-	-	-	-	(1)
R6	Vss																		
R7	USB_X1	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
R8	DP0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
R9	DM0	I/O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
R10	USBVss																		
R11	EXTAL	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(13)
R12	Vss																		
R13	RTC_X1	I	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(14)
R14	AVss																		
R15	Vcc																		

[Legend]
(s): Schmitt
(a): Analog
(o): Open drain

2. CPU

This product incorporates the Arm single-core Cortex-A9 MPCore, where the IP version is r4p1.

2.1 Features*1

- Instruction cache size: 32 Kbytes
- Data cache size*2: 32 Kbytes
- TLB size: 128 entries
- BTAC size: 512 entries
- GHB size: 4096 descriptors
- Instruction micro TLB: 32 entries
- Jazelle architecture extension: Full
- Media processing engine with NEON technology: Included
- FPU: Included
- PTM interface: Included
- Wrappers to support for power off and dormant mode: Not included
- Preload engine: Not included
- Number of interrupts: 0
- Accelerator Coherence Port: Not included
- Support for cache parity error detection*4: Yes

Note 1. For details, refer to Cortex-A9 MPCore Technical Reference Manual issued by Arm Ltd.
Note 2. Contents of memory regions which are set as write-through are not cached even if data caching is enabled. For details, refer to Cortex-A9 Technical Reference Manual issued by Arm Ltd.
Note 3. Hardware resets in response to requests from the watchdog timer for the Snoop Control Unit (SCU) of the MPCore are not supported.
Note 4. For handling of the parity error signal, see section 15, Watchdog Timer.

2.2 Configuration Signals

Table 2.1 shows the Cortex-A9 configuration signals and the settings.

Table 2.1 Cortex-A9 Configuration Signal Settings

Configuration Signal	Setting Values
CFGEND	1'b0
CFGNMFI	1'b0
CLUSTERID	4'h0
FILTEREN*1	1'b1
FILTERSTART[31:20]*1	12'hE00
FILTEREND[31:20]*1	12'hFFF
PERIPHBASE[31:13]*2	19'b111_1000_0000_0000_0000
TEINIT	1'b0
VINITHI	1'b1

Note 1. Do not change the initial settings (setting values) of these signals by software.
Note 2. The base address for the private memory area in the Cortex-A9 processor is H'F0000000. For details and overview of the registers located in the addresses relative to this base address, refer to Cortex-A9 MPCore Technical Reference Manual issued by Arm Ltd.

3. Boot Mode

This LSI can be booted from the memory connected to the CS0 space, the serial flash memory, the OctaFlash™*1, Xccela™ flash memory*2, and the HyperFlash™*3.

Note 1. OctaFlash is a trademark of Macronix International Co., Ltd.
Note 2. Xccela™ flash memory is a trademark of Micron Technology, Inc.
Note 3. HyperFlash is a trademark of Cypress Semiconductor Corporation.
Note 4. Booting from the NAND flash memory with an SD controller and the NAND flash memory with an MMC controller are not currently supported.

3.1 Features

Eight boot modes
Boot mode 0: Boots the LSI from the memory (bus width: 16 bits) connected to the CS0 space
Boot mode 1: Boots the LSI from the NAND flash memory with the SD controller*1
Boot mode 2: Boots the LSI from the NAND flash memory with the MMC controller*2 (data bus width: 8 bits*3)
Boot mode 3: Boots the LSI from the serial flash memory (SPI mode, 3.3-V products) connected to the SPI multi I/O bus space
Boot mode 4: Boots the LSI from the Octal-SPI flash memory*4 (SPI mode, 1.8-V products) connected to the SPI multi I/O bus space
Boot mode 5: Boots the LSI from the HyperFlash (1.8-V products) connected to the SPI multi I/O bus space
Boot mode 6: Boots the LSI from the OctaFlash (SPI mode, 1.8-V products) connected to the OctaFlash space
Boot mode 7: Boots the LSI from the HyperFlash (1.8-V products) connected to the HyperFlash space

Note 1. It is possible to boot the LSI from the embedded SD (eSD) defined by the SD specification part 1 eSD addendum version 2.10 standard.
Note 2. It is possible to boot the LSI from the eMMC device corresponding to the alternative boot operation mode of the JEDEC standard JESD84 A44 (MMCA 4.4) Standard. (It is not possible to boot the LSI from the MMC card.)
Note 3. The boot program is assumed to run when the data bus width is 8 bits.
Booting up by the boot program cannot proceed normally when the data bus width is not 8 bits.
Note 4. In this manual, the 8-bit SPI flash memory (OctaFlash or Xccela flash memory) with 1 chip select, 1 clock source, and 1 data strobe configuration is called "Octal-SPI flash memory".

3.2 Boot Mode and Pin Function Setting

This LSI can determine the boot mode using external pins when RES# is low. The external pin settings for selecting the boot mode are shown in Table 3.1.

Table 3.1 External Pin (MD_BOOT2 to MD_BOOT0) Settings and Corresponding Boot Modes

MD_BOOT2	MD_BOOT1	MD_BOOT0	Boot Mode
0	0	0	Boot Mode 0 (CS0-space 16-bit booting) Boots the LSI from the memory (bus width: 16 bits) connected to the CS0 space.
0	0	1	Boot Mode 1 (eSD booting) Boots the LSI from the NAND flash memory with the SD controller. The only way to boot this LSI chip is from channel 0 of the SD/MMC host interface in this mode.
0	1	0	Boot Mode 2 (eMMC booting) Boots the LSI from the NAND flash memory with the MMC controller. The only way to boot this LSI chip is from channel 0 of the SD/MMC host interface in this mode.
0	1	1	Boot Mode 3 (serial flash booting, 3.3-V products) Boots the LSI from the serial flash memory (SPI mode) connected to the SPI multi I/O bus space.
1	0	0	Boot Mode 4 (Octal-SPI flash booting, 1.8-V products) Boots the LSI from the Octal-SPI flash memory (SPI mode) connected to the SPI multi I/O bus space.

Table 3.1 External Pin (MD_BOOT2 to MD_BOOT0) Settings and Corresponding Boot Modes

MD_BOOT2	MD_BOOT1	MD_BOOT0	Boot Mode
1	0	1	Boot Mode 5 (HyperFlash booting 1, 1.8-V products) Boots the LSI from the HyperFlash connected to the SPI multi I/O bus space.
1	1	0	Boot Mode 6 (OctaFlash booting, 1.8-V products) Boots the LSI from the OctaFlash (SPI mode) connected to the OctaFlash space.
1	1	1	Boot Mode 7 (HyperFlash booting 2, 1.8-V products) Boots the LSI from the HyperFlash connected to the HyperFlash space.

3.3 Hardware Used in Each Boot Mode

Table 3.2 gives information about the hardware used in each boot mode.

Table 3.2 Hardware Used in Each Boot Mode

Boot Mode	Peripheral Module	Pins Used	Remarks*2
Boot Mode 0 (CS0-space 16-bit booting)	Bus state controller	A[20:1] D[15:0] CS0# RD# CKIO *1	CKIO clock frequency: P1φ
Boot Mode 1 (eSD booting)	SD/MMC host interface (channel 0)	SD0_CLK SD0_CMD SD0_D[3:0]	SD0_CLK frequency: Bφ/8
Boot Mode 2 (eMMC booting)	SD/MMC host interface (channel 0)	SD0_CLK SD0_CMD SD0_D[7:0] SD0_RST#	SD0_CLK frequency: Bφ/8
Boot Mode 3 (Serial flash booting, 3.3-V products)	SPI multi I/O bus controller	QSPI0_SPCLK QSPI0_SSL QSPI0_IO[1:0]	QSPI0_SPCLK frequency: P0φ/2
Boot Mode 4 (Octal-SPI flash booting, 1.8-V products)	SPI multi I/O bus controller	QSPI0_SPCLK QSPI0_SSL QSPI0_IO[1:0]	QSPI0_SPCLK frequency: P0φ/2
Boot Mode 5 (HyperFlash booting 1, 1.8-V products)	SPI multi I/O bus controller	QSPI0_SSL QSPI0_SPCLK QSPI1_SPCLK QSPI0_IO[3:0] QSPI1_IO[3:0] QSPI1_SSL RPC_RESET#	QSPI0_SPCLK, QSPI1_SPCLK frequency: P1φ/2
Boot Mode 6 (OctaFlash booting, 1.8-V products)	Octa Memory controller	OM_SCLK OM_CS0# OM_SIO[1:0]	OM_SCLK frequency: P0φ/2
Boot Mode 7 (HyperFlash booting 2, 1.8-V products)	HyperBus controller	HM_CS0# HM_CK HM_CK# HM_DQ[7:0] HM_RWDS HM_RESET#	HM_CK, HM_CK# frequency: P1φ/2

Note 1. The pins used in boot mode 0 work as general I/O pins immediately after this LSI is released from the reset state. The pin function should be changed to the external bus function through execution of the boot program.

Note 2. This column shows the operating frequency of the peripheral module in each boot mode while the boot program is being executed.

3.4 Exception Vector Address at a Reset in Each Boot Mode

In this LSI, the boot program is executed at the exception vector address H'FFFF_0000 (high vector) in all boot modes when a reset is generated. The address of the exception vector after the boot program is executed depends on the boot mode. In the boot mode 0, the exception vector is located at H'0000_0000 (low vector). In the boot modes 1 to 7, the exception vector is located at H'FFFF_0000 (high vector).
In this LSI, an on-chip ROM is allocated in area H'FFFF_0000 to H'FFFF_FFFF. The on-chip ROM has a boot program which executes processing corresponding to the boot mode set by the MD_BOOT2 to MD_BOOT0 external pins.
Table 3.3 lists the exception vector address for each boot mode.

Table 3.3 Exception Vector Address in Each Boot Mode

Boot Mode	Exception Vector Address after boot program execution	Memory Allocated at the Exception Vector Address
Boot Mode 0 (CS0-space 16-bit booting)	H'0000 0000 (low vector)	Memory connected to the CS0 space
Boot Mode 1 (eSD booting)	H'FFFF 0000 (high vector)	On-chip ROM
Boot Mode 2 (eMMC booting)	H'FFFF 0000 (high vector)	On-chip ROM
Boot Mode 3 (Serial flash booting, 3.3-V products)	H'FFFF 0000 (high vector)	On-chip ROM
Boot Mode 4 (Octal-SPI flash booting, 1.8-V products)	H'FFFF 0000 (high vector)	On-chip ROM
Boot Mode 5 (HyperFlash booting 1, 1.8-V products)	H'FFFF 0000 (high vector)	On-chip ROM
Boot Mode 6 (OctaFlash booting, 1.8-V products)	H'FFFF 0000 (high vector)	On-chip ROM
Boot Mode 7 (HyperFlash booting 2, 1.8-V products)	H'FFFF 0000 (high vector)	On-chip ROM

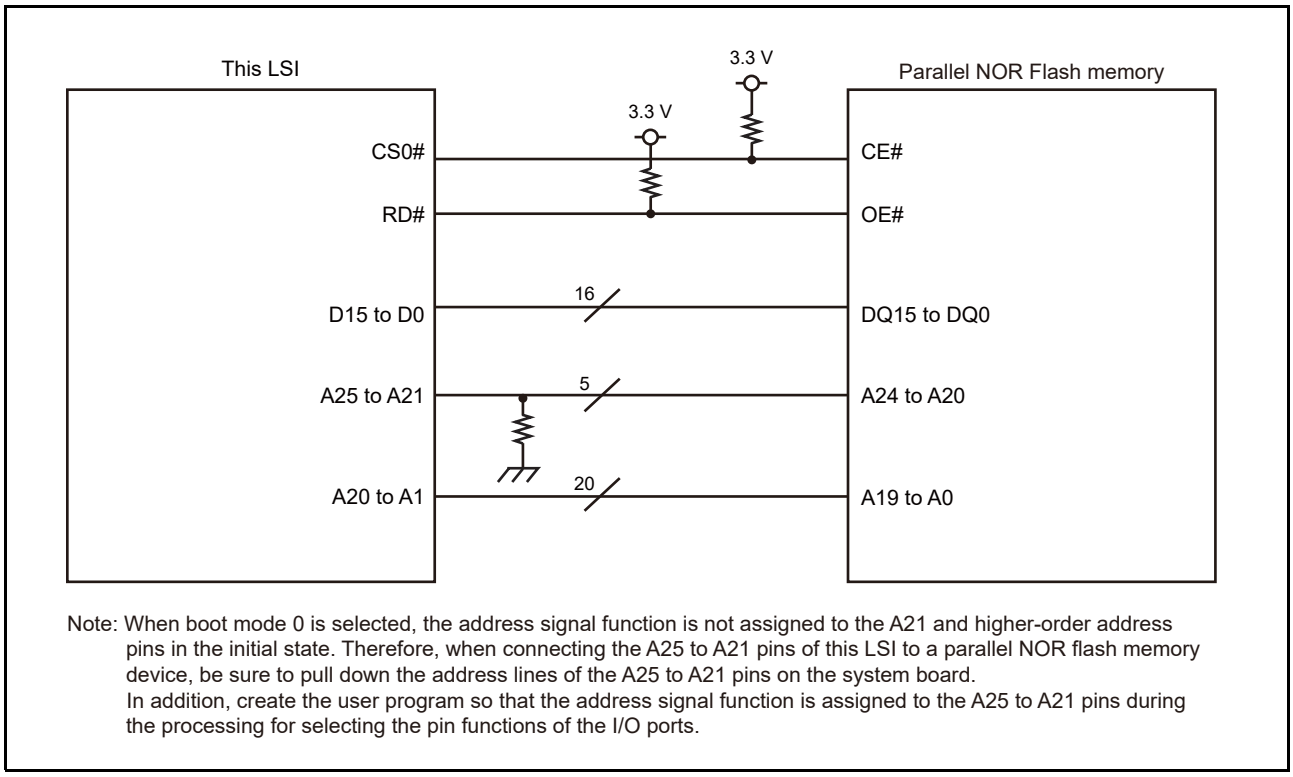
3.5 Operation

3.5.1 Boot Mode 0

In boot mode 0, this LSI is booted from the memory connected to the CS0 space. In this mode, this LSI operates as follows:

After the power-on reset is canceled, the boot program stored in the on-chip ROM (starting from H'FFFF_0000) is executed. In the boot program, set the pins to access the CS0 space. Set the V bit of the CP15 system control register (SCTLR) to 0 and switch to the low vector. After the power-on reset is canceled, program execution is started from H'0000_0000 in the memory connected to the CS0 space.

Figure 3.1 shows an example of connection with an external memory of CS0 space. Table 3.4 shows the register setting values of each peripheral module in boot mode 0.



Note: When boot mode 0 is selected, the address signal function is not assigned to the A21 and higher-order address pins in the initial state. Therefore, when connecting the A25 to A21 pins of this LSI to a parallel NOR flash memory device, be sure to pull down the address lines of the A25 to A21 pins on the system board.
In addition, create the user program so that the address signal function is assigned to the A25 to A21 pins during the processing for selecting the pin functions of the I/O ports.

Figure 3.1 Example of connection with an external memory of CS0 space (Parallel NOR Flash memory)

Table 3.4 Register setting values of each peripheral module in boot mode 0

Peripheral module	Register	Initial values after reset	Setting values after boot program execution
GPIO	PORT0.PMR	H'00	H'7F
	PORT1.PMR	H'00	H'1F
	PORT2.PMR	H'00	H'0F
	PORT7.PMR	H'00	H'80
	PORT8.PMR	H'00	H'FE
	PORT9.PMR	H'00	H'FF
	PORTA.PMR	H'00	H'1F
	PORTB.PMR	H'00	H'08
	P00PFS	H'00	H'01
	P01PFS	H'00	H'01
	P02PFS	H'00	H'01
	P03PFS	H'00	H'01
	P04PFS	H'00	H'01
	P05PFS	H'00	H'01
	P06PFS	H'00	H'01
	P10PFS	H'00	H'01
	P11PFS	H'00	H'01
	P12PFS	H'00	H'01
	P13PFS	H'00	H'01
	P14PFS	H'00	H'01
	P20PFS	H'00	H'01
	P21PFS	H'00	H'01
	P22PFS	H'00	H'01
	P23PFS	H'00	H'01
	P77PFS	H'00	H'01
	P81PFS	H'00	H'01
	P82PFS	H'00	H'01
	P83PFS	H'00	H'01
	P84PFS	H'00	H'01
	P85PFS	H'00	H'01
	P86PFS	H'00	H'01
	P87PFS	H'00	H'01
	P90PFS	H'00	H'01
	P91PFS	H'00	H'01
	P92PFS	H'00	H'01
	P93PFS	H'00	H'01
	P94PFS	H'00	H'01
	P95PFS	H'00	H'01
	P96PFS	H'00	H'01
	P97PFS	H'00	H'01
	PA0PFS	H'00	H'01
	PA1PFS	H'00	H'01
	PA2PFS	H'00	H'01
	PA3PFS	H'00	H'01

Table 3.4 Register setting values of each peripheral module in boot mode 0

Peripheral module	Register	Initial values after reset	Setting values after boot program execution
GPIO	PA4PFS	H'00	H’01
	PB3PFS	H'00	H’01

3.5.2 Boot Mode 1

In boot mode 1, this LSI is booted from the NAND flash memory with the SD controller that is connected to channel 0 of the SD/MMC host interface. In this mode, this LSI operates as follows.
After this LSI is released from the power-on reset state, it executes the boot program stored in the on-chip ROM (starting from the address H'FFFF_0000).
The boot program releases channel 0 of the SD/MMC host interface from the module standby state, sets up the registers in the channel, transfers the program stored in the NAND flash memory with the SD controller to the area starting from the address H'8002_4000 in the large-capacity on-chip RAM for the size of the program.
The program that the boot program transfers to the large-capacity on-chip RAM is called a loader program.
The size of the loader program should be between 512 bytes and 4,046,848 bytes. If the size of the user application exceeds 4,046,848 bytes, create a program for transferring the application to the external memory and implement it in the loader program.
The loader program needs to be stored in the NAND flash memory with the SD controller in accordance with the loader program storage specifications.
After the processing of the boot program is completed, execution branches to the address H'8002_4000 (the large-capacity on-chip RAM).
The boot program uses the address area from H'8002_0000 to H'8002_3FFF as work memory.
Figure 3.2 is a schematic diagram of the specifications of boot mode 1. Figure 3.3 shows the loader program storage specifications. Figure 3.4 shows the specifications of the loader program size block used in boot mode 1.
In boot mode 1, the SD0_WP and SD0_CD pins are not used. In the SD/MMC host interface, dedicated pins are assigned to the command, clock, and data signals, but the write protect and card detection pins are not dedicated pins but multiplexed pins. For boot operation, be sure to supply the 3.3-V power to the SDVcc pin.

Note: Developing a host device compliant with the SD specification requires execution of an SD Host/Ancillary Product License Agreement (SD HALA).

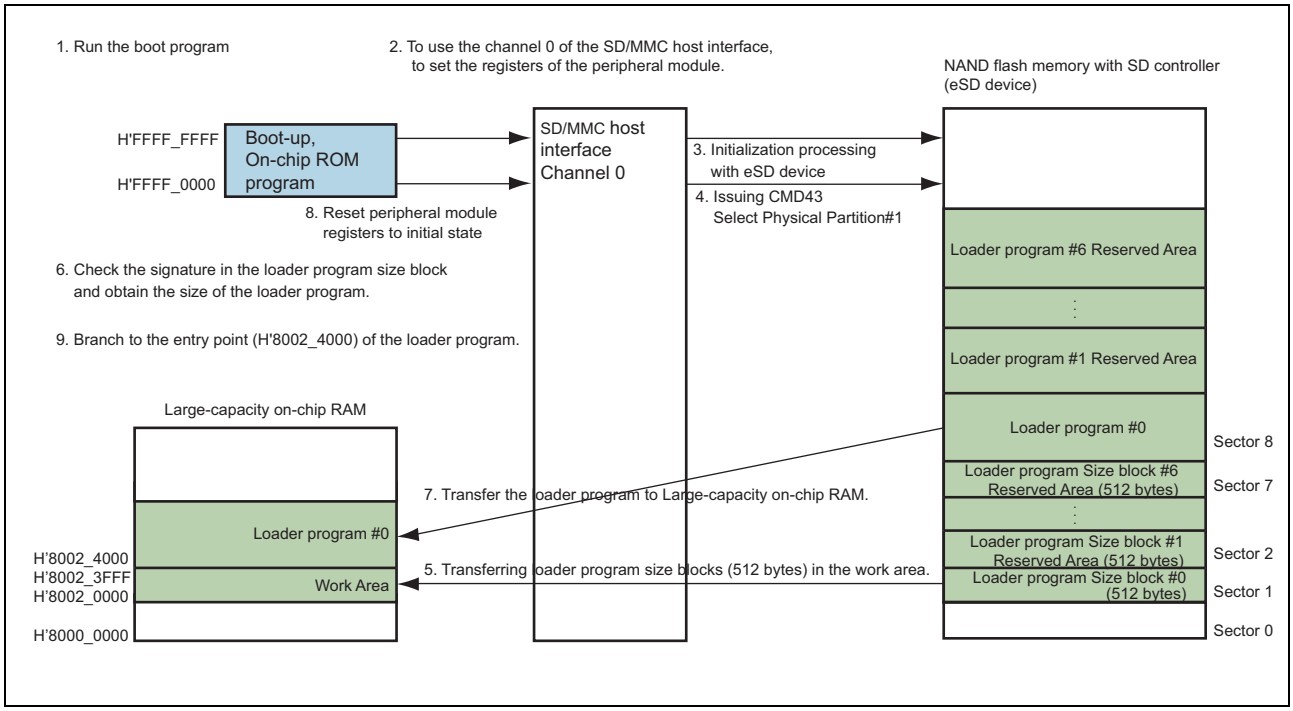


Figure 3.2 Schematic View of Specification for Boot Mode 1

1. The program stored in the on-chip ROM is executed after this LSI is released from the power-on reset state.
2. The program stored in the on-chip ROM sets up the peripheral module registers to access the eSD device connected to channel 0 of the SD/MMC host interface.
3. The processing for initializing the eSD device is executed.
4. The partition switch command (CMD43) is issued to the eSD device to select physical partition #1 (boot code area). When the connected eSD device supports this command, the processing after this step is applied to physical partition #1.
   When an SD card is connected, the processing after this step handles the SD card as a single-partition device, so booting from the SD card becomes possible.
   (When an SD card is connected, issuing the partition switch command will generate an error because the SD card does not support this command. However, the program stored in the on-chip ROM does not check this error.)
5. The read command is issued from channel 0 of the SD/MMC host interface to the eSD device to transfer loader program size block #0 to the work area in the large-capacity on-chip RAM.
6. The signature (H'AA55) in the loader program size block transferred to the large-capacity on-chip RAM is checked. When the signature matches the expected value, the size of the loader program is obtained from the loader program size block and the processing for transfer of the loader program (step 7) is executed. When the signature does not match the expected value, transfer of the loader program size block from the next multiplexed area begins. When reading of loader program size blocks #0 to #6 has failed, execution enters an infinite loop in the program stored in the on-chip ROM and the boot processing is terminated.
7. The read command is issued from channel 0 of the SD/MMC host interface to the eSD device to transfer loader program #0 to the area starting from the address H'8002_4000 in the large-capacity on-chip RAM for the size of the loader program.
   When the SD/MMC host interface detects a data transfer error during transfer, transfer of the loader program from the next multiplexed area begins. When reading of loader programs #0 to #6 has failed, execution enters an infinite loop in the program stored in the on-chip ROM and the boot processing is terminated.
8. The values of the registers changed to use channel 0 of the SD/MMC host interface are restored to the initial state.
9. Execution branches to the entry point (address H'8002_4000) of the loader program.

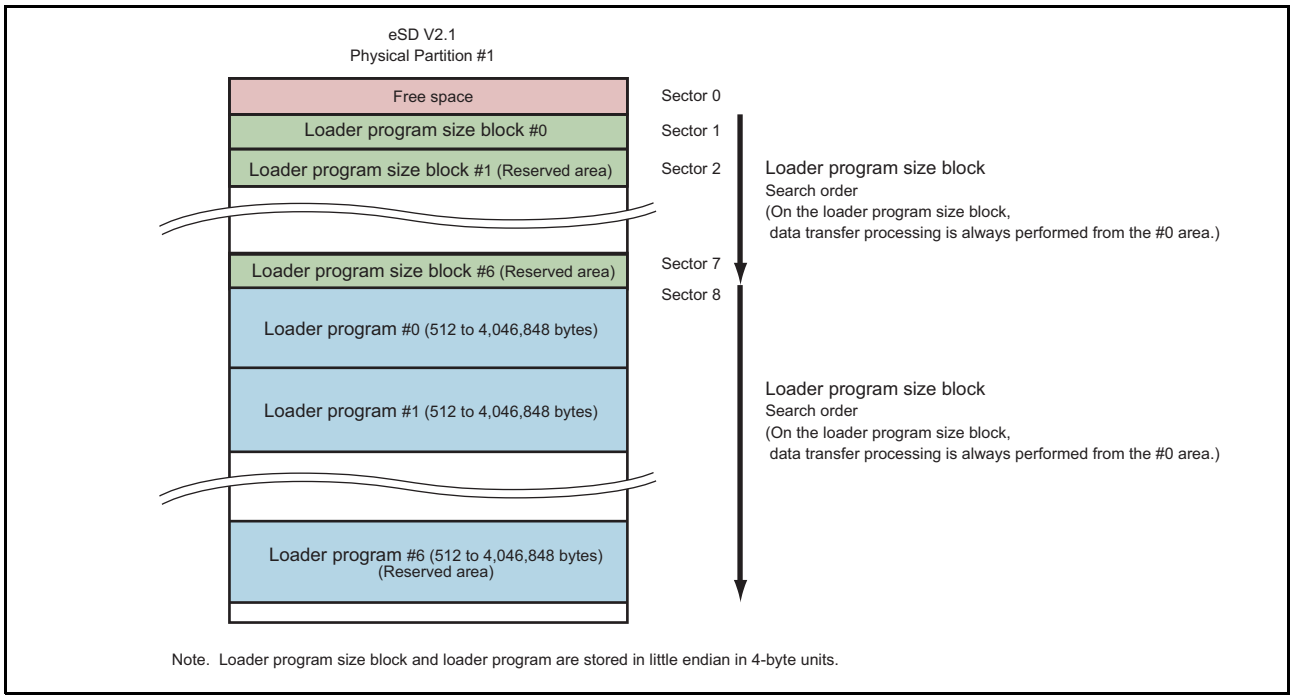


Figure 3.3 Loader program storage specification

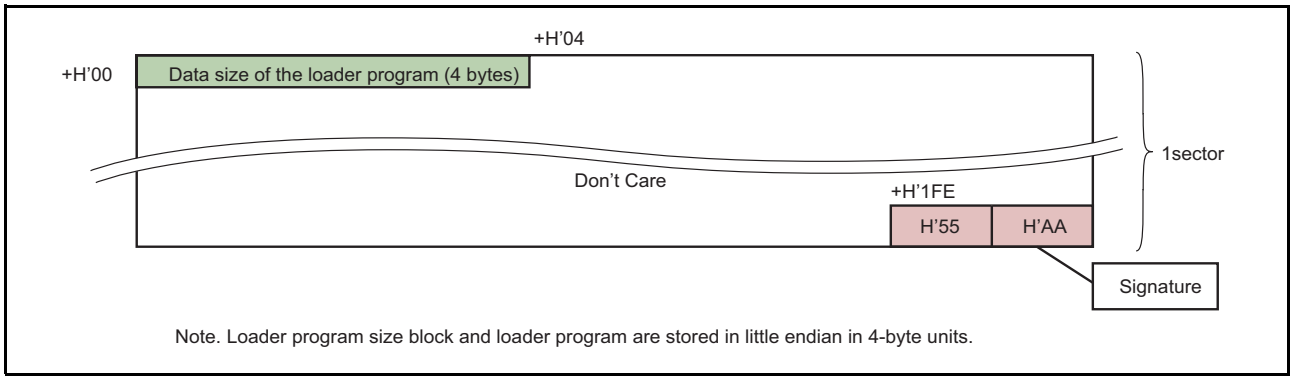


Figure 3.4 Loader program size Block configuration

3.5.3 Boot Mode 2

In boot mode 2, this LSI is booted from the NAND flash memory with the MMC controller that is connected to channel 0 of the SD/MMC host interface. In this mode, this LSI operates as follows.
After this LSI is released from the power-on reset state, it executes the boot program stored in the on-chip ROM (starting from the address H'FFFF_0000).
The boot program releases channel 0 of the SD/MMC host interface from the module standby state, sets up the registers in the channel, transfers the program stored in the NAND flash memory with the MMC controller to the area starting from the address H'8002_4000 in the large-capacity on-chip RAM for the size of the program. The program that the boot program transfers to the large-capacity on-chip RAM is called a loader program. The size of the loader program should be between 512 bytes to 4,046,848 bytes. If the size of the user application exceeds 4,046,848 bytes, create a program for transferring the application to the external memory and implement it in the loader program.
The loader program needs to be stored in the NAND flash memory with the MMC controller in accordance with the loader program storage specifications.
After the processing of the boot program is completed, execution branches to the address H'8002_4000 (the large-capacity on-chip RAM). The boot program uses the address area from H'8002_0000 to H'8002_3FFF as work memory. Note that the data bus width is eight bits in boot mode 2.
Figure 3.5 is a schematic diagram of the specifications of boot mode 2. Figure 3.6 shows the loader program storage specifications. Figure 3.7 shows the specifications of the loader program size block used in boot mode 2. Figure 3.8 shows an example of connections with the eMMC device.
In addition, the extended CSD register in the eMMC device needs to be set up appropriately. Table 3.5 shows the values that should be specified in extended CSD register.

Note: For details of the specifications of the eMMC device, refer to the JESD84 A44 (MMCA 4.4) standard.

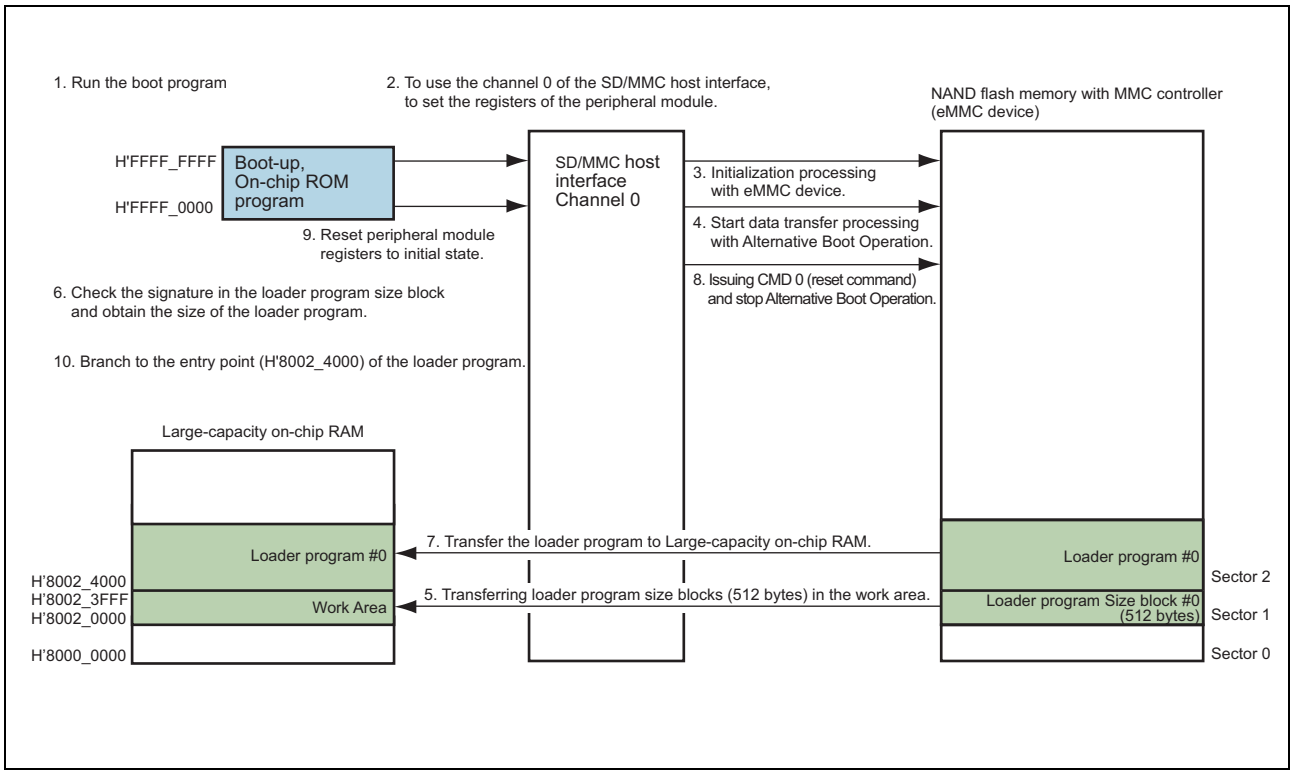


Figure 3.5 Schematic View of Specification for Boot Mode 2

1. The program stored in the on-chip ROM is executed after this LSI is released from the power-on reset state.
2. The program stored in the on-chip ROM sets up the peripheral module registers to access the eMMC device connected to channel 0 of the SD/MMC host interface.
3. The processing for initializing the eMMC device is executed.
4. The Alternative Boot Operation start command is issued to the eMMC device. In the Alternative Boot Operation mode, reading from sector 0 in the partition selected in the [179] field (PARTITION_CONFIG) of the extended CSD (EXT_CSD) register in the eMMC device begins.
5. After transfer from sector 0 to the work area in the large-capacity on-chip RAM is completed (Note), loader program block size #0 is transferred from sector 1.
   Note: In the Alternative Boot Operation mode, the read processing begins from sector 0 of the partition selected in the EXT_CSD[179] field. The boot program does not use the data in sector 0.
6. The signature (H'AA55) in the loader program size block transferred in step 5 is checked.
   When the signature matches the expected value, the size of the loader program is obtained from the loader program size block and the processing for transfer of the loader program (step 7) is executed.
   When the signature does not match the expected value, execution enters an infinite loop in the program stored in the on-chip ROM and the boot processing is terminated.
7. In the Alternative Boot Operation mode, loader program #0 is transferred to the area starting from the address H'8002_4000 in the large-capacity on-chip RAM for the size of the loader program.
   When the SD/MMC host interface detects a data transfer error during transfer, execution enters an infinite loop in the program stored in the on-chip ROM and the boot processing is terminated.
8. The reset command (CMD0/Reset) is issued from channel 0 of the SD/MMC host interface to the eMMC device to exit from the Alternative Boot Operation mode.
9. The values of the registers changed to use channel 0 of the SD/MMC host interface are restored to the initial state.
10. Execution branches to the entry point (address H'8002_4000) of the loader program.

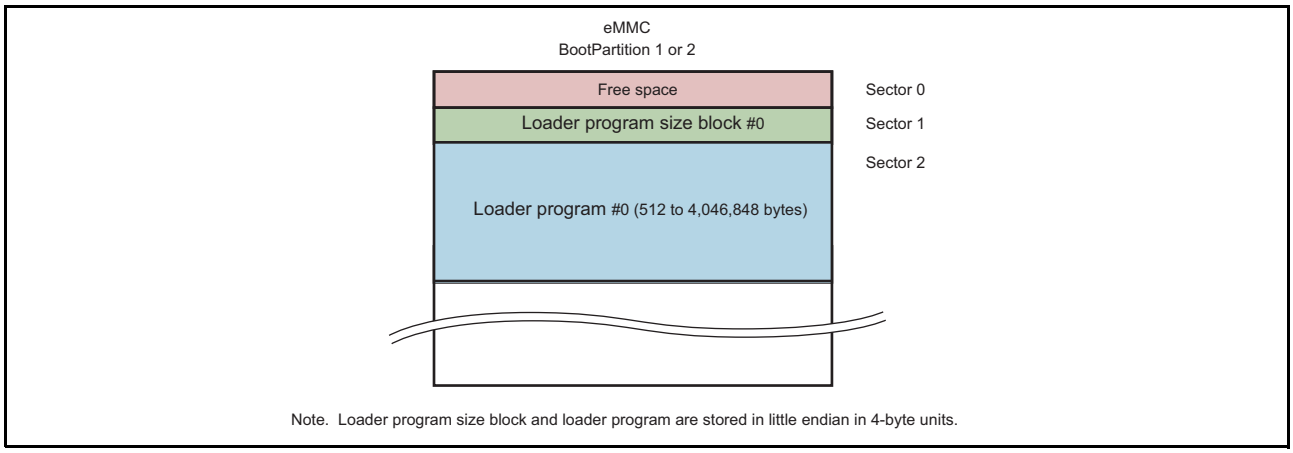


Figure 3.6 Loader program storage specification

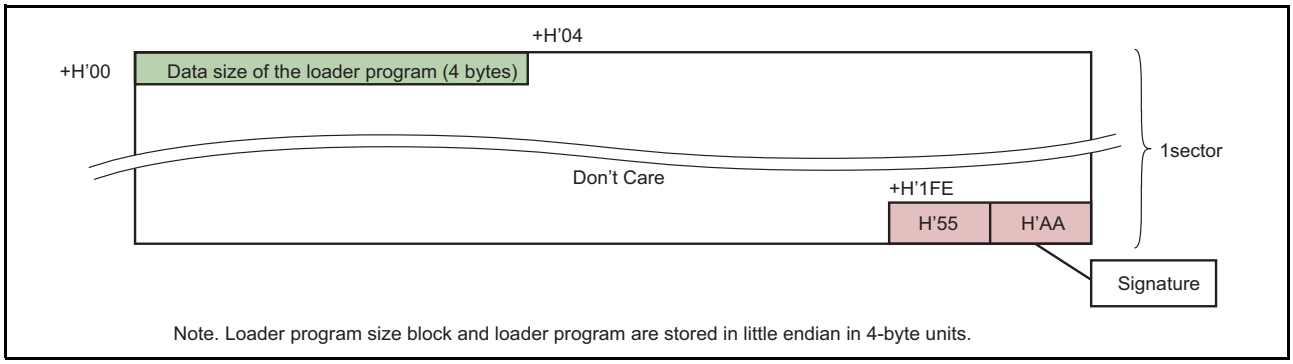


Figure 3.7 Loader program size Block configuration

Table 3.5 Extended CSD register setting value

eMMC device Fields in Extended CSD	Setting contents
BOOT_BUS_WIDTH[177]	Sets BOOT_MODE (Bit[4:3]) to B'00. (During Alternative Boot Operation, data is transferred in SDR mode.) Sets BOOT_BUS_WIDTH (Bit[1:0]) to B'10. (During Alternative Boot Operation, the data bus operates with 8 bits width.)
PARTITION_CONFIG[179]	Sets BOOT_ACK (Bit[6]) to B'0. (During Alternative Boot Operation, eMMC device does not output BOOT Acknowledge.) Set the boot partition to BOOT_PARTITION_ENABLE (Bit[5:3]). B'001: Boot from sector 0 of boot partition 1 in boot operation. B'010: Boot from sector 0 of boot partition 2 in boot operation.

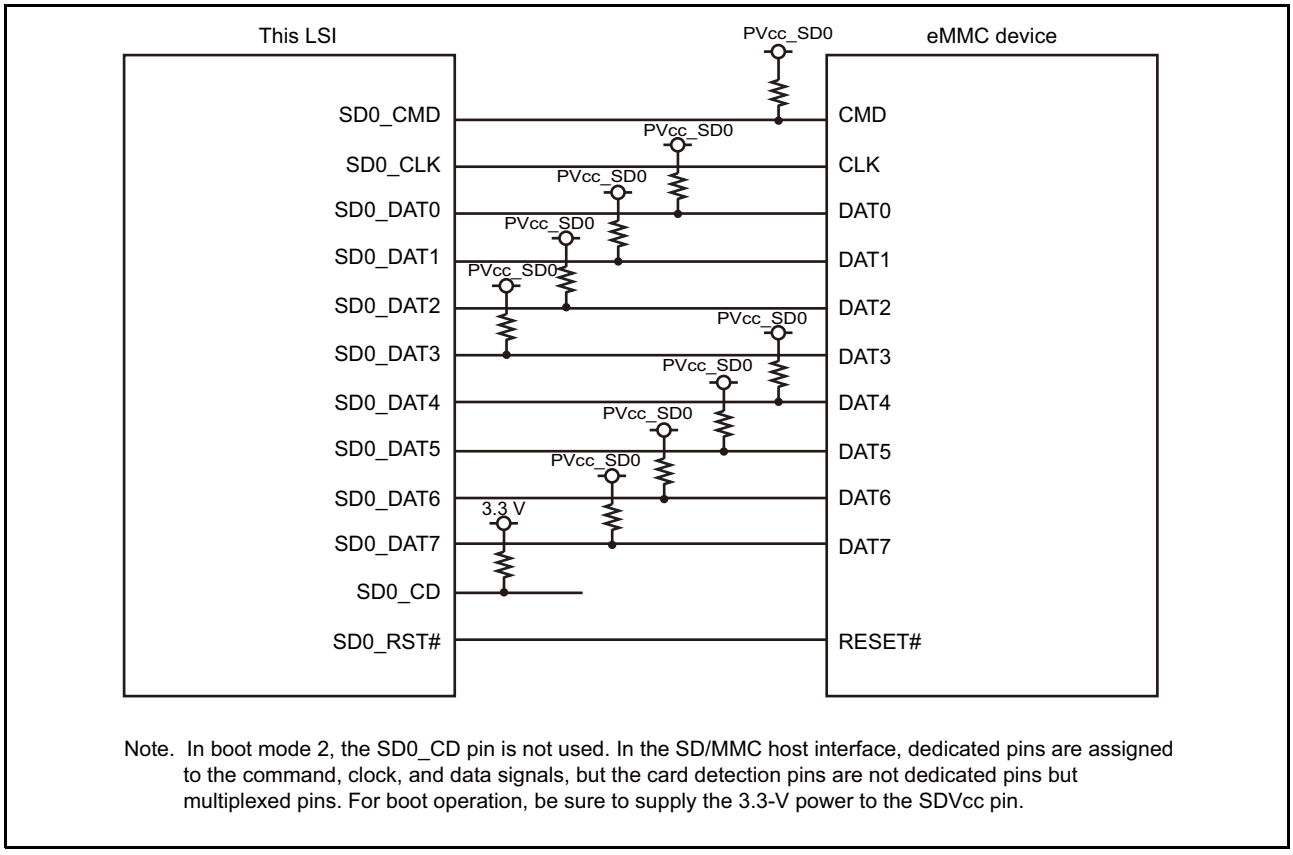


Figure 3.8 Example of connection with eMMC device

3.5.4 Boot Mode 3

In boot mode 3, booting up is from the serial flash memory (3.3-V products) connected to the SPI multi I/O bus space. In this mode, this LSI operates as follows:

After the power-on reset is canceled, the boot program stored in the on-chip ROM (starting from H'FFFF_0000) is executed. The boot program clears the module standby state of the SPI multi I/O bus controller and configures the SPI multi I/O bus controller in external address space read mode.

With this configuration, this LSI converts reads from the SPI multi I/O bus space to SPI communications and is ready to read directly from the connected serial flash memory.

The boot program configures a read command (opcode: 03H, address: 3 bytes, dummy cycle: none) as the command to the serial flash memory used for SPI communication conversion.

Figure 3.9 shows the control signals output to the serial flash memory through SPI communication conversion.

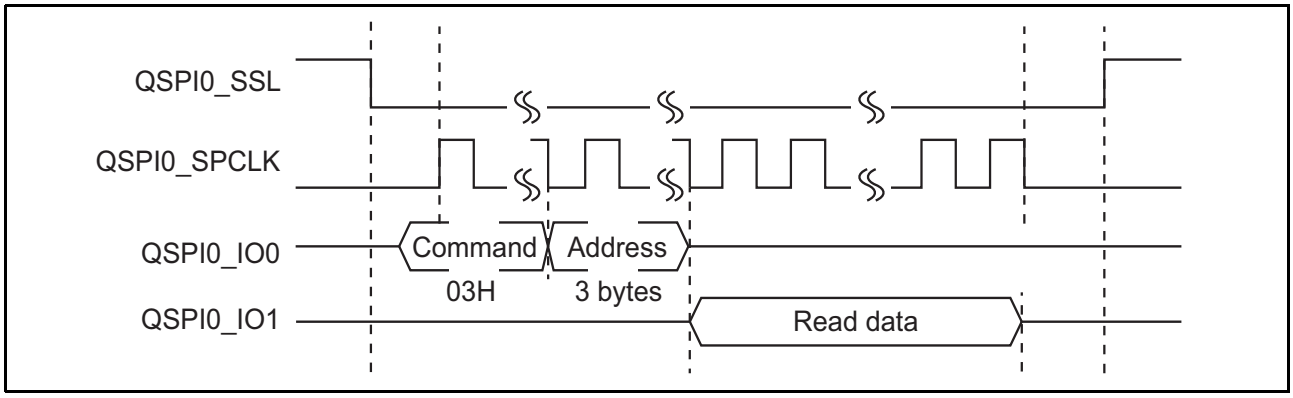


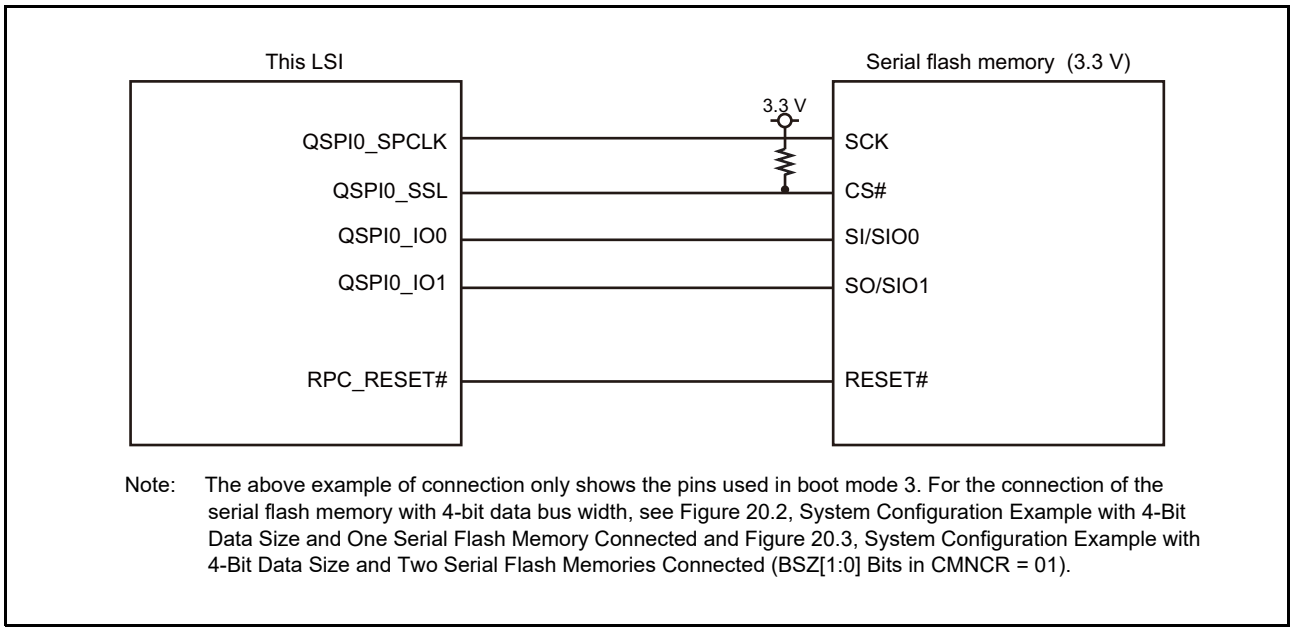
Figure 3.9 Control signals output to the serial flash memory through SPI communication conversion

The boot program uses the area at H'8002_0000 to H'8002_3FFF as work memory. It branches to H'2000_0000 (SPI multi I/O bus space) at the end of the processing.

Table 3.6 shows the setting values of each peripheral module after executing the boot program, and Figure 3.10 shows an example of connection with the serial flash memory.

Table 3.6 Register setting value of each peripheral module in boot mode 3

Peripheral module	Register	Initial value after reset	Setting value after boot program execution
Power-Down Modes	STBCR8	H'FF	H'F7
Clock Pulse Generator	SCLKSEL	H'0000	H'0000
SPI multi I/O bus controller	CMNCR	H'0155_7301	H'01AA_A200
	SSLDR	H'0000_0000	H'0000_0000
	DRCR	H'001F_0100	H'0003_0100
	DRCMR	H'00A0_0000	H'0003_0000
	DRENR	H'A222_D400	H'0000_4700
	DRDMCR	H'0000_000B	H'0000_0000
	DRDRENR	H'0000_5101	H'0000_0000
	PHYCNT	H'0000_0263	H'0000_0260
	PHYOFFSET1	H'2151_1144	H'3151_1144
	PHYINT	H'0707_0002	H'0707_0002
GPIO	PPOC	H'0000_0000F	H'0000_0101
	PSPIBSC	H'0FFF_FFFF	H'0555_5555

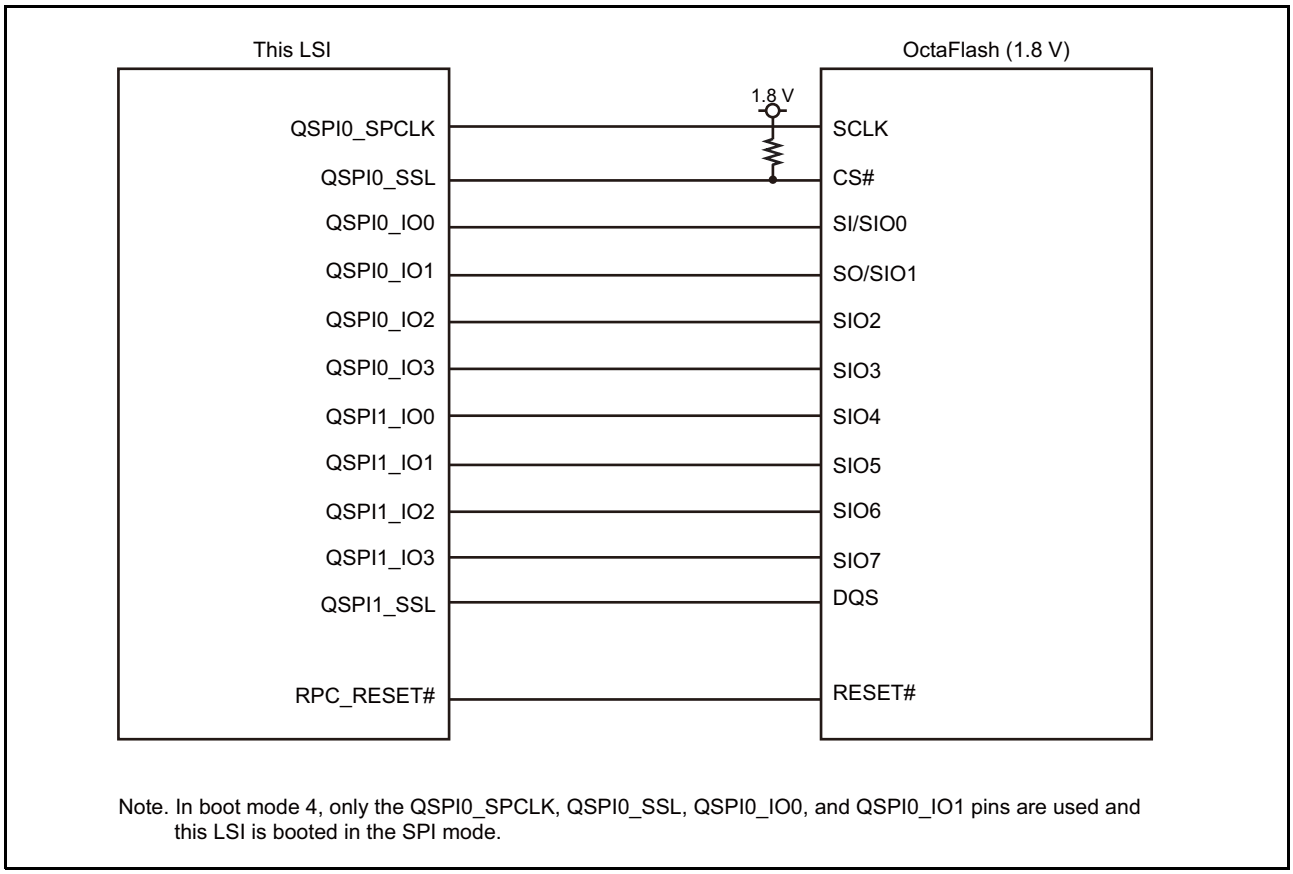


Note: The above example of connection only shows the pins used in boot mode 3. For the connection of the serial flash memory with 4-bit data bus width, see Figure 20.2, System Configuration Example with 4-Bit Data Size and One Serial Flash Memory Connected and Figure 20.3, System Configuration Example with 4-Bit Data Size and Two Serial Flash Memories Connected (BSZ[1:0] Bits in CMNCR = 01).

Figure 3.10 Connection example when using boot mode 3

3.5.5 Boot Mode 4

In boot mode 4, this LSI is booted from the Octal-SPI flash memory (1.8-V products) connected to the SPI multi I/O bus space. In this mode, booting from the serial flash memory is not supported.
The operation of this LSI in this mode is the same as that in boot mode 3. Refer to section 3.5.4, Boot Mode 3. Table 3.7 shows the setting values of each peripheral module after executing the boot program, and Figure 3.11 shows an example of connection with the OctaFlash.



Note. In boot mode 4, only the QSPI0_SPCLK, QSPI0_SSL, QSPI0_IO0, and QSPI0_IO1 pins are used and this LSI is booted in the SPI mode.

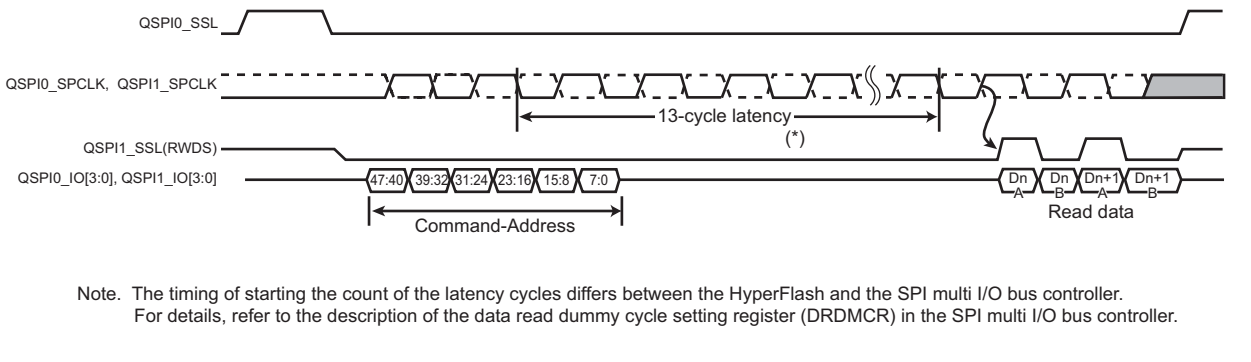
Figure 3.11 Connection example when using boot mode 4

Table 3.7 Register setting value of each peripheral module in boot mode 4

Peripheral module	Register	Initial value after reset	Setting value after boot program execution
general I/O port	PPOC	H'0000_000F	H'0000_0100
	PSPIBSC	H'0FFF_FFFF	H'0FFF_FFFF
The register settings of modules other than the general I/O port registers listed above are the same as those for boot mode 3.			

3.5.6 Boot Mode 5

In boot mode 5, this LSI is booted from the HyperFlash (1.8-V products) connected to the SPI multi I/O bus space.
In this mode, this LSI operates as follows.
After this LSI is released from the power-on reset state, it executes the boot program stored in the on-chip ROM (starting from the address H'FFFF_0000).
The boot program releases the SPI multi I/O bus controller from the module standby state and places the SPI multi I/O bus controller in the external address space read mode that supports the HyperFlash. In this mode, this LSI can convert read transactions to the SPI multi I/O bus space into the HyperBus protocol and read data directly from the connected HyperFlash.
Figure 3.12 shows the control signals output to the HyperFlash after conversion to the HyperBus protocol.
Note that the parameters for read access to the HyperFlash should be set to the fixed values for boot mode 5. Be sure to set up the HyperFlash configuration register (Non-Volatile) as shown in Table 3.8



Note. The timing of starting the count of the latency cycles differs between the HyperFlash and the SPI multi I/O bus controller. For details, refer to the description of the data read dummy cycle setting register (DRDMCR) in the SPI multi I/O bus controller.

Figure 3.12 Control signal output to HyperFlash by HyperBus protocol conversion

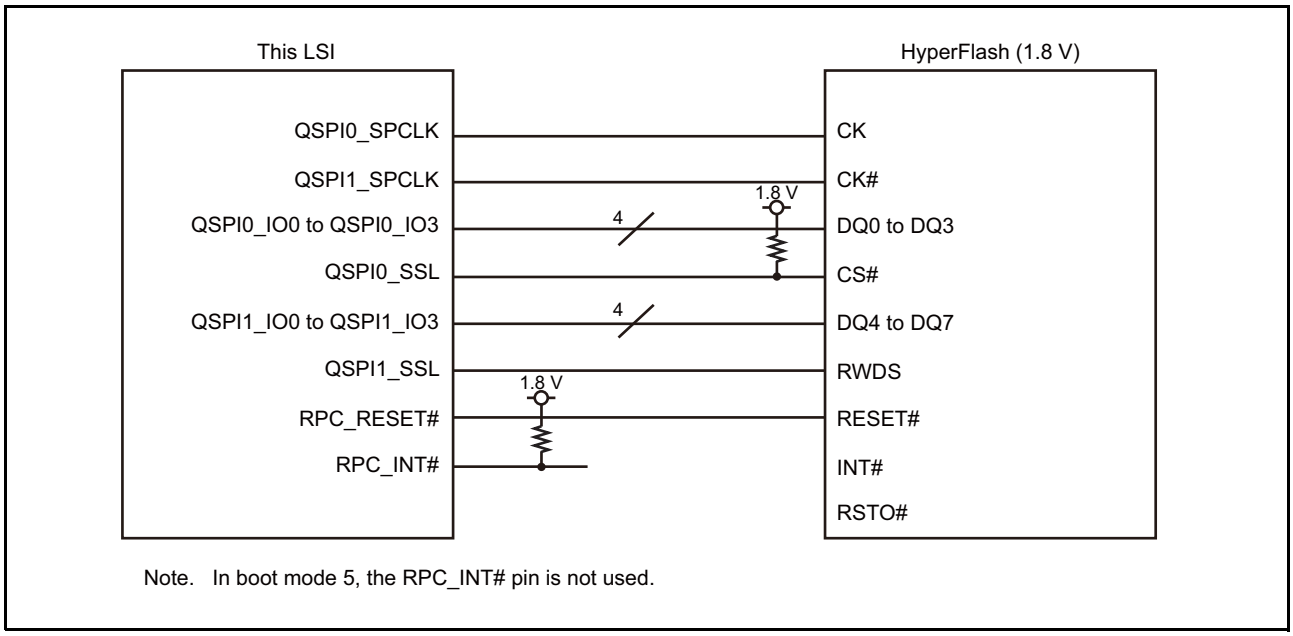
The boot program uses the area at H'8002_0000 to H'8002_3FFF as work memory. It branches to H'2000_0000 (SPI multi I/O bus space) at the end of the processing.
Table 3.9 shows the setting values of each peripheral module after executing the boot program, and Figure 3.13 shows an example of connection with HyperFlash.

Table 3.8 HyperFlash configuration register (Non-Volatile) setting value

Bit	Bit name	Description
[11]	NVCR Freeze	B'1: VCR, NVCR rewritable
[7:4]	Read Latency	B'1000: 13-Clock

Table 3.9 Register setting value of each peripheral module of the boot mode 5

Peripheral module	Register	Initial values after reset	Setting values after boot program execution
Power-Down Modes	STBCR8	H'FF	H'F7
Clock Pulse Generator	SCLKSEL	H'0000	H'0001
SPI multi I/O bus controller	CMNCR	H'0155_7301	H'0155_7301
	SSLDR	H'0000_0000	H'0000_0000
	DRCR	H'001F_0100	H'0003_0100
	DRCMR	H'00A0_0000	H'00A0_0000
	DRENR	H'A222_D400	H'A222_D400
	DRDMCR	H'0000_000B	H'0000_000B
	DRDRENR	H'0000_5101	H'0000_5101
	PHYCNT	H'0000_0263	H'0000_0263
	PHYOFFSET1	H'2151_1144	H'2151_1144
	PHYINT	H'0707_0002	H'0707_0002
General I/O Port	PPOC	H'0000_000F	H'0000_000F
	PSPIBSC	H'0FFF_FFFF	H'0FFF_FFFF



Note. In boot mode 5, the RPC_INT# pin is not used.

Figure 3.13 Connection example when using boot mode 5

3.5.7 Boot Mode 6

In boot mode 6, booting up is from the OctaFlash (1.8-V products) connected to the OctaFlash space. In this boot mode, booting from the serial flash memory is not supported.
In this mode, this LSI operates as follows:
After the power-on reset is canceled, the boot program stored in the on-chip ROM (starting from H'FFFF_0000) is executed.
The boot program cancels the module standby state of the Octa memory controller and configures the Octa memory controller to read mode compatible with SPI communication.
With this configuration, this LSI converts the read transaction to OctaFlash space into SPI communication and ready to read directly from the OctaFlash connected to the OctaFlash space.
The boot program configures a read command (opcode: 03H, address: 3 bytes, dummy cycle: none) as the command to the OctaFlash used for SPI communication conversion.
Figure 3.14 shows the control signals output to the OctaFlash by SPI communication conversion.

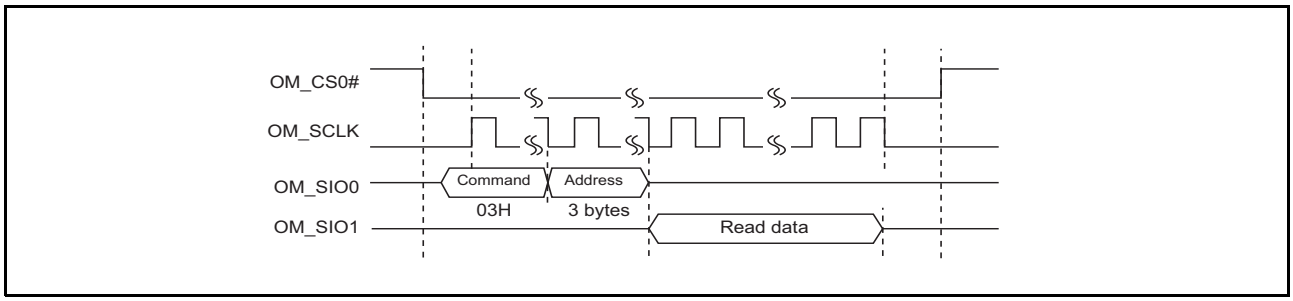
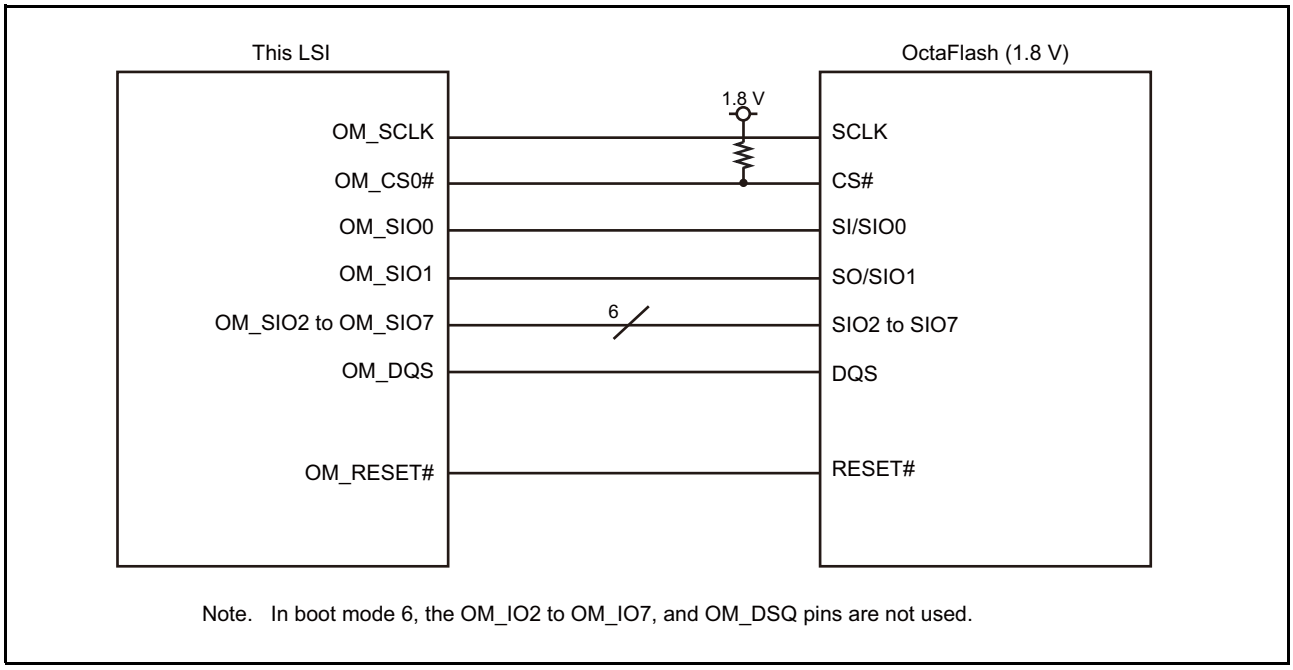


Figure 3.14 Control signals output to the OctaFlash

The boot program uses the area at H'8002_0000 to H'8002_3FFF as work memory. It branches to H'5000_0000 (OctaFlash space) at the end of the processing.
Table 3.10 shows the setting values of each peripheral module after executing the boot program, and Figure 3.15 shows an example of connection with OctaFlash.

Table 3.10 Register setting value of each peripheral module in boot mode 6

Peripheral module	Register	Initial values after reset	Setting values after boot program execution
Power-Down Modes	STBCR9	H'FF	H'FB
Clock Pulse Generator	SCLKSEL	H'0000	H'0000
Octa Memory controller	DSR0	H'0000_0000	H'0100_0000
	DRCSTR	H'0000_0000	H'0000_0000
	CDSR	H'0000_0000	H'8000_0000
	MDLR	H'0000_0000	H'0000_0000
	MRWCR0	H'0000_0000	H'0000_0003
	MRWCSR	H'0000_0000	H'0000_000B
General I/O Port	PHMOM0	3FFF_7FFE	3FFF_7FFF



Note. In boot mode 6, the OM_IO2 to OM_IO7, and OM_DSQ pins are not used.

Figure 3.15 Connection example when using boot mode 6

3.5.8 Boot Mode 7

In boot mode 7, this LSI is booted from the HyperFlash (1.8-V products) connected to the HyperFlash space.
In this mode, this LSI operates as follows.
After this LSI is released from the power-on reset state, it executes the boot program stored in the on-chip ROM (starting from the address H'FFFF_0000).
The boot program releases the HyperBus controller from the module standby state and sets up the registers in the HyperBus controller.
With these register settings, this LSI can convert read transactions to the HyperFlash space into the HyperBus protocol and read data directly from the connected HyperFlash. Figure 3.16 shows the control signals output to the HyperFlash after conversion to the HyperBus protocol.

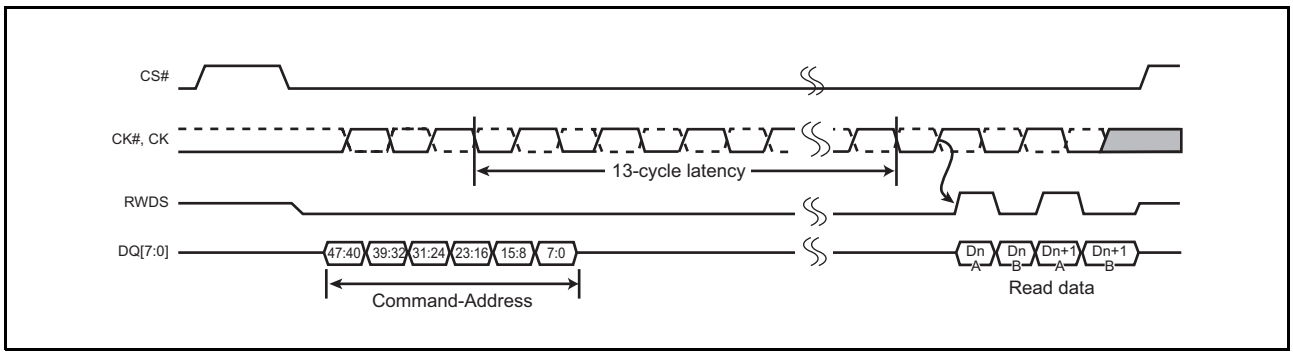
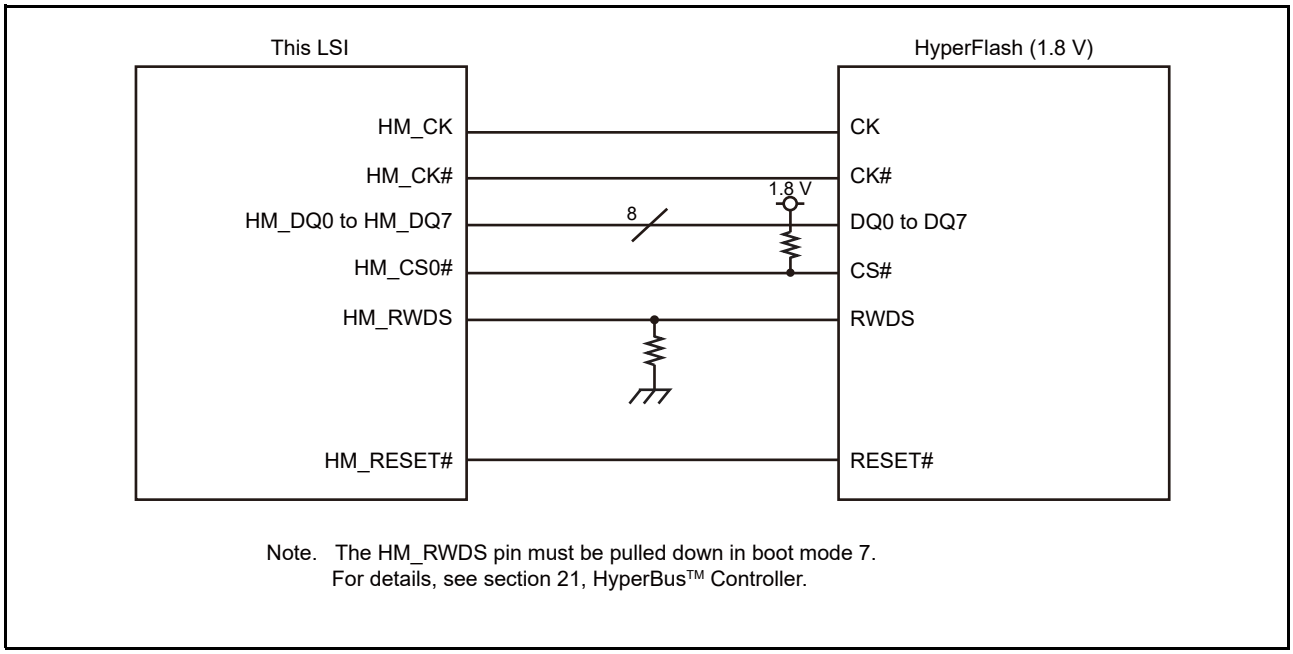


Figure 3.16 Control signal output to HyperFlash by HyperBus protocol conversion

The boot program uses the area at H'8002_0000 to H'8002_3FFF as work memory. It branches to H'3000_0000 (HyperFlash space) at the end of the processing.
Table 3.11 shows the setting values of each peripheral module after executing the boot program, and Figure 3.17 shows an example of connection with HyperFlash.

Table 3.11 Register setting value of each peripheral module in boot mode 7

Peripheral module	Register	Initial values after reset	Setting values after boot program execution
Power-Down Modes	STBCR9	H'FF	H'F7
Clock Pulse Generator	SCLKSEL	H'0000	H'0010
HyperBus controller	IEN	H'0000_0000	H'0000_0000
	MCR0	H'0000_0003	H'0000_0003
	MTR0	H'0000_0001	H'0000_0001
General I/O Port	PHMOM0	3FFF_7FFE	3FFF_7FFE



Note. The HM_RWDS pin must be pulled down in boot mode 7.
For details, see section 21, HyperBus™ Controller.

Figure 3.17 Connection example when using boot mode 7

3.6 Current Program Status Register (CPSR) Setting Value

Table 3.12 shows the setting values of the current program status register (CPSR).

Table 3.12 Current program status register (CPSR) setting value

Bit	Bit name	Initial values after reset	Setting values after boot program execution
7	I	1 (IRQ exception prohibited)	1 (IRQ exception prohibited)
6	F	1 (FIQ exception prohibited)	1 (FIQ exception prohibited)
5	T	0 (Arm state)	0 (Arm state)
[4:0]	M	B’10011 (SVC: Supervisor)	B’10011 (SVC: Supervisor)

3.7 Notes

3.7.1 Boot Related Pins

The initial states and output states in deep standby mode of the pins related to CS0 space memory read, SPI multi I/O bus space memory read, channel 0 of the SD/MMC host interface, Octa memory controller, and the HyperBus controller are different in each boot mode.
For details, refer to section 8, Bus State Controller, section 51, GPIO, and section 52, Power-Down Modes.

3.7.2 Operation when an Exception Occurs with the Exception Vector Set to the High Vector Address

In this LSI, the program counter loops to its own address (exception vector address) in the on-chip ROM if an exception (except for a reset) occurs when the exception vector is set to the high vector address. In any mode from among boot modes 1 to 7, set the V bit in SCTLR to 0 to set the exception vector to the low vector address before an exception (except for a reset) occurs. For the details about the CP15 system control register (SCTLR), refer to the Arm Architecture Reference Manual.

Table 3.13 Self loop address

Occurrence exception	Self loop address
Undefined instruction exception	H'FFFF_0004
Software interrupt exception	H'FFFF_0008
Prefetch abort exception	H'FFFF_000C
Data abort exception	H'FFFF_0010
IRQ exception	H'FFFF_0018
FIQ exception	H'FFFF_001C

3.7.3 Notes on Serial Flash Booting (Boot Mode 3, 4, 6) after This LSI is Reset

In boot mode 3 (booting from a 3.3-V serial flash memory products) or boot mode 4 (Octal-SPI flash boot) or boot mode 6 (OctaFlash boot), read commands (opcode: 03H; address: 3 bytes; dummy cycles: none) are issued to the flash memory. Therefore, if this LSI enters the reset state while the flash memory cannot accept read commands, this LSI may not boot up correctly.
For example, if this LSI is reset while the flash memory is being erased (in the busy state), the flash memory cannot accept read commands.
In such system configuration that the LSI may be reset while the flash memory cannot accept read commands, take appropriate measures to ensure that the flash memory accepts read commands after this LSI is released from the reset state. For example, use the flash memory that has a reset pin or turn off the power to the flash memory when a reset occurs.

4. Secondary Cache

This product incorporates Arm's PL310 as a secondary cache. The IP version is r3p3.

4.1 Features

- Total cache size: 128 Kbytes
- Number of cache ways: 8
- Number of master ports: 2
- Number of slave ports: 2
- Lockdown by master: No
- Lockdown by line: Defined
- Speculative read: No
- Sideband signal from CA9: No
- Support for cache parity error detection: Yes

For details, see CoreLink™ Level 2 Cache Controller L2C-310 Technical Reference Manual issued by Arm Ltd.

4.2 Configuration Signals

The setting values of the configuration signals are shown in Table 4.1.

Table 4.1 Setting Values of Configuration Signals

Configuration Signals	Setting Values
ASSOCIATIVITY*1	1'b0 (8 ways)
CACHEID[5:0]	6'b000000
CFGADDRFILTEN*1	1'b1
CFGADDRFILTEND[11:0]*1	12'h300
CFGADDRFILTSTART[11:0]*1	12'h1F8
CFGBIGEND	1'b0
DATAREADLAT[2:0]*1	3'b000
DATASETUPLAT[2:0]*1	3'b000
DATAWRITELAT[2:0]*1	3'b000
REGFILEBASE[19:0]*1	20'h1F003
TAGREADLAT[2:0]*1	3'b000
TAGSETUPLAT[2:0]*1	3'b000
TAGWRITELAT[2:0]*1	3'b000
WAYSIZE[2:0]*1	3'b001 (16 Kbytes)

Note 1. Do not change the initial settings (setting values) of these signals by software.
Note 2. The base address for the PL310 registers is H'1F003000. For the details and overview of the registers, see CoreLink™ Level 2 Cache Controller L2C-310 Technical Reference Manual issued by Arm Ltd.

5. LSI Internal Bus

5.1 LSI Internal Bus

5.1.1 Configuration

This LSI has two main buses: the north main bus where peripheral modules are connected and the south main bus where on-chip RAM and external ROM and RAM are connected. Figure 5.1 is a schematic diagram of the internal buses.

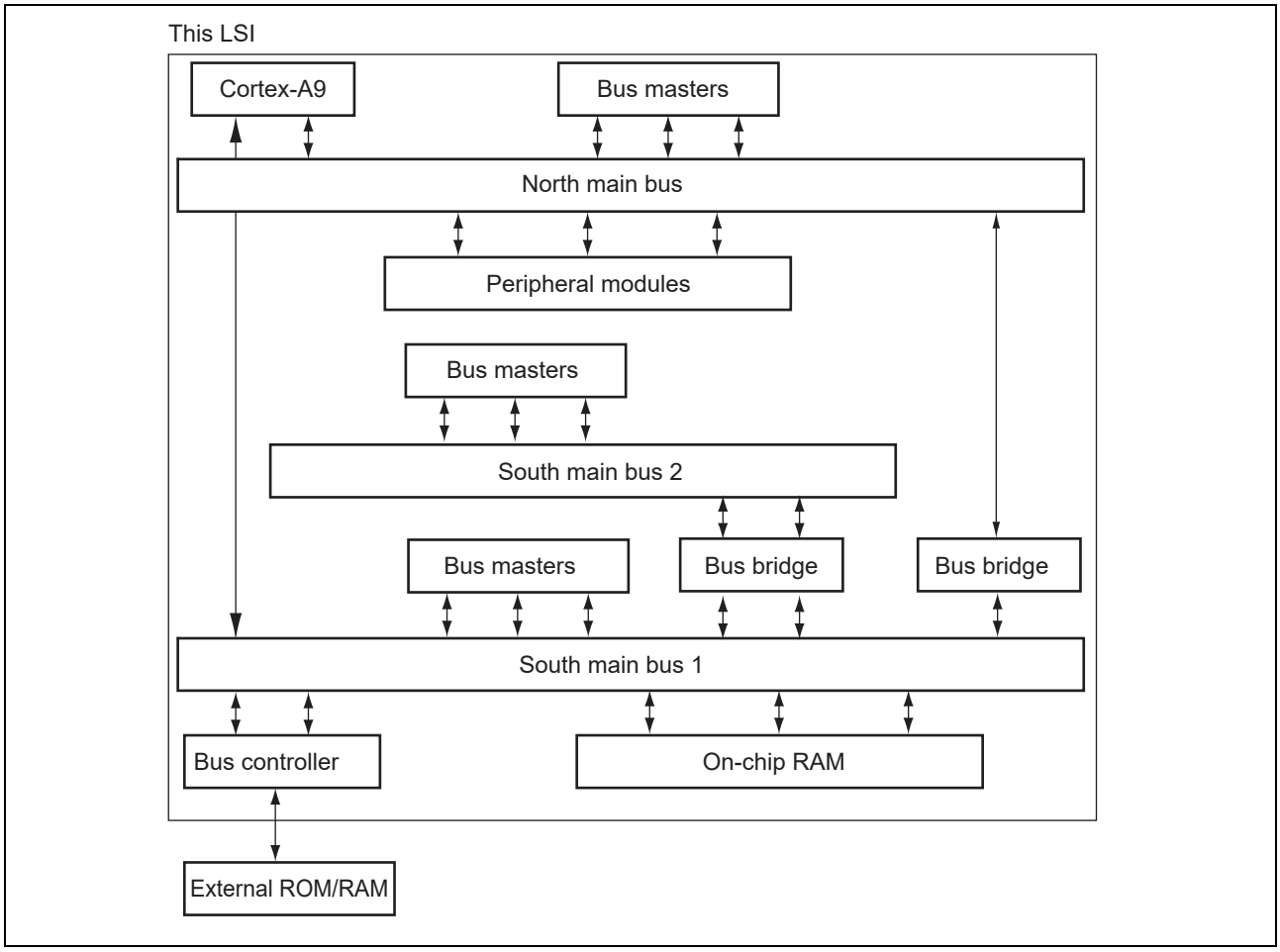


Figure 5.1 Schematic Diagram of LSI Internal Bus

5.1.2 Operation

Cortex-A9 has separate interfaces for the north main bus and south main bus 1. The addresses assigned to the north main bus are accessed through the north main bus interface, and those assigned to the south main bus 1 are accessed through the south main bus 1 interface.

When a bus master connected to the north main bus, except for Cortex-A9, accesses the on-chip RAM or external ROM or RAM, access is executed through the bus bridge for access from the north main bus to the south main bus. The bus masters connected to the south main buses 1 or 2 cannot access an address assigned to the north main bus.

5.2 North Main Bus

5.2.1 Configuration

Various peripheral modules are connected to the north main bus. Figure 5.2 shows the configuration of the north main bus.

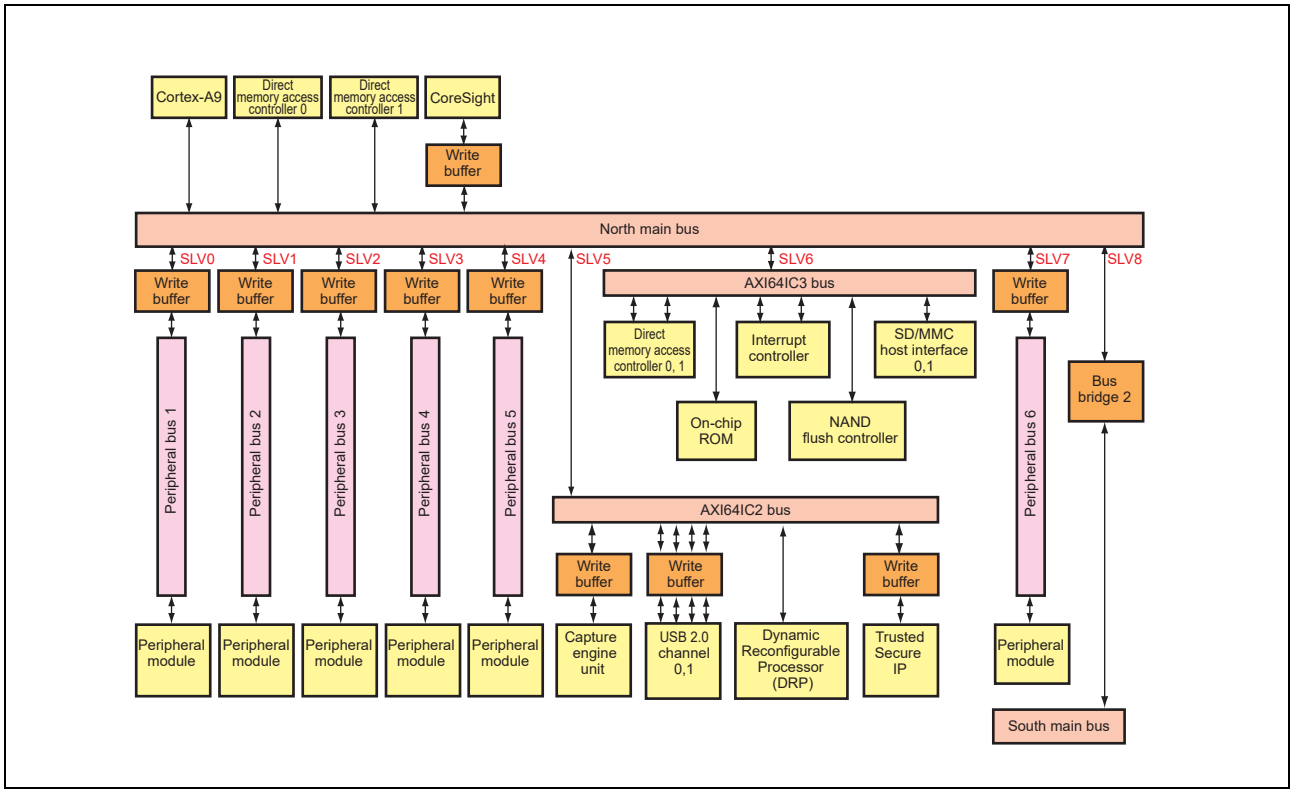


Figure 5.2 North Main Bus Configuration

5.2.2 Features

Table 5.1 shows the features of the north main bus.

Table 5.1 North Main Bus

Item	Description
Bus protocol	AMBA® AXI protocol
Bus system configuration	AXI interconnect with multi-layer configuration for all channels
Bus clock frequency	Bφ
Bus width	64 bits
Arbitration	Round robin

5.2.3 Peripheral Buses

Table 5.2 is a list of the peripheral buses connected to the north main bus.

Table 5.2 List of Peripheral Buses

Item	Description
Peripheral bus 1	
Bus clock frequency	P0φ
Bus width	32 bits
Connected peripheral modules	Realtime clock channels 0 and 1
	Video display controller 6
	Image renderer (IMR-LS2)
Peripheral bus 2	
Bus clock frequency	P0φ
Bus width	32 bits
Connected peripheral modules	Clock pulse generator
	Interrupt controller
	Direct memory access controller units 0 and 1
	General I/O ports
Peripheral bus 3	
Bus clock frequency	P1φ
Bus width	32 bits
Connected peripheral modules	CANFD interface
	Sprite engine
	OS timers 0 to 2
	I^2C bus interfaces 0 to 3
	Renesas SPDIF interface
	Serial sound interfaces 0 to 3
	Multi-function timer pulse unit 3
	General-purpose PWM timer
	Port output enable
	OTP
Peripheral bus 4	
Bus clock frequency	P1φ
Bus width	32 bits
Connected peripheral modules	Serial communication interface with FIFO channels 0 to 4
	Serial communication interface channels 0 and 1
	Renesas serial peripheral interface channels 0 to 2
	A/D converter
	JPEG codec unit
Peripheral bus 5	
Bus clock frequency	Bφ
Bus width	32 bits
Connected peripheral modules	Ethernet MAC controllers 0 and 1
	2D graphic engine
	MIPI CSI-2 interface
	Video input module
AXI64IC2 and AXI64IC3 buses	

Table 5.2 List of Peripheral Buses

Item	Description
Bus protocol	AMBA AXI protocol
Bus system configuration	AXI interconnect with multi-layer configuration for all channels
Bus clock frequency	Bφ
Bus width	64 bits
Arbitration	Round robin
Peripheral bus 6	
Bus clock frequency	P1φ
Bus width	32 bits
Connected peripheral modules	CoreSight

5.3 South Main Bus

5.3.1 Configuration

On-chip RAM and external ROM and RAM are connected to the south main bus. Figure 5.3 shows the configuration of the south main bus.

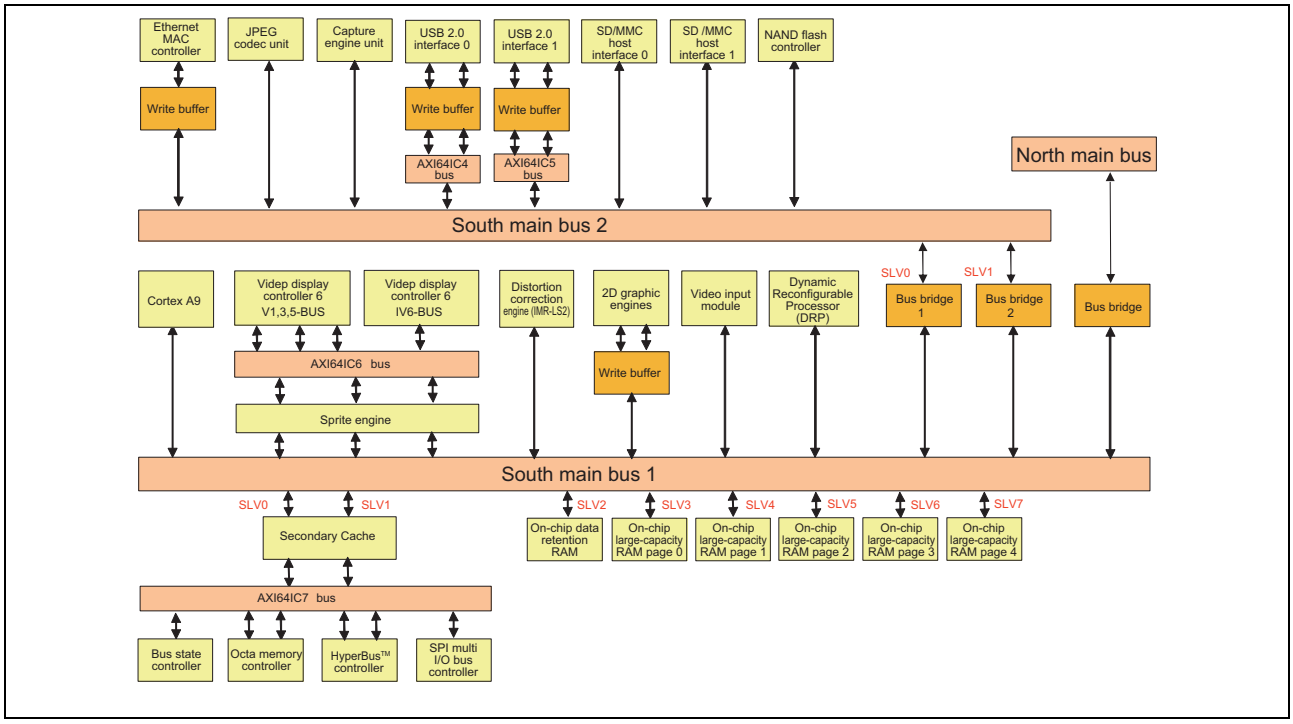


Figure 5.3 Configuration of the South Main Buses

5.3.2 Features

Table 5.3 shows the features of the south main buses.

Table 5.3 South Main Bus

Item	Description
South main bus 1	
Bus protocol	AMBA AXI protocol
Bus system configuration	AXI interconnect with multi-layer configuration for all channels
Bus clock frequency	Bφ
Bus width	128 bits
Arbitration	Round robin
South main bus 2	
Bus protocol	AMBA AXI protocol
Bus system configuration	AXI interconnect with multi-layer configuration for all channels
Bus clock frequency	Bφ
Bus width	64 bits
Arbitration	Round robin

5.3.3 Connected Buses

Table 5.4 is a list of the buses connected to the south main bus and their features.

Table 5.4 List of Buses Connected to South Main Bus and their Features

Item	Description
AXI64IC4, AXI64IC5, AXI64IC6, and AXI64IC7 buses	
Bus protocol	AMBA AXI protocol
Bus system configuration	AXI interconnect with multi-layer configuration for all channels
Bus clock frequency	Bφ
Bus width	64 bits
Arbitration	Round robin

5.4 Address Map

Table 5.5 shows the address map of this LSI.

Table 5.5 Address Map

Address	Area	Slave Area Viewed from North Main Bus Masters	Slave area viewed from the AXI64IC4 bus masters	Slave area viewed from the south main bus masters
H'FFFF_0000 to H'FFFF_FFFF	IO area	SLV6 on-chip ROM	-	-
H'FD00_0000 to H'FFFE_FFFF	Reserved	-		
H'FCFF_0000 to H'FCFF_FFFF	IO area	SLV0		
H'FCFE_0000 to H'FCFE_FFFF	IO area	SLV1		
H'FC04_0000 to H'FCFD_FFFF	Reserved	-		
H'FC00_0000 to H'FC03_FFFF	IO area	SLV7*1*2		
H'F000_2000 to H'FBFF_FFFF	Reserved	-		
H'F000_0000 to H'F000_1FFF	Cortex-A9 private area	-		
H'EB00_0000 to H'EFFF_FFFF	Reserved	-		
H'EA00_0000 to H'EAFF_FFFF	IO area	SLV5		
H'E824_0000 to H'E9FF_FFFF	Reserved	-		
H'E822_0000 to H'E823_FFFF	IO area	SLV6*3		
H'E821_0000 to H'E821_FFFF	IO area	SLV5*4		
H'E820_0000 to H'E820_FFFF	IO area	SLV4		
H'E805_0000 to H'E81F_FFFF	Reserved	-		
H'E802_0000 to H'E804_FFFF	IO area	SLV2		
H'E800_0000 to H'E801_FFFF	IO area	SLV3		
H'E000_0000 to H'E7FF_FFFF	Reserved	-		
H'8040_0000 to H'DFFF_FFFF	Reserved	-		-
H'8030_0000 to H'803F_FFFF	On-chip large-capacity RAM page 4 (1 Mbyte)	SLV8	SLV1	SLV7
H'8020_0000 to H'802F_FFFF	On-chip large-capacity RAM page 3 (1 Mbyte)			SLV6
H'8010_0000 to H'801F_FFFF	On-chip large-capacity RAM page 2 (1 Mbyte)		SLV0	SLV5
H'8008_0000 to H'800F_FFFF	On-chip large-capacity RAM page 1 (512 Kbytes)			SLV4
H'8002_0000 to H'8007_FFFF	On-chip large-capacity RAM page 0 (including on-chip data retention RAM) (512 Kbytes)			SLV3
H'8000_0000 to H'8001_FFFF				SLV2
H'7000_0000 to H'7FFF_FFFF	Reserved	-		-
H'6000_0000 to H'6FFF_FFFF	OctaRAM™ space (256 Mbytes)	SLV8	SLV0	SLV0
H'5000_0000 to H'5FFF_FFFF	OctaFlash™ space (256 Mbytes)			
H'4000_0000 to H'4FFF_FFFF	HyperRAM™ space (256 Mbytes)			
H'3000_0000 to H'3FFF_FFFF	HyperFlash™ space (256 Mbytes)			
H'2000_0000 to H'2FFF_FFFF	SPI multi I/O bus space (256 Mbytes)		SLV1	SLV1
H'1F80_9000 to H'1FFF_FFFF	Reserved	-		-
H'1F80_8000 to H'1F80_8FFF	IO area	SLV8	SLV1	SLV1
H'1F80_1000 to H'1F80_7FFF	Reserved	-		-
H'1F80_0000 to H'1F80_0FFF	IO area	SLV8	SLV1	SLV1
H'1F40_2000 to H'1F7F_FFFF	Reserved	-		-
H'1F40_1000 to H'1F40_1FFF	IO area	SLV8	SLV0	SLV0
H'1F40_0000 to H'1F40_0FFF				

Table 5.5 Address Map

Address	Area	Slave Area Viewed from North Main Bus Masters	Slave area viewed from the AXI64IC4 bus masters	Slave area viewed from the south main bus masters
H'1F00_4000 to H'1F3F_FFFF	Reserved	-		-
H'1F00_3000 to H'1F00_3FFF	IO area	SLV8	SLV0	SLV0
H'1F00_0000 to H'1F00_2FFF				
H'1C00_0000 to H'1EFF_FFFF	Reserved	-		-
H'1800_0000 to H'1BFF_FFFF	Reserved	-	-	-
H'1400_0000 to H'17FF_FFFF	CS5 area (64 Mbytes)	SLV8	SLV0	SLV0
H'1000_0000 to H'13FF_FFFF	CS4 area (64 Mbytes)			
H'0C00_0000 to H'0FFF_FFFF	CS3 area (64 Mbytes)			
H'0800_0000 to H'0BFF_FFFF	CS2 area (64 Mbytes)			
H'0400_0000 to H'07FF_FFFF	CS1 area (64 Mbytes)			
H'0000_0000 to H'03FF_FFFF	CS0 area (64 Mbytes)			

Note 1. Only Cortex-A9 and CoreSight can access this area. If any other north main bus master accesses this area, a decode error will occur.
Note 2. A slave error may occur depending on the CoreSight state.
Note 3. If any address from H'E822_3000 to H'E822_5FFF or from H'E822_C000 to H'E823_FFFF is accessed, a slave error will occur.
Note 4. If any addresses from H'E821_5000 to H'E821_7FFF or from H'E821_C000 to H'E821_FFFF is accessed, a slave error will occur.
Note 5. If the on-chip large-capacity RAM is accessed while access is disabled, a slave error will occur.
Note 6. If the area indicated as " - " is accessed, a decode error or a slave error will occur.
Note 7. I/O areas should be accessed in the size specified for each slave module.

5.5 Address Remapping

5.5.1 Overview

Execution in Cortex-A9 jumps to an exception vector placed in addresses H'0000_0000 to H'0000_001C when an exception such as a reset or an interrupt occurs. The interrupt response time depends on the time to access the memory connected to this area, and when low-speed memory is connected, the overhead is large. To avoid this, the exception vectors can be remapped to the on-chip high-speed RAM by using the MMU or vector base address register, or the address remapping function can be used to allocate the addresses where the exception vectors are placed to the on-chip high-speed RAM.

Figure 5.4 show the address maps before and after address remapping.

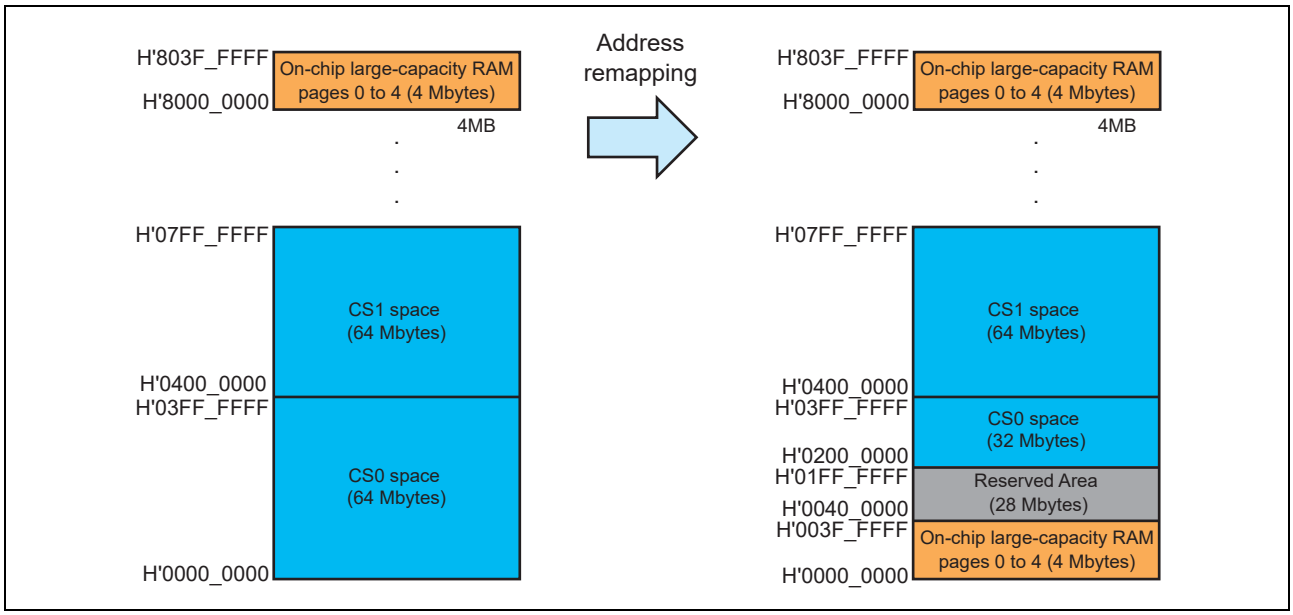


Figure 5.4 Address Remapping

5.5.2 Operation

Addresses are remapped by setting the AXI128 bit in the remap register to 0. After address remapping, pages 0 to 4 of the on-chip large-capacity RAM are allocated to addresses H'0000_0000 to H'003F_FFFF.
During address remapping, access to addresses H'0000_0000 to H'003F_FFFF is prohibited. Accordingly, to modify the remap register, use the following steps.

(1) Stop the bus masters except for Cortex-A9, or make settings so that addresses H'0000_0000 to H'003F_FFFF are never accessed.
(2) Execute a program outside addresses H'0000_0000 to H'003F_FFFF.
(3) After modifying the value of the remap register, execute a dummy read of the remap register.

5.6 AXI Interconnect

5.6.1 Configuration

The AXI interconnect in this LSI has a multi-layer configuration in all channels.
Figure 5.5 shows a conceptual diagram of the AXI interconnect configuration.

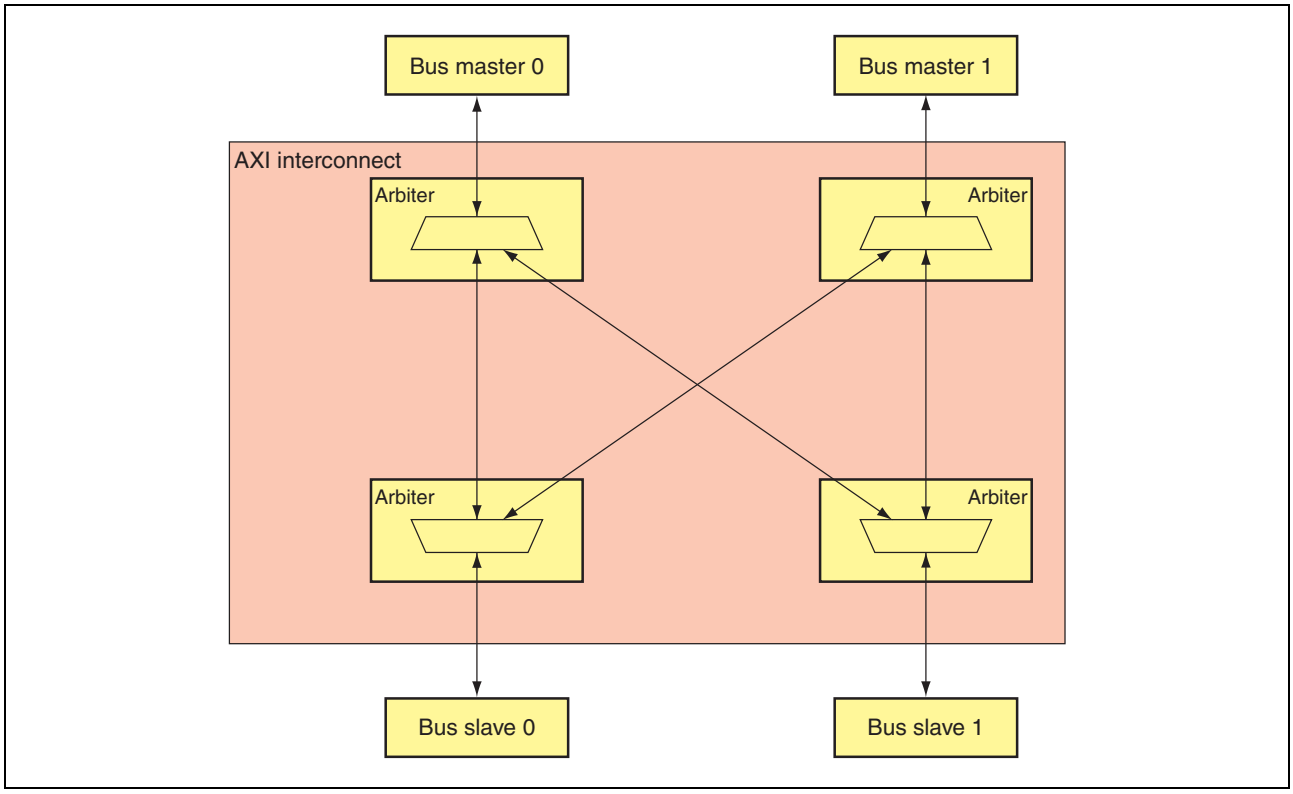


Figure 5.5 Conceptual Diagram of AXI Interconnect Configuration

5.6.2 Operation

In the AXI interconnect, the necessary wiring is prepared for connection between all bus masters and all bus slaves in all channels. When bus masters and slaves access bus slaves and masters, transfer will proceed after bus arbitration by the arbiter. The bus mastership priority changes in a round-robin manner. When multiple bus masters or slaves access different bus slaves or masters, multiple accesses can be executed in parallel. However, when multiple bus masters or slaves access a single bus slave or master at the same time, the bus arbiter executes bus arbitration. When a bus master or slave cannot obtain the bus mastership, it enters a wait state until the bus master or slave that has the bus mastership completes transfer unless the access destination bus master or slave cannot accept multiple transfers. When the destination bus master or slave can accept multiple transfers, bus arbitration is done again with the next transfer timing.

5.7 Bus Bridges

Access from the north main bus to the south main bus and access to the AXI64IC4 bus from the south main bus are executed through a bus bridge. There is only one bus bridge used for access to the south main bus from the north main bus. There are two bus bridges used for access to the south main bus from the AXI64IC4 bus and which bus bridge is used is determined depending on the slave area to be accessed. For assignment of the slave areas to be accessed, see Table 5.5.
Each bus bridge can accept up to eight transfers at the same time. Out-of-order transfer is also supported. Therefore, when access to low-speed external ROM and access to on-chip high-speed RAM from different bus masters occur sequentially in this order, the on-chip RAM access can be done without waiting for completion of the previous external ROM access completion.

5.8 AXI Protocol Control Signals

The AXI protocol control signals can be set as desired for each bus master. For details of the AXI protocol control signals, refer to the AMBA AXI Protocol Specification prepared by Arm Ltd.

5.8.1 Bus Masters other than Cortex-A9, CoreSight, and the Direct Memory Access Controller

(1) Cache control signals (ARCACHE[3:0], AWCACHE[3:0])

Use the AXI bus control register (AXIBUSCTL) to make settings of the ARCACHE[3:0] and AWCACHE[3:0] signals for each bus master. Be sure to make settings while the target bus master does not use the AXI bus.

(2) Response signals (RRESP[1:0], BRESP[1:0])

Use the AXI bus response error status register (AXIRERRST) to read the RRESP[1:0] and BRESP[1:0] signals received by each bus master. The register value is updated when a response error occurs. The status register value can be cleared to 00 through the AXI bus response error clear register (AXIRERRCLR).
In addition, enabling interrupts through the AXI bus response error interrupt control register (AXIRERRCTL) allows an interrupt to be generated when a response error occurs.
This interrupt should be used only for debugging purposes. Make sure that no response error occurs during system operation.

(3) Protection unit information (ARPROT[2:0], AWPROT[2:0])

Use the master access control register (MSTACCCTL) to make settings of the ARPROT[1] and AWPROT[1] signals for each bus master. Be sure to make settings while the target bus master does not use the AXI bus. Note that signals ARPROT[2], ARPROT[0], AWPROT[2], and AWPROT[0] are fixed as follows and cannot be modified.
ARPROT[2], AWPROT[2]: 0 (data access)
ARPROT[0], AWPROT[0]: 0 (normal access)

(4) Atomic access (ARLOCK[1:0], AWLOCK[1:0])

Signals ARLOCK[1:0] and AWLOCK[1:0] are fixed as follows and cannot be modified.
ARLOCK[1:0], AWLOCK[1:0]: 00 (normal access)

5.8.2 Cortex-A9

For details on the Cortex-A9, refer to the Arm Architecture Reference Manual.

5.8.3 CoreSight

For details on CoreSight, refer to the technical reference manual issued by Arm Ltd.
The bus master side (AHB access port) of CoreSight is connected to the main bus via the AHB-AXI bus conversion circuit.
The signals are converted as follows for connection to the AXI bus.

(1) Cache control (ARCACHE[3:0], AWCACHE[3:0])

ARCACHE[3], AWCACHE[3]: 0 when HPROT[3] is 0, 1 when HPROT[3] is 1.
ARCACHE[2], AWCACHE[2]: 0 when HPROT[3] is 0, 1 when HPROT[3] is 1.
ARCACHE[1], AWCACHE[1]: Value of HPROT[3] (cacheable)
ARCACHE[0], AWCACHE[0]: Value of HPROT[2] (bufferable)

(2) Response unit (RRESP[1:0], BRESP[1:0])

OKAY is returned when RRESP[1:0] and BRESP[1:0] are 00 or 01.
ERROR is returned when RRESP[1:0] and BRESP[1:0] are 10 or 11.

(3) Protection unit information (ARPROT[2:0], AWPROT[2:0])

ARPROT[2], AWPROT[2]: Inverse of HPROT[0] (data/opcode)
ARPROT[1], AWPROT[1]: Value of HPROT[6] (nonsecure/secure)
ARPROT[0], AWPROT[0]: Value of HPROT[1] (privileged)

(4) Atomic access (ARLOCK[1:0], AWLOCK[1:0])

ARLOCK[1:0], AWLOCK[1:0]: Fixed to 00 (normal access)

5.8.4 Direct Memory Access Controller

For details on the direct memory access controller, refer to section 9, Direct Memory Access Controller.

5.8.5 Slave Area

The control signals are handled as follows by the modules in the slave area.

(1) Cache control (ARCACHE[3:0], AWCACHE[3:0])

The secondary cache and write buffer refer to these signals. Other modules in the slave area do not refer to them.

(2) Response unit (RRESP[1:0], BRESP[1:0])

See Table 5.5.

(3) Protection unit information (ARPROT[2:0], AWPROT[2:0])

ARPROT[2], AWPROT[2] (instruction/data): The modules in the slave area do not refer to these signals.
ARPROT[1], AWPROT[1] (non-secure/secure): The interrupt controller and secondary cache refer to these signals. For details, refer to the technical reference manual issued by Arm Ltd.
The slave access control registers (SLVACCCTL0 to SLVACCCTL4) are available to specify whether secure access to some modules* in the slave area other than the interrupt controller and secondary cache is only allowed or not. When the respective bit in the slave access controlling register is set to 0 (enabling only secure access), secure access (access with ARPROT[1] or AWPROT[1] is set to 0) is only allowed and non-secure access (access with ARPROT[1] or AWPROT[1] is set to 1) is not allowed. If a non-secure access is attempted when the given bit is set to 0, ERROR (slave error or decode error) is returned.
ARPROT[0], AWPROT[0] (privileged/user): The modules in the slave area do not refer to these signals.

Note: • Secure access is only allowed for the following modules in the slave area; direct memory access controller unit 0, realtime clock channel 0, on-chip ROM, internal bus (address range from H'FCFE_9000 to H'FCFE_97FF), and OTP.
• Secure access or non-secure access is allowed for the following modules in the slave area; direct memory access controller unit 1, realtime clock channel 1, and internal bus (address range from H'FCFE_8000 to H'FCFE_87FF).

(4) Atomic access (ARLOCK[1:0], AWLOCK[1:0])

This LSI does not support atomic access. Signals ARLOCK[1:0] and AWLOCK[1:0] should be fixed to 00 for normal access by the bus master.*

Note: * This restriction means that instructions for exclusive access (LDREX, STREX, LDREXB, STREXB, LDREXD, STREXD, LDREXH, STREXH) and semaphore instructions (SWP, SWPB) cannot be used by the Cortex-A9 in the internal non-cacheable areas.

5.9 Write Buffers

A write buffer is provided at each connection between the north main bus and a peripheral bus and at each connection between the AXI64IC2 bus and some slave modules.
When the write buffer proceeds writing with the AWCACHE[1:0] cache control signals being set to cache-enabled or buffer-enabled (either of the AWCACHE[1:0] signals is set to 1), it sends a write completion response to the bus master before accessing the slave area under the write buffer.
At this time, even if a slave error response is returned from the slave area to be accessed, it is ignored. However, reception of a slave error response during writing can be notified to the interrupt controller by the interrupt signal.
Note that the write buffers connected between the AXI64IC2 bus and slave modules do not receive the slave error response, so the interrupt signal is not generated.
A write buffer is also provided at each connection between some bus master modules and the north main bus and south main buses 1 and 2. These write buffers always return the OKAY response to the bus master modules regardless of the settings of the cache control signals AWCACHE[1:0] in response to the decode error or slave error from the modules in the slave area during writing. However, reception of an error response during writing can also be notified to the interrupt controller by the interrupt signal. For details on the interrupt signals for the write buffers, see section 5.12, Interrupt Request.

5.10 Control Over the Order of Access to the Memory and I/O Areas

In access to the memory and I/O areas, the order with which write instructions appear in the software and the order in which the details of the actual instructions are reflected do not necessarily match. This depends on factors such as the settings for memory attributes, bus configuration, write-back, and so on.
In Figure 5.2, for example, in a case where writing to the register of a peripheral module connected to peripheral bus 2 after writing to the register of a peripheral modules connected to peripheral bus 1 is required, preserving the desired order in which instructions are written in the software may not be possible when the I/O area for peripheral modules connected to peripheral bus 1 has the bufferable attribute. This is because the buffer has to return a response signal before actual access to peripheral bus 1.
Furthermore, even in the case of settings where caching is disabled for the I/O area for peripheral modules connected to peripheral bus 1 and the area has the non-bufferable attribute, preserving the order may still not be possible in cases where actually reflecting the written value in the peripheral module takes a long enough time.
Accordingly, in cases requiring control over the order of processing, add processing to set suitable attributes for memory and, as required, to dummy-read the same register after writing to it.

5.11 Register Descriptions

Table 5.6 shows the registers related to the internal bus.

Table 5.6 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
BSID register	BSID	R	H'083B6447	H'FCFE_8004	32
Remap register	RMPR	R/W	H'0000_0001	H'FCFE_8400	32
AXI bus control register 0	AXIBUSCTL0	R/W	H'0000_0000	H'FCFE_8404	32
AXI bus control register 1	AXIBUSCTL1	R/W	H'0000_0000	H'FCFE_8408	32
AXI bus control register 2	AXIBUSCTL2	R/W	H'0000_0000	H'FCFE_840C	32
AXI bus control register 3	AXIBUSCTL3	R/W	H'0000_0000	H'FCFE_8410	32
AXI bus control register 4	AXIBUSCTL4	R/W	H'0000_0000	H'FCFE_8414	32
AXI bus control register 5	AXIBUSCTL5	R/W	H'0000_0000	H'FCFE_8418	32
AXI bus control register 6	AXIBUSCTL6	R/W	H'0000_0000	H'FCFE_841C	32
AXI bus control register 7	AXIBUSCTL7	R/W	H'0000_0000	H'FCFE_8420	32
AXI bus response error interrupt control register 0	AXIRERRCTL0	R/W	H'0000_0000	H'FCFE_8430	32
AXI bus response error interrupt control register 1	AXIRERRCTL1	R/W	H'0000_0000	H'FCFE_8434	32
AXI bus response error interrupt control register 2	AXIRERRCTL2	R/W	H'0000_0000	H'FCFE_8438	32
AXI bus response error status register 0	AXIRERRST0	R/W	H'0000_0000	H'FCFE_8440	32
AXI bus response error status register 1	AXIRERRST1	R/W	H'0000_0000	H'FCFE_8444	32
AXI bus response error status register 2	AXIRERRST2	R/W	H'0000_0000	H'FCFE_8448	32
AXI bus response error status register 3	AXIRERRST3	R/W	H'0000_0000	H'FCFE_844C	32
AXI bus response error clear register 0	AXIRERRCLR0	R/W	H'0000_0000	H'FCFE_8450	32
AXI bus response error clear register 1	AXIRERRCLR1	R/W	H'0000_0000	H'FCFE_8454	32
AXI bus response error clear register 2	AXIRERRCLR2	R/W	H'0000_0000	H'FCFE_8458	32
Master access control register 0	MSTACCCTL0	R/W	H'2202_2222	H'FCFE_9000	32
Master access control register 1	MSTACCCTL1	R/W	H'2202_2222	H'FCFE_9004	32
Master access control register 2	MSTACCCTL2	R/W	H'2222_0202	H'FCFE_9008	32
Master access control register 3	MSTACCCTL3	R/W	H'2220_0020	H'FCFE_900C	32
Master access control register 4	MSTACCCTL4	R/W	H'0202_0202	H'FCFE_9010	32
Slave access control register 0	SLVACCCTL0	R/W	H'4555_1154	H'FCFE_9020	32
Slave access control register 1	SLVACCCTL1	R/W	H'5155_4555	H'FCFE_9024	32
Slave access control register 2	SLVACCCTL2	R/W	H'5555_5550	H'FCFE_9028	32
Slave access control register 3	SLVACCCTL3	R/W	H'1550_0000	H'FCFE_902C	32
Slave access control register 4	SLVACCCTL4	R/W	H'5555_5500	H'FCFE_9030	32

5.11.1 BSID Register (BSID)

This register indicates the same ID number as the ID register (BSID) of the debugger interface.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DID[31:16]															
Initial Value :	0	0	0	0	1	0	0	0	0	0	1	1	1	0	1	1
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DID[15:0]															
Initial Value :	0	1	1	0	0	1	0	0	0	1	0	0	0	1	1	1
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DID[31:0]	H’083B6447	R	Device ID These bits indicate the device ID of the RZ/A2 series products. Note that the higher-order four bits may be changed according to the chip version.

5.11.2 Remap Register (RMPR)

This register controls the address remapping function.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AXI 128
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	AXI128	1	R/W	AXI128 Address Remapping This bit enables or disables allocation of addresses H’0000_0000 to H'003F_FFFF to on-chip RAM pages 0 to 4. 0: Address remapping is enabled. 1: Address remapping is disabled. Note: The addresses H'0040_0000 to H'01FF_FFFF are reserved areas.

5.11.3 AXI Bus Control Register 0 (AXIBUSCTL0)

This register controls the cache operation for the JPEG codec unit and Ethernet MAC controller.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	JCUARCACHE[3:0]				—	—	—	—	JCUAWCACHE[3:0]			
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ETHAXCACHE [1:0]	
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	JCU ARCACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for JPEG Codec Unit These bits specify the system cache operation when the JPEG codec unit performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the JPEG codec unit. Modify the values of these bits only while the JPEG codec unit does not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	JCU AWCACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for JPEG Codec Unit These bits specify the system cache operation when the JPEG codec unit performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the JPEG codec unit. Modify the values of these bits only while the JPEG codec unit does not use the internal bus.
15 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	ETH AXCACHE [1:0]	All 0	R/W	AWCACHE[3:0] and ARCACHE[3:0] Signals for Ethernet MAC Controller These bits specify the system cache operation when the Ethernet MAC controller performs read/write access. The values of these bits are used as the AWCACHE[3:0] and ARCACHE[3:0] signals for the Ethernet MAC controller. The value of the ETHAXCACHE[0] is used as the ARCACHE[0] and AWCACHE[0] signals. When ETHAXCACHE[1] is set to 0, ARCACHE[3:1] and AWCACHE[3:1] are all 0. When ETHAXCACHE[1] is set to 1, ARCACHE[3:1] and AWCACHE[3:1] are all 1. Modify the values of these bits only while the Ethernet MAC controller does not use the internal bus.

5.11.4 AXI Bus Control Register 1 (AXIBUSCTL1)

This register controls the cache operation for the distortion correction engine (IMR-LS2), and MIPI CSI-2 interface.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	IMR20ARCACHE[3:0]				—	—	—	—	IMR20AWCACHE[3:0]			
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	VINAWCACHE[3:0]			
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	IMR20 ARCACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Image Renderer (IMR-LS2) These bits specify the system cache operation when image renderer (IMR-LS2) performs read access. The values of these bits are used as the ARCACHE[3:0] signals for image renderer (IMR-LS2). Modify the values of these bits only while image renderer (IMR-LS2) does not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	IMR20 AWCACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Image Renderer (IMR-LS2) These bits specify the system cache operation when image renderer (IMR-LS2) performs write access. The values of these bits are used as the AWCACHE[3:0] signals for image renderer (IMR-LS2). Modify the values of these bits only while image renderer (IMR-LS2) does not use the internal bus.
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	VIN AWCACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Video Input Module These bits specify the system cache operation when the video input module performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the video input module. Modify the values of these bits only while the video input module does not use the internal bus.

5.11.5 AXI Bus Control Register 2 (AXIBUSCTL2)

This register controls the cache operation for the capture engine unit.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CEUAWCACHE[3:0]			
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	CEU AWCACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Capture Engine Unit These bits specify the system cache operation when capture engine unit performs write access. The values of these bits are used as the AWCACHE[3:0] signals for capture engine unit. Modify the values of these bits only while capture engine unit does not use the internal bus.

5.11.6 AXI Bus Control Register 3 (AXIBUSCTL3)

This register controls the cache operation for the SD/MMC host interface.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	SDMMC0ARCACHE[3:0]				—	—	—	—	SDMMC0AWCACHE[3:0]			
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SDMMC1ARCACHE[3:0]				—	—	—	—	SDMMC1AWCACHE[3:0]			
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	SDMMC0 ARCACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for SD/MMC Host Interface 0 These bits specify the system cache operation when the SD/MMC host interface 0 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the SD/MMC host interface 0. Modify the values of these bits only while the SD/MMC host interface 0 does not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	SDMMC0 AWCACHE [3:0]	All 0	R/W	AWCACHE[3:0] Signals for SD/MMC Host Interface 0 These bits specify the system cache operation when the SD/MMC host interface 0 performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the SD/MMC host interface 0. Modify the values of these bits only while the SD/MMC host interface 0 does not use the internal bus.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	SDMMC1 ARCACHE [3:0]	All 0	R/W	ARCACHE[3:0] Signals for SD/MMC Host Interface 1 These bits specify the system cache operation when the SD/MMC Host Interface 1 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the SD/MMC host interface 1. Modify the values of these bits only while the SD/MMC host interface 1 does not use the internal bus.
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	SDMMC1 AWCACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for SD/MMC Host Interface 1 These bits specify the system cache operation when the SD/MMC host interface 1 performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the SD/MMC host interface 1. Modify the values of these bits only while the SD/MMC host interface 1 does not use the internal bus.

5.11.7 AXI Bus Control Register 4 (AXIBUSCTL4)

This register controls the cache operation for the NAND flash controller and dynamic reconfigurable processor (DRP).

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	NANDARCACHE[3:0]				—	—	—	—	NANDAWCACHE[3:0]			
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DRPARCACHE[3:0]				—	—	—	—	DRPAWCACHE[3:0]			
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	NAND ARCACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for NAND flash controller These bits specify the system cache operation when the NAND flash controller performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the NAND flash controller. Modify the values of these bits only while the NAND flash controller does not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	NAND AWCACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for NAND flash controller These bits specify the system cache operation when the NAND flash controller performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the NAND flash controller. Modify the values of these bits only while the NAND flash controller does not use the internal bus.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	DRP ARCACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for DRP These bits specify the system cache operation when the DRP performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the DRP. Modify the values of these bits only while the DRP does not use the internal bus. The setting of these bits is only effective for products with a DRP. For products without a DRP, always write 0 to these bits.
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	DRP AWCACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for DRP These bits specify the system cache operation when the DRP performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the DRP. Modify the values of these bits only while the DRP does not use the internal bus. The setting of these bits is only effective for products with a DRP. For products without a DRP, always write 0 to these bits.

5.11.8 AXI Bus Control Register 5 (AXIBUSCTL5)

This register controls the cache operation for the 2D Drawing Engine.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	D2D0AXCACHE [1:0]	
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	D2D1AXCACHE [1:0]	
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	D2D0 AXCACHE [1:0]	All 0	R/W	AWCACHE[3:0] and ARCACHE[3:0] Signals for 2D Drawing Engine 0 These bits specify the system cache operation when the 2D drawing engine 0 performs read or write access. The values of these bits are used as the AWCACHE[3:0] and ARCACHE[3:0] signals for the media local bus. The D2D0AXCACHE[0] value is used as ARCACHE[0] and AWCACHE[0] without change. When D2D0AXCACHE[1] = 0, ARCACHE[3:1] and AWCACHE[3:1] signals are all set to 0. When D2D0AXCACHE[1] = 1, ARCACHE[3:1] and AWCACHE[3:1] signals are all set to 1. Modify the values of these bits only while the 2D drawing engine 0 does not use the internal bus.
15 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	D2D1 AXCACHE [1:0]	All 0	R/W	AWCACHE[3:0] and ARCACHE[3:0] Signals for 2D Drawing Engine 1 These bits specify the system cache operation when the 2D drawing engine 1 performs read or write access. The values of these bits are used as the AWCACHE[3:0] and ARCACHE[3:0] signals for the media local bus. The D2D1AXCACHE[0] value is used as ARCACHE[0] and AWCACHE[0] without change. When D2D1AXCACHE[1] = 0, ARCACHE[3:1] and AWCACHE[3:1] signals are all set to 0. When D2D1AXCACHE[1] = 1, ARCACHE[3:1] and AWCACHE[3:1] signals are all set to 1. Modify the values of these bits only while the 2D drawing engine 1 does not use the internal bus.

5.11.9 AXI Bus Control Register 6 (AXIBUSCTL6)

This register controls the cache operation for video display controller 6.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VDC601ARCACHE[3:0]				—	—	—	—	VDC601AWCACHE[3:0]			
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VDC602ARCACHE[3:0]				—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	VDC601 ARCACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 6 IV3-BUS and Sprite Engine (RLED) These bits specify the system cache operation when the IV3-BUS in video display controller 6, or Sprite Engine (RLED) performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV3-BUS in video display controller 6, and Sprite Engine (RLED). Modify the values of these bits only while the video display controller 6 and Sprite Engine (RLED) do not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	VDC601 AWCACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Video Display Controller 6 IV1-BUS These bits specify the system cache operation when the IV1-BUS in the video display controller 6 performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the IV1-BUS in video display controller 6. Modify the values of these bits only while the video display controller 6 does not use the internal bus.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	VDC602 ARCACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 6 IV5-BUS and Sprite Engine (SU0) These bits specify the system cache operation when the IV5-BUS in the video display controller 6, or Sprite Engine (SU0) performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV5-BUS in video display controller 6, and Sprite Engine (SU0). Modify the values of these bits only while the video display controller 6 and Sprite Engine (SU0) do not use the internal bus.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.10 AXI Bus Control Register 7 (AXIBUSCTL7)

This register controls the cache operation for video display controller 6.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VDC604ARCACHE[3:0]				—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	VDC604 ARCACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 6 IV6-BUS and Sprite Engine (SU1) These bits specify the system cache operation when the IV6-BUS in the video display controller 6, or Sprite Engine (SU1) performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV6-BUS in video display controller 6, and Sprite Engine (SU1). Modify the values of these bits only while the video display controller 6 and Sprite Engine (SU1) do not use the internal bus.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.11 AXI Bus Response Error Interrupt Control Register 0 (AXIRERRCTL0)

This register controls AXI bus response error interrupts.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	JCUR ERREN	—	—	—	—	—	—	—	IMR20R ERREN	—	—	—	VINR ERREN
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CEUR ERREN	—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	JCU RERREN	0	R/W	Response Error Interrupt Enable for JPEG Codec Unit Enables or disables interrupt requests when access from the JPEG codec unit generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
27 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	IMR20 RERREN	0	R/W	Response Error Interrupt Enable for Image Renderer (IMR-LS2) Enables or disables interrupt requests when access from image renderer (IMR-LS2) generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	VIN RERREN	0	R/W	Response Error Interrupt Enable for Video Input Module Enables or disables interrupt requests when access from the video input module generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	CEU RERREN	0	R/W	Response Error Interrupt Enable for Capture Engine Unit Enables or disables interrupt requests when access from capture engine unit generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.12 AXI Bus Response Error Interrupt Control Register 1 (AXIRERRCTL1)

This register controls AXI bus response error interrupts.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SDMMC0 RERREN	—	—	—	SDMMC1 RERREN	—	—	—	NAND RERREN	—	—	—	DRP RERREN
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	SDMMC0 RERREN	0	R/W	Response Error Interrupt Enable for SD/MMC Host Interface 0 Enables or disables interrupt requests when access from the SD/MMC host interface 0 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	SDMMC1 RERREN	0	R/W	Response Error Interrupt Enable for SD/MMC Host Interface 1 Enables or disables interrupt requests when access from the SD/MMC host interface 1 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	NAND RERREN	0	R/W	Response Error Interrupt Enable for NAND flash controller Enables or disables interrupt requests when access from the NAND flash controller generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	DRP RERREN	0	R/W	Response Error Interrupt Enable for DRP Enables or disables interrupt requests when access from the DRP generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled. The setting of this bit is only effective for products with a DRP. For products without a DRP, always write 0 to this bit.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.13 AXI Bus Response Error Interrupt Control Register 2 (AXIRERRCTL2)

This register controls AXI bus response error interrupts.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	VDC601 RERREN	—	—	—	VDC602 RERREN	—	—	—	—	—	—	—	VDC604 RERREN
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	VDC601 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 6 IV1/3-BUS and Sprite Engine (RLED) Enables or disables interrupt requests when access from the IV1/3-BUS in video display controller 6, or the sprite engine (RLED) generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	VDC602 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 6 IV5-BUS and Sprite Engine (SU0) Enables or disables interrupt requests when access from the IV5-BUS in video display controller 6, or the sprite engine (SU0) generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	VDC604 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 6 IV6-BUS and Sprite Engine (SU1) Enables or disables interrupt requests when access from the IV6-BUS in video display controller 6, or the sprite engine (SU1) generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.14 AXI Bus Response Error Status Register 0 (AXIRERRST0)

This register indicates occurrence of AXI bus response errors.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	JCURRESP [1:0]		JCUBRESP [1:0]		—	—	—	—	IMR20RRESP [1:0]		IMR20BRESP [1:0]		—	—	VINBRESP [1:0]	
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CEUBRESP [1:0]		—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	JCU RRESP [1:0]	00	R	RRESP[1:0] Signals for JPEG Codec Unit These bits indicate the RRESP[1:0] signals received by the JPEG codec unit. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
29, 28	JCU BRESP [1:0]	00	R	BRESP[1:0] Signals for JPEG Codec Unit These bits indicate the BRESP[1:0] signals received by the JPEG codec unit. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
27 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23, 22	IMR20 RRESP [1:0]	00	R	RRESP[1:0] Signals for Image Renderer (IMR-LS2) These bits indicate the RRESP[1:0] signals received by image renderer (IMR-LS2). The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
21, 20	IMR20 BRESP [1:0]	00	R	BRESP[1:0] Signals for Image Renderer (IMR-LS2) These bits indicate the BRESP[1:0] signals received by image renderer (IMR-LS2). The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	VIN BRESP [1:0]	00	R	BRESP[1:0] Signals for Video Input Module These bits indicate the BRESP[1:0] signals received by the video input module. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	CEU BRESP [1:0]	00	R	BRESP[1:0] Signals for Capture Engine Unit These bits indicate the BRESP[1:0] signals received by capture engine unit. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.15 AXI Bus Response Error Status Register 1 (AXIRERRST1)

This register indicates occurrence of AXI bus response errors.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMMC0 RRESP[1:0]		SDMMC0 BRESP[1:0]		SDMMC1 RRESP[1:0]		SDMMC1 BRESP[1:0]		NAND RRESP[1:0]		NAND BRESP[1:0]		DRP RRESP[1:0]		DRP BRESP[1:0]	
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	SDMMC0 RRESP [1:0]	00	R	RRESP[1:0] Signals for SD/MMC Host Interface 0 These bits indicate the RRESP[1:0] signals received by the SD/MMC Host Interface 0. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
29, 28	SDMMC0 BRESP [1:0]	00	R	BRESP[1:0] Signals for SD/MMC Host Interface 0 These bits indicate the BRESP[1:0] signals received by the SD/MMC Host Interface 0. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
27, 26	SDMMC1 RRESP [1:0]	00	R	RRESP[1:0] Signals for SD/MMC Host Interface 1 These bits indicate the RRESP[1:0] signals received by the SD/MMC Host Interface 1. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
25, 24	SDMMC1 BRESP [1:0]	00	R	BRESP[1:0] Signals for SD/MMC Host Interface 1 These bits indicate the BRESP[1:0] signals received by the SD/MMC Host Interface 1. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
23, 22	NAND RRESP [1:0]	00	R	RRESP[1:0] Signals for NAND flash Controller These bits indicate the RRESP[1:0] signals received by the NAND flash controller. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
21, 20	NAND BRESP [1:0]	00	R	BRESP[1:0] Signals for NAND flash Controller These bits indicate the BRESP[1:0] signals received by the NAND flash controller. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
19, 18	DRP RRESP [1:0]	00	R	RRESP[1:0] Signals for DRP These bits indicate the RRESP[1:0] signals received by the DRP. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR Note that these bits are always read as 0 in products without a DRP.

Bit	Bit Name	Initial Value	R/W	Description
17, 16	DRP BRESP [1:0]	00	R	BRESP[1:0] Signals for DRP These bits indicate the BRESP[1:0] signals received by the DRP. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR Note that these bits are always read as 0 in products without a DRP.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.16 AXI Bus Response Error Status Register 2 (AXIRERRST2)

This register indicates occurrence of AXI bus response errors.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VDC601 RRESP[1:0]		VDC601 BRESP[1:0]		VDC602 RRESP[1:0]		—	—	—	—	—	—	VDC604 RRESP[1:0]		—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	VDC601 RRESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 6 Channel 0 IV3-BUS and Sprite Engine (RLED) These bits indicate the RRESP[1:0] signals received by the IV3-BUS in channel 0 of video display controller 6 and the sprite engine (RLED). The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
29, 28	VDC601 BRESP [1:0]	00	R	BRESP[1:0] Signals for Video Display Controller 6 Channel 0 IV1-BUS These bits indicate the BRESP[1:0] signals received by the IV1-BUS in channel 0 of video display controller 6. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
27, 26	VDC602 RRESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 6 Channel 0 IV5-BUS and Sprite Engine (SU0) These bits indicate the RRESP[1:0] signals received by the IV3-BUS in channel 0 of video display controller 6 and the sprite engine (SU0). The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
25 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19, 18	VDC604 RRESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 6 Channel 0 IV6-BUS and Sprite Engine (SU1) These bits indicate the RRESP[1:0] signals received by the IV6-BUS in channel 0 of video display controller 6 and the sprite engine (SU1). The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
17 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.17 AXI Bus Response Error Clear Register 0 (AXIRERRCLR0)

This register clears the AXI bus response error status.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	JCURRESPCLR	—	JCUBRESPCLR	—	—	—	—	—	IMR20RRESPCL	—	IMR20BRESPCL	—	—	—	VINBRESPCLR
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W	R	R	R	R	R	R/W	R	R/W	R	R	R	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SERRESPCLR	—	SERBRESPCLR	—	—	—	CEUBRESPCLR	—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	JCU RRESP CLR	0	R/W	JCURRESP[1:0] Clear Writing 1 to this bit clears the JCURRESP[1:0] bits to 00. This bit is always read as 0.
29	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
28	JCU BRESP CLR	0	R/W	JCUBRESP[1:0] Clear Writing 1 to this bit clears the JCUBRESP[1:0] bits to 00. This bit is always read as 0.
27 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22	IMR20 RRESP CLR	0	R/W	IMR20RRESP[1:0] Clear Writing 1 to this bit clears the IMR20RRESP[1:0] bits to 00. This bit is always read as 0.
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20	IMR20 BRESP CLR	0	R/W	IMR20BRESP[1:0] Clear Writing 1 to this bit clears the IMR20BRESP[1:0] bits to 00. This bit is always read as 0.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	VIN BRESP CLR	0	R/W	VINBRESP[1:0] Clear Writing 1 to this bit clears the VINBRESP[1:0] bits to 00. This bit is always read as 0.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	SER RRESP CLR	0	R/W	SERRRESP[1:0] Clear Writing 1 to this bit clears the SERRRESP[1:0] bits to 00. This bit is always read as 0.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	SER BRESP CLR	0	R/W	SERBRESP[1:0] Clear Writing 1 to this bit clears the SERBRESP[1:0] bits to 00. This bit is always read as 0.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	CEU BRESP CLR	0	R/W	CEUBRESP[1:0] Clear Writing 1 to this bit clears the CEUBRESP[1:0] bits to 00. This bit is always read as 0.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.18 AXI Bus Response Error Clear Register 1 (AXIRERRCLR1)

This register clears the AXI bus response error status.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	SDMMC0 RRESP CLR	—	SDMMC0 BRESP CLR	—	SDMMC1 RRESP CLR	—	SDMMC1 BRESP CLR	—	NAND RRESP CLR	—	NAND BRESP CLR	—	DRP RRESP CLR	—	DRP BRESP CLR
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	SDMMC0 RRESP CLR	0	R/W	SDMMC0RRESP[1:0] Clear Writing 1 to this bit clears the SDMMC0RRESP[1:0] bits to 00. This bit is always read as 0.
29	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
28	SDMMC0 BRESP CLR	0	R/W	SDMMC0BRESP[1:0] Clear Writing 1 to this bit clears the SDMMC0BRESP[1:0] bits to 00. This bit is always read as 0.
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26	SDMMC1 RRESP CLR	0	R/W	SDMMC1RRESP[1:0] Clear Writing 1 to this bit clears the SDMMC1RRESP[1:0] bits to 00. This bit is always read as 0.
25	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
24	SDMMC1 BRESP CLR	0	R/W	SDMMC1BRESP[1:0] Clear Writing 1 to this bit clears the SDMMC1BRESP[1:0] bits to 00. This bit is always read as 0.
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	NAND RRESP CLR	0	R/W	NANDRRESP[1:0] Clear Writing 1 to this bit clears the NANDRRESP[1:0] bits to 00. This bit is always read as 0.
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20	NAND BRESP CLR	0	R/W	NANDBRESP[1:0] Clear Writing 1 to this bit clears the NANDBRESP[1:0] bits to 00. This bit is always read as 0.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	DRP RRESP CLR	0	R/W	DRPRRESP[1:0] Clear Writing 1 to this bit clears the DRPRRESP[1:0] bits to 00. This bit is always read as 0. For products without a DRP, always write 0 to this bit.
17	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
16	DRP BRESP CLR	0	R/W	DRPBRESP[1:0] Clear Writing 1 to this bit clears the DRPBRESP[1:0] bits to 00. This bit is always read as 0. For products without a DRP, always write 0 to this bit.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.19 AXI Bus Response Error Clear Register 2 (AXIRERRCLR2)

This register clears the AXI bus response error status.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	VDC601 RRESP CLR	—	VDC601 BRESP CLR	—	VDC602 RRESP CLR	—	—	—	—	—	—	—	VDC604 RRESP CLR	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W	R	R/W	R	R	R	R	R	R	R	R/W	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	VDC601 RRESP CLR	0	R/W	VDC601RRESP[1:0] Clear Writing 1 to this bit clears the VDC601RRESP[1:0] bits to 00. This bit is always read as 0.
29	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
28	VDC601 BRESP CLR	0	R/W	VDC601BRESP[1:0] Clear Writing 1 to this bit clears the VDC601BRESP[1:0] bits to 00. This bit is always read as 0.
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26	VDC602 RRESP CLR	0	R/W	VDC602RRESP[1:0] Clear Writing 1 to this bit clears the VDC602RRESP[1:0] bits to 00. This bit is always read as 0.
25 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18	VDC604 RRESP CLR	0	R/W	VDC604RRESP[1:0] Clear Writing 1 to this bit clears the VDC604RRESP[1:0] bits to 00. This bit is always read as 0.
17 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.20 Master Access Control Register 0 (MSTACCCTL0)

This register controls the access attribute when the master module accesses the slave area.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	JCU ARNS	—	—	—	JCU AWNS	—	—	—	—	—	—	—	ETH AxNS	—
Initial value :	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0
R/W :	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R/W	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	IMR20 ARNS	—	—	—	IMR20 AWNS	—	—	—	—	—	—	—	VIN AWNS	—
Initial value :	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0
R/W :	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29	JCU ARNS	1	R/W	JPEG Codec Unit Access Attribute when Reading This bit specifies access attribute when the JPEG codec unit performs read access. The value of this bit is used as the ARPROT[1] signal for the JPEG codec unit. 0: Secure access 1: Non-secure access Modify the value of this bit only while the JPEG codec unit does not use the internal bus.
28 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	JCU AWNS	1	R/W	JPEG Codec Unit Access Attribute when Writing This bit specifies access attribute when the JPEG codec unit performs write access. The value of this bit is used as the AWPROT[1] signal for the JPEG codec unit. 0: Secure access 1: Non-secure access Modify the value of this bit only while the JPEG codec unit does not use the internal bus.
24 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	ETH AxNS	1	R/W	Ethernet MAC controller Access Attribute when Reading/Writing This bit specifies access attribute when the Ethernet MAC controller performs read/write access. The value of this bit is used as the ARPROT[1]/AWPROT[1] signal for the Ethernet MAC controller. 0: Secure access 1: Non-secure access Modify the value of this bit only while the Ethernet MAC controller does not use the internal bus.
16 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	IMR20 ARNS	1	R/W	Image Renderer (IMR-LS2) Access Attribute when Reading This bit specifies access attribute when the Image Renderer (IMR-LS2) performs read access. The value of this bit is used as the ARPROT[1] signal for the Image Renderer (IMR-LS2) . 0: Secure access 1: Non-secure access Modify the value of this bit only while the Image Renderer (IMR-LS2) does not use the internal bus.
12 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9	IMR20 AWNS	1	R/W	Image Renderer (IMR-LS2) Access Attribute when Writing This bit specifies access attribute when the Image Renderer (IMR-LS2) performs write access. The value of this bit is used as the ARPROT[1] signal for the Image Renderer (IMR-LS2) . 0: Secure access 1: Non-secure access Modify the value of this bit only while the Image Renderer (IMR-LS2) does not use the internal bus.
8 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	VIN AWNS	1	R/W	Video Input Module Access Attribute when Writing This bit specifies access attribute when the Video Input Module performs write access. The value of this bit is used as the ARPROT[1] signal for the Video Input Module. 0: Secure access 1: Non-secure access Modify the value of this bit only while the Video Input Module does not use the internal bus.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

5.11.21 Master Access Control Register 1 (MSTACCCTL1)

This register controls the access attribute when the master module accesses the slave area.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CEU AWNS	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SDMMC 0ARNS	—	—	—	SDMMC 0AWNS	—	—	—	SDMMC 1ARNS	—	—	—	SDMMC 1AWNS	—
Initial value :	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0
R/W :	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	CEU AWNS	1	R/W	Capture Engine Unit Access Attribute when Writing This bit specifies access attribute when the Capture Engine Unit performs write access. The value of this bit is used as the AWPROT[1] signal for the Capture Engine Unit. 0: Secure access 1: Non-secure access Modify the value of this bit only while the Capture Engine unit does not use the internal bus.
16 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	SDMMC0 ARNS	1	R/W	SD/MMC host interface channel 0 Access Attribute when Reading This bit specifies access attribute when the SD/MMC host interface channel 0 performs read access. The value of this bit is used as the ARPROT[1] signal for the SD/MMC host interface channel 0. 0: Secure access 1: Non-secure access Modify the value of this bit only while the SD/MMC host interface channel 0 does not use the internal bus.
12 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	SDMMC0 AWNS	1	R/W	SD/MMC host interface channel 0 Access Attribute when Writing This bit specifies access attribute when the SD/MMC host interface channel 1 performs write access. The value of this bit is used as the AWPROT[1] signal for the SD/MMC host interface channel 0. 0: Secure access 1: Non-secure access Modify the value of this bit only while the SD/MMC host interface channel 0 does not use the internal bus.
8 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	SDMMC1 ARNS	1	R/W	SD/MMC host interface channel 1 Access Attribute when Reading This bit specifies access attribute when the SD/MMC host interface channel 1 performs read access. The value of this bit is used as the ARPROT[1] signal for the SD/MMC host interface channel 1. 0: Secure access 1: Non-secure access Modify the value of this bit only while the SD/MMC host interface channel 1 does not use the internal bus.

Bit	Bit Name	Initial Value	R/W	Description
4 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	SDMMC1 AWNS	1	R/W	SD/MMC host interface channel 1 Access Attribute when Writing This bit specifies access attribute when the SD/MMC host interface channel 1 performs write access. The value of this bit is used as the AWPROT[1] signal for the SD/MMC host interface channel 1. 0: Secure access 1: Non-secure access Modify the value of this bit only while the SD/MMC host interface channel 1 does not use the internal bus.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

5.11.22 Master Access Control Register 2 (MSTACCCTL2)

This register controls the access attribute when the master module accesses the slave area.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	NAND ARNS	—	—	—	NAND AWNS	—	—	—	DRP ARNS	—	—	—	DRP AWNS	—
Initial value :	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0
R/W :	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	D2D0 AxNS	—	—	—	—	—	—	—	D2D1 AxNS	—
Initial value :	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
R/W :	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29	NAND ARNS	1	R/W	NAND flash controller Access Attribute when Reading This bit specifies access attribute when the NAND flash controller performs read access. The value of this bit is used as the ARPROT[1] signal for the NAND flash controller. 0: Secure access 1: Non-secure access Modify the value of this bit only while the NAND flash controller does not use the internal bus.
28 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	NAND AWNS	1	R/W	NAND flash controller Access Attribute when Writing This bit specifies access attribute when the NAND flash controller performs write access. The value of this bit is used as the AWPROT[1] signal for the NAND flash controller. 0: Secure access 1: Non-secure access Modify the value of this bit only while the NAND flash controller does not use the internal bus.
24 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	DRP ARNS	1	R/W	DRP Access Attribute when Reading This bit specifies access attribute when the DRP performs read access. The value of this bit is used as the ARPROT[1] signal for the DRP. 0: Secure access 1: Non-secure access Modify the value of this bit only while the DRP does not use the internal bus. The setting of this bit is only effective for products with a DRP. For products without a DRP, always write 1 to this bit.
20 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	DRP AWNS	1	R/W	DRP Access Attribute when Writing This bit specifies access attribute when the DRP performs write access. The value of this bit is used as the ARPROT[1] signal for the DRP. 0: Secure access 1: Non-secure access Modify the value of this bit only while the DRP does not use the internal bus. The setting of this bit is only effective for products with a DRP. For products without a DRP, always write 1 to this bit.
16 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9	D2D0 AxNS	1	R/W	2D Drawing Engine channel 0 Access Attribute when Reading/Writing This bit specifies access attribute when the 2D Drawing Engine channel 0 performs read/write access. The value of this bit is used as the ARPROT[1]/AWPROT[1] signal for the 2D Drawing Engine channel 0. 0: Secure access 1: Non-secure access Modify the value of this bit only while the 2D Drawing Engine channel 0 does not use the internal bus.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	D2D1 AxNS	1	R/W	2D Drawing Engine channel 1 Access Attribute when Reading/Writing This bit specifies access attribute when the 2D Drawing Engine channel 1 performs read/write access. The value of this bit is used as the ARPROT[1]/AWPROT[1] signal for the 2D Drawing Engine channel 1. 0: Secure access 1: Non-secure access Modify the value of this bit only while the 2D Drawing Engine channel 1 does not use the internal bus.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

5.11.23 Master Access Control Register 3 (MSTACCCTL3)

This register controls the access attribute when the master module accesses the slave area.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	VDC601 ARNS	—	—	—	VDC601 AWNS	—	—	—	VDC602 ARNS	—	—	—	—	—
Initial value :	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0
R/W :	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	VDC604 ARNS	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29	VDC601 ARNS	1	R/W	Video Display Controller 6 IV3-BUS and Sprite Engine (RLED) Access Attribute when Reading This bit specifies access attribute when the Video Display Controller 6 IV3-BUS and Sprite Engine (RLED) perform read access. The value of this bit is used as the ARPROT[1] signal for the Video Display Controller 6 IV3-BUS and Sprite Engine (RLED). 0: Secure access 1: Non-secure access Modify the value of this bit only while the Video Display Controller 6 IV3-BUS and Sprite Engine (RLED) do not use the internal bus.
28 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	VDC601 AWNS	1	R/W	Video Display Controller 6 IV1-BUS Access Attribute when Writing This bit specifies access attribute when the Video Display Controller 6 IV1-BUS performs write access. The value of this bit is used as the AWPROT[1] signal for the Video Display Controller 6 IV1-BUS. 0: Secure access 1: Non-secure access Modify the value of this bit only while the Video Display Controller 6 IV3-BUS does not use the internal bus.
24 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	VDC602 ARNS	1	R/W	Video Display Controller 6 IV5-BUS and Sprite Engine (SU0) Access Attribute when Reading This bit specifies access attribute when the Video Display Controller 6 IV5-BUS and Sprite Engine (SU0) perform read access. The value of this bit is used as the ARPROT[1] signal for the Video Display Controller 6 IV5-BUS and Sprite Engine (SU0). 0: Secure access 1: Non-secure access Modify the value of this bit only while the Video Display Controller 6 IV5-BUS and Sprite Engine (SU0) do not use the internal bus.
20 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5	VDC604 ARNS	1	R/W	Video Display Controller 6 IV6-BUS and Sprite Engine (SU1) Access Attribute when Reading This bit specifies access attribute when the Video Display Controller 6 IV6-BUS and Sprite Engine (SU1) perform read access. The value of this bit is used as the ARPROT[1] signal for the Video Display Controller 6 IV6-BUS and Sprite Engine (SU1). 0: Secure access 1: Non-secure access Modify the value of this bit only while the Video Display Controller 6 IV6-BUS and Sprite Engine (SU1) do not use the internal bus.
4 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.24 Master Access Control Register 4 (MSTACCCTL4)

This register controls the access attribute when the master module accesses the slave area.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	USB00 AxNS	—	—	—	—	—	—	—	USB01 AxNS	—
Initial value :	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
R/W :	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	USB10 AxNS	—	—	—	—	—	—	—	USB11 AxNS	—
Initial value :	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
R/W :	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	USB00 AxNS	1	R/W	USB 2.0 Host/Function Module channel 0 port 0 Access Attribute when Reading/Writing This bit specifies access attribute when the USB 2.0 Host/Function Module channel 0 port 0 performs read/write access. The value of this bit is used as the ARPROT[1]/AWPROT[1] signal for the USB 2.0 Host/Function Module channel 0 port 0. 0: Secure access 1: Non-secure access Modify the value of this bit only while the USB 2.0 Host/Function Module channel 0 port 0 does not use the internal bus.
24 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	USB01 AxNS	1	R/W	USB 2.0 Host/Function Module channel 0 port 1 Access Attribute when Reading/Writing This bit specifies access attribute when the USB 2.0 Host/Function Module channel 0 port 1 performs read/write access. The value of this bit is used as the ARPROT[1]/AWPROT[1] signal for the USB 2.0 Host/Function Module channel 0 port 1. 0: Secure access 1: Non-secure access Modify the value of this bit only while the USB 2.0 Host/Function Module channel 0 port 1 does not use the internal bus.
16 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	USB10 AxNS	1	R/W	USB 2.0 Host/Function Module channel 1 port 0 Access Attribute when Reading/Writing This bit specifies access attribute when the USB 2.0 Host/Function Module channel 1 port 0 performs read/write access. The value of this bit is used as the ARPROT[1]/AWPROT[1] signal for the USB 2.0 Host/Function Module channel 1 port 0. 0: Secure access 1: Non-secure access Modify the value of this bit only while the USB 2.0 Host/Function Module channel 1 port 0 does not use the internal bus.
8 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	USB11 AxNS	1	R/W	USB 2.0 Host/Function Module channel 1 port 1 Access Attribute when Reading/Writing This bit specifies access attribute when the USB 2.0 Host/Function Module channel 1 port 1 performs read/write access. The value of this bit is used as the ARPROT[1]/AWPROT[1] signal for the USB 2.0 Host/Function Module channel 1 port 1. 0: Secure access 1: Non-secure access Modify the value of this bit only while the USB 2.0 Host/Function Module channel 1 port 1 does not use the internal bus.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

5.11.25 Slave Access Control Register 0 (SLVACCCTL0)

This register specifies whether secure access to the modules in the slave area is only allowed or not.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	SYS NS	—	—	—	—	—	VDC60 NS	—	IMR20 NS	—	MTU3 NS	—	GPT NS	—	POE3 NS
Initial value :	0	1	0	0	0	1	0	1	0	1	0	1	0	1	0	1
R/W :	R	R/W	R	R	R	R	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	POEG NS	—	—	—	—	—	—	—	INTC2 NS	—	WDT NS	—	—	—	—
Initial value :	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0	0
R/W :	R	R/W	R	R	R	R	R	R	R	R/W	R	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	SYS NS	1	R/W	System Control Registers Access Attribute This bit specifies whether secure access to the registers for low-power mode and clock-pulse generator is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the registers for low-power mode and clock-pulse generator are not being accessed.
29 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
25	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
24	VDC60 NS	1	R/W	Video Display Controller 6 Access Attribute This bit specifies whether secure access to the video display controller 6 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the video display controller 6 is not being accessed.
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	IMR20 NS	1	R/W	Distortion correction engine (IMR-LS2) Access Attribute This bit specifies whether secure access to the Distortion correction engine (IMR-LS2) is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Distortion correction engine (IMR-LS2) is not being accessed.
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20	MTU3 NS	1	R/W	Multi-function timer pulse unit 3 Access Attribute This bit specifies whether secure access to the Multi-function timer pulse unit 3 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Multi-function timer pulse unit 3 is not being accessed.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
18	GPT NS	1	R/W	General-Purpose PWM Timer Access Attribute This bit specifies whether secure access to the General-Purpose PWM Timer is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the General-Purpose PWM Timer is not being accessed.
17	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
16	POE3 NS	1	R/W	Port Output Enable 3 Access Attribute This bit specifies whether secure access to the Port Output Enable 3 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Port Output Enable 3 is not being accessed.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	POEG NS	1	R/W	Port Output Enable Access Attribute for GPT This bit specifies whether secure access to the port output enable for GPT is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the port output enable for GPT is not being accessed.
13 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	INTC2 NS	1	R/W	Interrupt Controller (other than GIC-400 register) Access Attribute This bit specifies whether secure access to the Interrupt Controller (other than GIC-400 register) is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Interrupt Controller (other than GIC-400 register) is not being accessed.
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4	WDT NS	1	R/W	Watchdog Timer Access Attribute This bit specifies whether secure access to the Watchdog Timer is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Watchdog Timer is not being accessed.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.26 Slave Access Control Register 1 (SLVACCCTL1)

This register specifies whether secure access to the modules in the slave area is only allowed or not.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	GPIO NS	—	I2C NS	—	OSTM0 NS	—	OSTM1 NS	—	OSTM2 NS	—	SSIF NS	—	SPDIF NS	—	RCAN NS
Initial value :	0	1	0	1	0	0	0	1	0	1	0	1	0	1	0	1
R/W :	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SE NS	—	—	—	AD NS	—	IRDA NS	—	SCI NS	—	SCIF NS	—	JCU NS	—	RSPI NS
Initial value :	0	1	0	0	0	1	0	1	0	1	0	1	0	1	0	1
R/W :	R	R/W	R	R	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
30	GPIO NS	1	R/W	General I/O ports Access Attribute This bit specifies whether secure access to the General I/O ports is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the General I/O ports is not being accessed.
29	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
28	I^2C NS	1	R/W	I^2C Bus Interface channel 0 to 3 Access Attribute This bit specifies whether secure access to the I^2C Bus Interface channel 0 to 3 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the I^2C Bus Interface channel 0 to 3 is not being accessed.
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26	OSTM0 NS	0	R/W	OS Timer channel 0 Access Attribute This bit specifies whether secure access to the OS Timer channel 0 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the OS Timer channel 0 is not being accessed.
25	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
24	OSTM1 NS	1	R/W	OS Timer channel 1 Access Attribute This bit specifies whether secure access to the OS Timer channel 1 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the OS Timer channel 1 is not being accessed.
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	OSTM2 NS	1	R/W	OS Timer channel 2 Access Attribute This bit specifies whether secure access to the OS Timer channel 2 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the OS Timer channel 2 is not being accessed.
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
20	SSIFNS	1	R/W	Serial Sound Interface Access Attribute This bit specifies whether secure access to the Serial Sound Interface is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Serial Sound Interface is not being accessed.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	SPDIFNS	1	R/W	Renesas SPDIF Interface Access Attribute This bit specifies whether secure access to the Renesas SPDIF Interface is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Renesas SPDIF Interface is not being accessed.
17	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
16	RCANNS	1	R/W	CANFD Interface Access Attribute This bit specifies whether secure access to the CANFD interface is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the CANFD interface is not being accessed.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	SENS	1	R/W	Sprite Engine Access Attribute This bit specifies whether secure access to the Sprite Engine is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Sprite Engine is not being accessed.
13 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	ADNS	1	R/W	A/D Converter Access Attribute This bit specifies whether secure access to the A/D Converter is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the A/D Converter is not being accessed.
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	IRDANS	1	R/W	IrDA Access Attribute This bit specifies whether secure access to the IrDA is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the IrDA is not being accessed.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	SCINS	1	R/W	Serial Communications Interface channel 0, 1 Access Attribute This bit specifies whether secure access to the Serial Communications Interface channel 0, 1 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Serial Communications Interface channel 0, 1 is not being accessed.
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4	SCIFNS	1	R/W	Serial Communications Interface with FIFO channel 0 to 4 Access Attribute This bit specifies whether secure access to the Serial Communications Interface with FIFO channel 0 to 4 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Serial Communications Interface with FIFO channel 0 to 4 is not being accessed.

Bit	Bit Name	Initial Value	R/W	Description
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	JCU NS	1	R/W	JPEG Codec Unit Access Attribute This bit specifies whether secure access to the JPEG Codec Unit is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the JPEG Codec Unit is not being accessed.
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	RSPI NS	1	R/W	Renesas Serial Peripheral Interface channel 0 to 2 Access Attribute This bit specifies whether secure access to the Renesas Serial Peripheral Interface channel 0 to 2 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Renesas Serial Peripheral Interface channel 0 to 2 is not being accessed.

5.11.27 Slave Access Control Register 2 (SLVACCCTL2)

This register specifies whether secure access to the modules in the slave area is only allowed or not.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	ETH NS	—	—	—	D2D NS	—	MIPI NS	—	VIN NS	—	—	—	—	—	—
Initial value :	0	1	0	0	0	1	0	1	0	1	0	1	0	1	0	1
R/W :	R	R/W	R	R	R	R/W	R	R/W	R	R/W	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	USB00 NS	—	USB01 NS	—	USB10 NS	—	USB11 NS	—	CEU NS	—	DRP NS	—	TSIP NS	—	—
Initial value :	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0
R/W :	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	ETH NS	1	R/W	Ethernet MAC controller channel 0/1 Access Attribute This bit specifies whether secure access to the Ethernet MAC controller channel 0/1 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Ethernet MAC controller channel 0/1 is not being accessed.
29 to 27	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	D2D NS	1	R/W	2D Drawing Engine Access Attribute This bit specifies whether secure access to the 2D Drawing Engine is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the 2D Drawing Engine is not being accessed.
25	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
24	MIPI NS	1	R/W	MIPI CSI-2 Interface Access Attribute This bit specifies whether secure access to the MIPI CSI-2 Interface is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the MIPI CSI-2 Interface is not being accessed.
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	VIN NS	1	R/W	Video Input Module Access Attribute This bit specifies whether secure access to the Video Input Module is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Video Input Module is not being accessed.
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

Bit	Bit Name	Initial Value	R/W	Description
17	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
16	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	USB00NS	1	R/W	USB 2.0 Host/Function Module channel 0 port 0 Access Attribute This bit specifies whether secure access to the USB 2.0 Host/Function Module channel 0 port 0 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the USB 2.0 Host/Function Module channel 0 port 0 is not being accessed.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	USB01NS	1	R/W	USB 2.0 Host/Function Module channel 0 port 1 Access Attribute This bit specifies whether secure access to the USB 2.0 Host/Function Module channel 0 port 1 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the USB 2.0 Host/Function Module channel 0 port 1 is not being accessed.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	USB10NS	1	R/W	USB 2.0 Host/Function Module channel 1 port 0 Access Attribute This bit specifies whether secure access to the USB 2.0 Host/Function Module channel 1 port 0 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the USB 2.0 Host/Function Module channel 1 port 0 is not being accessed.
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	USB11NS	1	R/W	USB 2.0 Host/Function Module channel 1 port 1 Access Attribute This bit specifies whether secure access to the USB 2.0 Host/Function Module channel 1 port 1 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the USB 2.0 Host/Function Module channel 1 port 1 is not being accessed.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	CEUNS	1	R/W	Capture Engine Unit Access Attribute This bit specifies whether secure access to the Capture Engine Unit is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Capture Engine Unit is not being accessed.
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4	DRPNS	1	R/W	DRP Access Attribute This bit specifies whether secure access to the DRP is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the DRP is not being accessed. The setting of this bit is only effective for products with a DRP. For products without a DRP, always write 1 to this bit.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	TSIPNS	0	R/W	Trusted Secure IP Access Attribute This bit specifies whether secure access to the Trusted Secure IP is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Trusted Secure IP is not being accessed. Note: Always write 0 to this bit in products without the Trusted Secure IP.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.28 Slave Access Control Register 3 (SLVACCCTL3)

This register specifies whether secure access to the modules in the slave area is only allowed or not.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SDMMC 0NS	—	SDMMC 1NS	—	NAND NS	—	—	—	CS NS	—	—	—	—
Initial value :	0	0	0	1	0	1	0	1	0	0	0	1	0	0	0	0
R/W :	R	R	R	R/W	R	R/W	R	R/W	R	R	R	R/W	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	SDMMC0 NS	1	R/W	SD/MMC host interface channel 0 Access Attribute This bit specifies whether secure access to the SD/MMC host interface channel 0 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the SD/MMC host interface channel 0 is not being accessed.
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26	SDMMC1 NS	1	R/W	SD/MMC host interface channel 1 Access Attribute This bit specifies whether secure access to the SD/MMC host interface channel 1 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the SD/MMC host interface channel 1 is not being accessed.
25	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
24	NAND NS	1	R/W	NAND flash controller Access Attribute This bit specifies whether secure access to the NAND flash controller is only allowed or not. 0: Secure access is only allowed./ 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the NAND flash controller is not being accessed.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	CS NS	1	R/W	CoreSight Access Attribute This bit specifies whether secure access to the CoreSight is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the CoreSight is not being accessed.
19 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.11.29 Slave Access Control Register 4 (SLVACCCTL4)

This register specifies whether secure access to the modules in the slave area is only allowed or not.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	BSC NS	—	SPI NS	—	OCTA NS	—	OCTAR NS	—	HYP NS	—	HYPR NS	—	RRAM NS	—	VRAM0 NS
Initial value :	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
R/W :	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VRAM1 NS	—	VRAM2 NS	—	VRAM3 NS	—	VRAM4 NS	—	—	—	—	—	—	—	—
Initial value :	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W	R	R/W	R	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	BSC NS	1	R/W	Bus State Controller Access Attribute This bit specifies whether secure access to the Bus State Controller is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Bus State Controller is not being accessed.
29	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
28	SPI NS	1	R/W	SPI Multi I/O bus controller Access Attribute This bit specifies whether secure access to the SPI Multi I/O bus controller is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the SPI Multi I/O bus controller is not being accessed.
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26	OCTA NS	1	R/W	Octa Memory Controller Access Attribute This bit specifies whether secure access to the Octa Memory Controller is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Octa Memory Controller is not being accessed.
25	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
24	OCTAR NS	1	R/W	Octa Memory Controller Register Area Access Attribute This bit specifies whether secure access to the Octa Memory Controller Register Area is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the Octa Memory Controller Register Area is not being accessed.
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	HYP NS	1	R/W	HyperBus™ Controller Access Attribute This bit specifies whether secure access to the HyperBus™ Controller is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the HyperBus™ Controller is not being accessed.

Bit	Bit Name	Initial Value	R/W	Description
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20	HYPRNS	1	R/W	HyperBus™ Controller Register Area Access Attribute This bit specifies whether secure access to the HyperBus™ Controller Register Area is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the HyperBus™ Controller Register Area is not being accessed.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	RRAMNS	1	R/W	On-Chip Data-Retention RAM Access Attribute This bit specifies whether secure access to the On-Chip Data-Retention RAM is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the On-Chip Data-Retention RAM is not being accessed.
17	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
16	VRAM0NS	1	R/W	On-chip large-capacity RAM page 0 Access Attribute This bit specifies whether secure access to the On-chip large-capacity RAM page 0 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the On-chip large-capacity RAM page 0 is not being accessed.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	VRAM1NS	1	R/W	On-chip large-capacity RAM page 1 Access Attribute This bit specifies whether secure access to the On-chip large-capacity RAM page 1 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the On-chip large-capacity RAM page 1 is not being accessed.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	VRAM2NS	1	R/W	On-chip large-capacity RAM page 2 Access Attribute This bit specifies whether secure access to the On-chip large-capacity RAM page 2 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the On-chip large-capacity RAM page 2 is not being accessed.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	VRAM3NS	1	R/W	On-chip large-capacity RAM page 3 Access Attribute This bit specifies whether secure access to the On-chip large-capacity RAM page 3 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the On-chip large-capacity RAM page 3 is not being accessed.
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
8	VRAM4NS	1	R/W	On-chip large-capacity RAM page 4 Access Attribute This bit specifies whether secure access to the On-chip large-capacity RAM page 4 is only allowed or not. 0: Secure access is only allowed. 1: Non-secure access or secure access is allowed. Modify the value of this bit only while the On-chip large-capacity RAM page 4 is not being accessed.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.12 Interrupt Request

When a decode error or a slave error occurs as described in the AXI bus response error status register (AXIRERRST), an AXI bus response error interrupt request (PRRI) is issued.
An interrupt request is issued when a response is returned from the bus for which interrupt requests are enabled through the AXI bus response error interrupt control register (AXIRERRCTL). To check the response error type, read the AXI bus response error status register (AXIRERRST). To clear the interrupt request, clear the AXI bus response error status register through the AXI bus response error clear register (AXIRERRCLR).
The write buffer listed in Table 5.7 outputs an interrupt request, when an error response is received during the transfer described in the condition.
The interrupt signal is asserted high for eight cycles of the bus clock (Bφ). See Table 5.7 for the interrupt signals for the write buffers.
This interrupt should be used only for debugging purposes. Make sure that no response error occurs during system operation.

Table 5.7 Interrupt Signals for the Write Buffers

Interrupt signals	Write Buffer Connection Source/Destination	Condition
X2HPERI12_ERRINT	North main bus / Peripheral buses 1 and 2	During write transfer with bufferable attribute
X2HPERI34_ERRINT	North main bus / Peripheral bus 3	
X2HPERI5_ERRINT	North main bus / Peripheral bus 4	
X2HPERI67_ERRINT	North main bus / Peripheral bus 5	
X2HDBGR_ERRINT	North main bus / Peripheral bus 6	
H2XDBG_ERRINT	CoreSight / North main bus	During write or read transfer
H2XETH_ERRINT	Ethernet MAC controller / South main bus 2	
H2XDAV0_ERRINT	2D Drawing Engine 0 / South main bus 1	
H2XDAV1_ERRINT	2D Drawing Engine 1 / South main bus 1	
H2XUSB00_ERRINT	USB 2.0 channel 0 / South main bus 2	
H2XUSB01_ERRINT	USB 2.0 channel 0 / South main bus 2	
H2XUSB10_ERRINT	USB 2.0 channel 1 / South main bus 2	
H2XUSB11_ERRINT	USB 2.0 channel 1 / South main bus 2	

6. Clock Pulse Generator

This LSI has a clock pulse generator that generates a CPU clock (Iφ), image processing clock (Gφ), internal bus clock (Bφ), peripheral clock 1 (P1φ), and peripheral clock 0 (P0φ). The clock pulse generator consists of a crystal oscillator, PLL circuits, and divider circuits.

6.1 Features

- Clock types
  A CPU clock (Iφ); an image processing clock (Gφ); an internal bus clock (Bφ); peripheral clock 1 (P1φ) for the external bus interface; peripheral clock 0 (P0φ) for the on-chip peripheral modules. (CKIO: P1φ or Bφ)
- Frequency change function
  CPU and image processing clock frequencies can be changed independently using the PLL (phase locked loop) circuits and divider circuits within this module. Frequencies are changed by software using frequency control register (FRQCR) settings.
- Clock selection
  The clocks of CKIO, SPI multi I/O bus controller, HyperBus™ controller, and Octa memory controller can be changed by setting register (CKIOSEL, SCLKSEL).
- Power-down mode control
  The clock can be stopped in sleep mode, software standby mode, and deep standby mode, and specific modules can be stopped using the module standby function. For details on clock control in the power-down modes, see section 52, Power-Down Modes.
- SSCG function
  The PLL (phase locked loop) circuit in the clock pulse generator includes an SSCG (spread spectrum clock generator).
  The SSCG can be used to decrease the peak value of EMI (electromagnetic interference) noise by frequency modulation, that is, by slightly modulating the output frequency.
  The specification of the SSCG for this LSI is as follows.
- Specification of SSCG
  (1) Modulation waveform (modulation profile): Triangle wave
  (2) Type of spreading: Down-spreading
  (3) Modulation rate: -2.2%
  (4) Modulation frequency: 20.00 to 24.00 kHz
  (clock mode 0: frequency on the EXTAL pin ÷ 500
  clock mode 1: frequency on the EXTAL pin ÷ 1000)

Figure 6.1 shows a block diagram of the clock pulse generator.

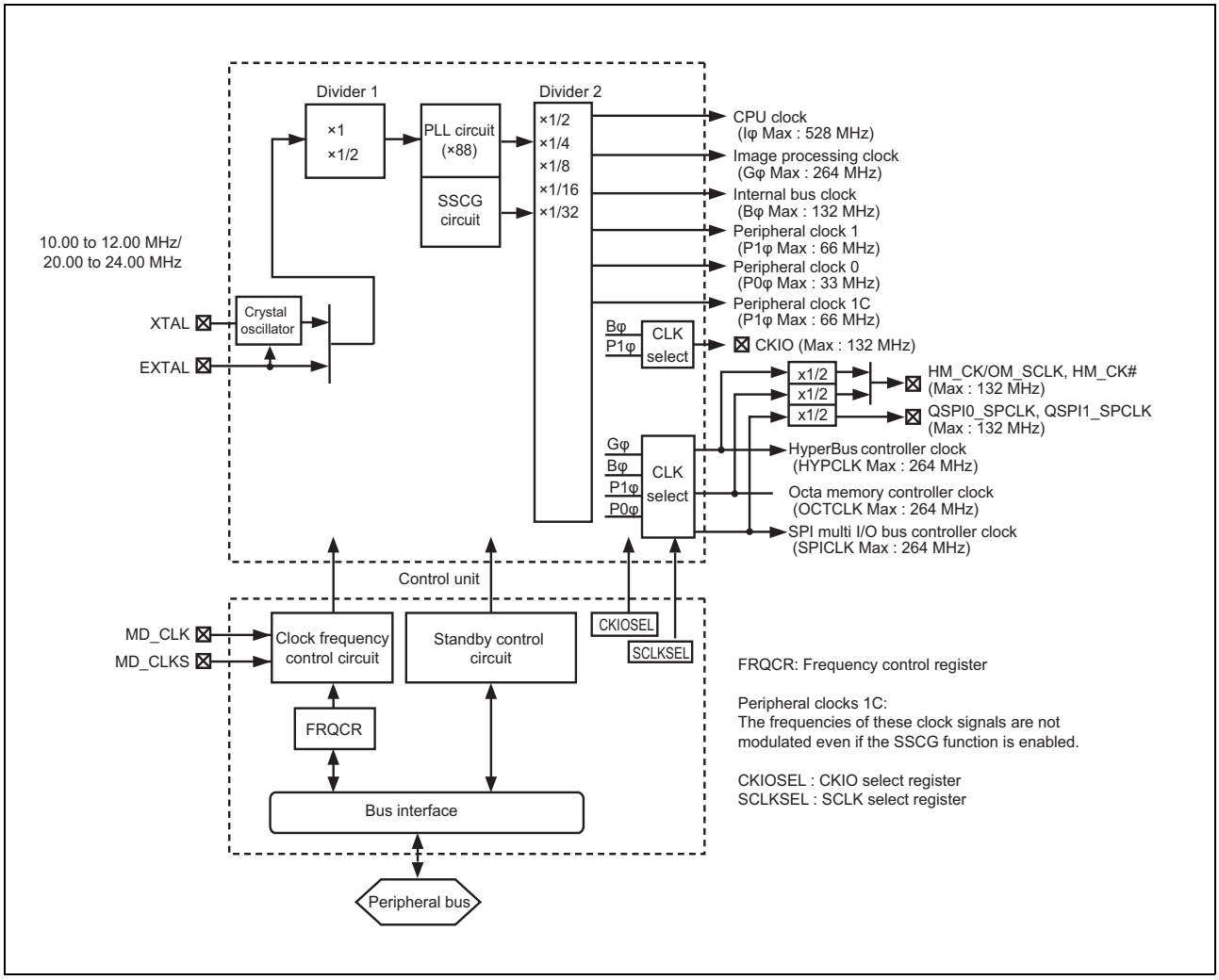


Figure 6.1 Block Diagram

The blocks of this module function as follows:

(1) Crystal Oscillator

A crystal oscillator is connected to the XTAL and EXTAL pins. The frequency range is selected by the clock mode settings.

(2) PLL Circuit

The PLL circuit is capable of multiplying the frequency of the input clock signal from the EXTAL pin by 88.

(3) Divider 1 and Divider 2

The frequency division ratio by divider 1 is one or one half, and is set by the clock mode.
Divider 2 generates a clock signal whose operating frequency can be used for the CPU clock, image processing clock, internal bus clock, peripheral clock 1, and peripheral clock 0. Except for peripheral clock 0, the division ratio is set by the frequency control register (FRQCR). The division ratios for peripheral clock 0 is fixed to 1/32.

(4) Clock Frequency Control Circuit

The clock frequency control circuit controls the clock frequency using the frequency control register (FRQCR).

(5) Standby Control Circuit

The standby control circuit controls the states of the on-chip oscillation circuit and other modules during clock switching, or, software standby or deep standby mode.
In addition, the standby control register is provided to control the power-down mode of other modules. For details on the standby control register, see section 52, Power-Down Modes.

(6) Frequency Control Register (FRQCR)

The frequency control register (FRQCR) has control bits assigned for the following functions: clock output/non-output from the CKIO pin during software standby mode or deep standby mode and the frequency division ratio of the CPU clock (Iφ), the internal bus clock (Bφ), and the peripheral clock 1 (P1φ). The frequency control register (FRQCR) has control bits assigned for the frequency division ratio of the image processing clock (Gφ).

(7) CKIO Select Register (CKIOSEL) / SCLK Select Register (SCLKSEL)

The clock select registers (CKIOSEL, SCLKSEL) have control bits assigned for clock selection of the CKIO, the SPI multi I/O bus controller, the HyperBus controller, and the Octa memory controller.

(8) SSCG Circuit

Operation of the SSCG circuit is switched on or off (enabled or disabled) by the MD_CLKS pin. When the SSCG function is disabled, all of the internal clock frequencies are fixed, i.e. not modulated. When the SSCG function is enabled, the frequencies of clock signals supplied to peripheral modules other than those listed below are modulated.

Peripheral modules to which non-modulated clock signals are supplied:
Multi-function timer pulse unit 3, serial communications interface with FIFO, CAN FD interface, OS timer, general PWM timer, serial communication interface, LVDS output interface (LVDS PLL circuit only), port output enable, and dynamic reconfigurable processor.

6.2 Input/Output Pins

Table 6.1 lists the clock pulse generator pins and their functions.

Table 6.1 Pin Configuration and Functions of the Clock Pulse Generator

Pin Name	Symbol	I/O	Function
Mode control pin	MD_CLK	Input	Switches the frequency range of EXTAL input.
	MD_CLKS	Input	Enables or disables the SSCG circuit.
Crystal input/output pins (clock input pins)	XTAL	Output	Connected to the crystal resonator. (Leave this pin open when the crystal resonator is not in use.)
	EXTAL	Input	Connected to the crystal resonator or used to input external clock.
Clock output pin	CKIO	Output	Clock output pin.

6.3 Clock Mode

Table 6.2 indicates the input/output clock frequency.

Table 6.2 Input/Output Clock Frequency

Mode	MD_CLK Pin Setting	Clock I/O		Divider 1	PLL Circuit	CKIO Frequency
		Source	Output			
0	0	EXTAL/crystal resonator (10MHz to 12MHz)	CKIO	1	ON (x 88)	(EXTAL/crystal) Bφ :x11/x5.5/x2.75 P1φ:x5.5/x2.75
1	1	EXTAL/crystal resonator (20MHz to 24MHz)	CKIO	1/2	ON (x 88)	(EXTAL/crystal) Bφ :x5.5/x2.75/x1.375 P1φ:x2.75/x1.375

This LSI switches the division ratio of the PLL input clock by the setting of the MD_CLKS pin while the RES# pin is being held low.
The SSCG function is switched on or off by the setting of the MD_CLKS pin while the RES# pin is being held low. The following table shows the correspondence between SSCG operation and pin settings. Note that the pin setting does not affect the PLL frequency multipliers and division ratios for individual clock signals.

Table 6.3 SSCG operation settings

MD_CLKS pin setting value	SSCG operation
0	OFF
1	ON

Table 6.4 lists the available frequency range and Table 6.5 lists the usable frequency range of output clock (CKIO).

Table 6.4 Available frequency range

Mode	FRQCR Register Setting Value*1	PLL frequency multiplier: PLL circuit	Internal Clock ratio (I : G : B : P1 : P0)*2	Settable frequency range (MHz) (*3): Input clock	CPU clock (Iφ)	Image processing clock (Gφ)	Internal bus clock (Bφ)	Peripheral clock 1 (P1φ)	Peripheral clock 0 (P0φ)
0	H'x012	ON (×88)	44 : 22 : 11 : 5.5 : 2.75	10.0 to 12.0	440 to 528	220 to 264	110 to 132	55 to 66	27.5 to 33
	H'x112		22 : 22 : 11 : 5.5 : 2.75		220 to 264				
	H'x212		11 : 22 : 11 : 5.5 : 2.75		110 to 132				
	H'x322		5.5 : 11 : 5.5 : 5.5 : 2.75		55 to 66	110 to 132	55 to 66		
	H'x333		5.5 : 5.5 : 2.75 : 2.75 : 2.75			55 to 66	27.5 to 33	27.5 to 33	
1	H'x012	ON (×88)	22 : 11 : 5.5 : 2.75 : 1.375	20.0 to 24.0	440 to 528	220 to 264	110 to 132	55 to 66	27.5 to 33
	H'x112		11 : 11 : 5.5 : 2.75 : 1.375		220 to 264				
	H'x212		5.5 : 11 : 5.5 : 2.75 : 1.375		110 to 132				
	H'x322		2.75 : 5.5 : 2.75 : 2.75 : 1.375		55 to 66	110 to 132	55 to 66		
	H'x333		2.75 : 2.75 : 1.375 : 1.375 : 1.375			55 to 66	27.5 to 33	27.5 to 33	

Table 6.5 Usable frequency range of output clock (CKIO)

CKIO select register (CKIOSEL[1:0]) setting value	External clock (CKIO pin)
00	Bφ (max.132MHz)
01	P1φ (max. 66MHz)

Note 1. x in the FRQCR register setting depends on the set value in bits 12, 13, and 14.
Note 2. The ratio of clock frequencies, where the input clock frequency is assumed to be 1.
Note 3. When using the MIPI CSI-2 interface, be sure to set Gφ = 264 MHz. Gφ is always twice Bφ.
Note 4. When using the Image Renderer, be sure to set FRQCR = H'x012 or H'x112.

Caution: Do not use this LSI with frequency settings other than those in Table 6.4.

6.4 Register Descriptions

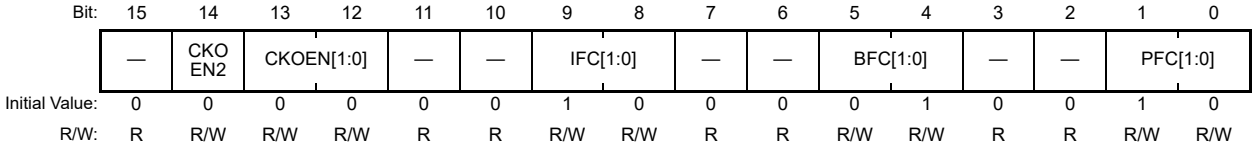
Table 6.6 shows the register configuration of the clock pulse generator.

Table 6.6 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Frequency Control Register	FRQCR	R/W	H'0212	H'FCFE0010	16
CKIO Select Register	CKIOSEL	R/W	H'0001	H'FCFE0100	16
SCLK Select Register	SCLKSEL	R/W	H'0000	H'FCFE0104	16

6.4.1 Frequency Control Register (FRQCR)

FRQCR is a 16-bit readable/writable register used to specify whether a clock is output from the CKIO pin during normal operation mode, change of gain of crystal oscillator for the XTAL pin, software standby mode, deep standby mode, and standby mode cancellation. The register specifies the frequency division ratio for the CPU clock (Iφ), the internal bus clock (Bφ), and the peripheral clock 1 (P1φ) used in the external bus interface. The image processing clock (Gφ) is always twice the internal bus clock (Bφ). FRQCR can be accessed in 16-bit units.



Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	CKOEN2	CKOEN[1:0]		—	—	IFC[1:0]		—	—	BFC[1:0]		—	—	PFC[1:0]	
Initial Value:	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0
R/W:	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	CKOEN2	0	R/W	Clock Output Enable 2 Specifies whether the CKIO pin outputs clock signals or is fixed to the low level when the gain of the crystal oscillator for the XTAL pin is changed. If this bit is set to 1, the CKIO pin is fixed to the low level when the gain of the crystal oscillator for the XTAL pin is changed. Therefore, the malfunction of an external circuit caused by an unstable CKIO clock while changing the gain of the crystal oscillator for the XTAL pin can be prevented. 0: Unstable clock output 1: Low-level output
13, 12	CKOEN [1:0]	00	R/W	Clock Output Enable These bits specify whether the CKIO pin outputs clock signals, or is set to a fixed level or high impedance (Hi-Z) during normal operation mode, deep standby mode, software standby mode, or cancellation of standby mode. If these bits are set to 01, the CKIO pin is fixed at low during deep standby mode, software standby mode, or cancellation of software standby mode. Therefore, the malfunction of an external circuit caused by an unstable CKIO clock during cancellation of software standby mode can be prevented. Table 6.7 lists CKOEN[1:0] settings.
11, 10	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9, 8	IFC [1:0]	10	R/W	CPU Clock Frequency Division Ratio These bits specify the frequency division ratio of the CPU clock with respect to the output frequency of PLL circuit. Note: See section 6.5.1, Changing the Division Ratio. 00: 1/2 time 01: 1/4 time 10: 1/8 time 11: 1/16 time
7, 6	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	BFC [1:0]	01	R/W	Internal bus clock Frequency Division Ratio These bits specify the frequency division ratio of the internal bus clock with respect to the output frequency of PLL circuit. Note: See section 6.5.1, Changing the Division Ratio. 00: Reserved (setting prohibited) 01: 1/8 time 10: 1/16 time 11: 1/32 time
3, 2	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	PFC [1:0]	10	R/W	Frequency division ratio of peripheral clock 1 used in the external bus interface These bits specify the frequency division ratio of the peripheral clock 1 with respect to the output frequency of PLL circuit. Note: See section 6.5.1, Changing the Division Ratio. 00: Reserved (setting prohibited) 01: Reserved (setting prohibited) 10: 1/16 time 11: 1/32 time

Table 6.7 CKOEN[1:0] Settings

Setting	Normal Operation	Software Standby Mode	Deep Standby Mode*
00	Output	Output off (Hi-Z)	Output off (Hi-Z)
01	Output	Low-level output	Low-level output
10	Output	Output (unstable clock output)	Low-level or high-level output
11	Output off (Hi-Z)	Output off (Hi-Z)	Output off (Hi-Z)

Note: Note that the first cycle of the output CKIO clock may be missing after release from deep standby.

6.4.2 CKIO Select Register (CKIOSEL)

CKIOSEL is a 16-bit readable/writable register used to select the CKIO output clock. CKIOSEL can be accessed in 16-bit units.

This register setting can not be processed while accessing the CS0 to CS5 external address spaces. This register setting must be processed by the program on the on-chip RAM.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CKIOSEL[1:0]	
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	CKIOSEL[1:0]	01	R/W	CKIO Output Clock Select 00: Bφ Clock output 01: P1φ Clock output 10: Reserved (Setting prohibited) 11: Reserved (Setting prohibited)

6.4.3 SCLK Select Register (SCLKSEL)

SCLKSEL is a 16-bit readable/writable register used to change the frequency of SPI multi I/O bus controller, HyperBus controller, and Octa memory controller. SCLKSEL can be accessed in 16-bit units.
Register setting of SCLKSEL can not be processed while accessing the setting target memory. This register setting must be processed by the program on the on-chip RAM.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OCTCR [1:0]		—	—	HYMCR [1:0]		—	—	SPICR[1:0]	
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	OCTCR [1:0]	00	R/W	Octa memory controller Clock Select*1 00: P0φ Clock output 01: P1φ Clock output 10: Bφ Clock output 11: Gφ Clock output
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	HYMCR [1:0]	00	R/W	HyperBus controller Clock Select*1 00: P0φ Clock output*2 01: P1φ Clock output 10: Bφ Clock output 11: Gφ Clock output
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	SPICR [1:0]	00	R/W	SPI multi I/O bus controller Clock Select*1 00: P0φ Clock output 01: P1φ Clock output 10: Bφ Clock output 11: Gφ Clock output

Note 1. The setting value that satisfies both the electrical characteristics of this LSI and the external memory must be selected.
Note 2. If the HyperBus controller is to be used, set these bits to a value other than 00.
Note 3. The frequency of the external output clock is one half that of the frequency selected by this register.

6.5 Changing the Frequency

The frequency of the CPU clock (Iφ), internal bus clock (Bφ), peripheral clock 1 (P1φ) and image processing clock (Gφ) can be changed by changing the division rate of divider. The division rate can be changed by software through the frequency control register (FRQCR).

6.5.1 Changing the Division Ratio

The division rate of divider can be changed by the following operation.

Change IFC[1:0], BFC[1:0] or PFC[1:0] of the frequency control register (FRQCR).
After waiting for the completion of the issued request from the bus master, hardware will automatically stop the bus master and start the frequency change.
If the issued request is not completed, the frequency can not be changed. Do not access registers of the modules in the standby state.

In addition, in order to suppress an unintended requests from the bus master, each bus master must be stopped by software before changing the frequency.
The clock selection registers (CKIOSEL, SCLKSEL) switch the clock of the CKIO, SPI multi I/O bus controller, HyperBus controller, and Octa memory controller without waiting for completion of the issued request for the bus master.

1. Set the standby_mode_en bit of the power control register in the PL310. For details of registers, refer to CoreLink Level 2 Cache Controller L2C-310 Technical Reference Manual issued by ARM Ltd.
2. In the initial state, IFC[1:0] = B'10, BFC[1:0] = B'01, and PFC[1:0] = B'10.
3. Set the desired value in the IFC[1:0], BFC[1:0], and PFC[1:0] bits. Note that if the wrong value is set, this LSI will malfunction.
4. After the register bits (IFC[1:0], BFC[1:0], and PFC[1:0]) have been set, the clock is supplied of the new division ratio.

Note: When executing the WFI instruction after changing the frequency, be sure to read the frequency control registers (FRQCR) to confirm that the new settings are in place and read the ISBUSY bit in the CPU status register (CPUSTS) to confirm that it is set to 0 beforehand.
For the CPUSTS register, see section 52, Power-Down Modes.

6.6 Usage of the Clock Pins

For the connection of a crystal resonator or the input of a clock signal, this LSI circuit has the pins listed in Table 6.8. With regard to these pins, take care on the following points. Furthermore, Xin pin and Xout pin are used in this section to refer to the pins listed in the table.

Table 6.8 Clock Pins

Xin Pins (Used for Connection of a Crystal Resonator and Input of External Clock Signals)	Xout Pins (Used for Connection of a Crystal Resonator)
EXTAL	XTAL
USB_X1	USB_X2
AUDIO_X1	AUDIO_X2
RTC_X1	RTC_X2

6.6.1 In the Case of Inputting an External Clock

An example of the connection of an external clock is shown in Figure 6.2. In cases where the Xout pin is left in the open-circuit state, make sure the parasitic capacitance is no greater than 10pF.

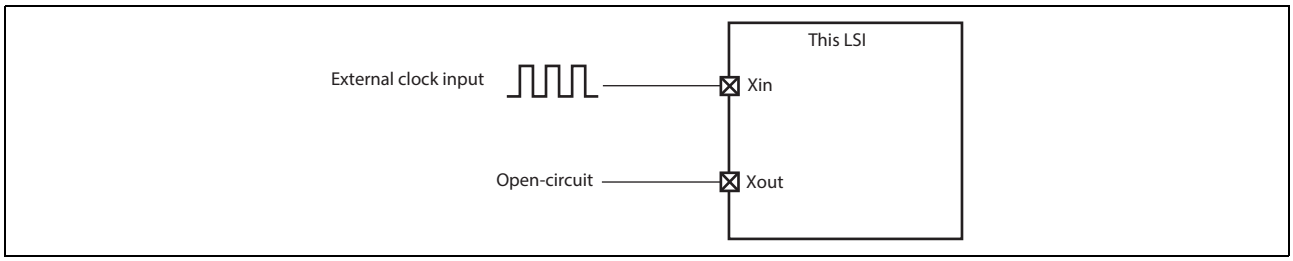


Figure 6.2 Example of the Connection of an External Clock

6.6.2 In the Case of Using a Crystal Resonator

An example of the connection of crystal resonator is shown in Figure 6.3.

Place the crystal resonator and capacitors (CL1 and CL2) as close to pins Xin and Xout as possible. Furthermore, to avoid inductance so that oscillation is correct, use the points where the capacitors are connected to the crystal resonator in common and do not place wiring patterns close to these components.

Since the design of the user board is closely connected with the effective characteristics of the crystal resonator, refer to the example of connection of the crystal resonator that is introduced in this section and perform thorough evaluation on the user side as well. The rated value of the crystal resonator will vary with the floating capacitances and so on of the crystal resonator and mounted circuit, so proceed with decisions on the basis of full discussions with the maker of the crystal resonator. Ensure that voltages applied to the clock pins do not exceed the maximum rated values.

Although the feedback resistor is included in this LSI, an external feedback resistor may be required in some cases. This depends on the characteristics of the crystal resonator.

Set the parameters (of resistors and capacitors) with thorough evaluation on the user side.

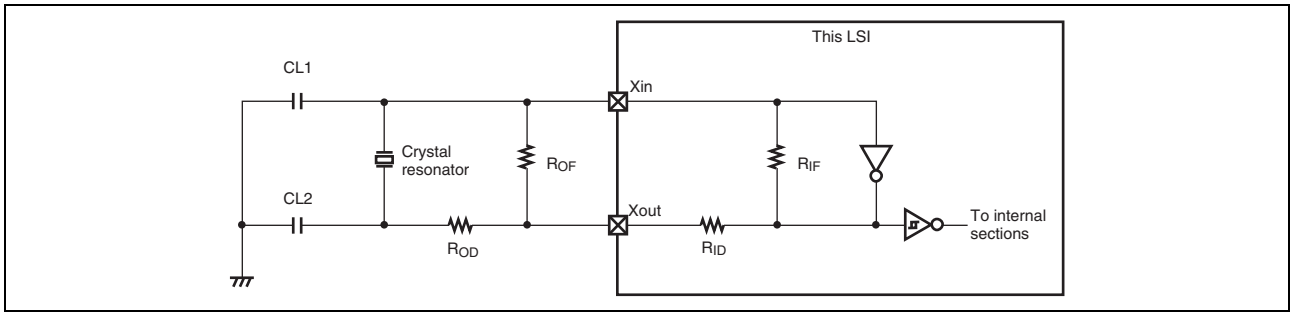


Figure 6.3 Example of the Connection of a Crystal Resonator

6.6.3 In the Case of Not Using the Clock Pin

In cases where the pins are not in use, fix the level on the Xin pin by connecting it to a pull-up resistor, pull-down resistor, the power supply, or to the ground, and leave the Xout pin open-circuit.

6.7 Oscillation Stabilizing Time

6.7.1 Oscillation Stabilizing Time of the On-chip Crystal Oscillator

In the case of using a crystal resonator, wait for the oscillation stabilizing time of the on-chip oscillation circuit at the following cases, to keep the oscillation stabilizing time of the on-chip crystal oscillator (in the case of inputting an external clock input, it is not necessary).

- Power on
- Releasing the software standby mode or deep standby mode by RES# pin
- Changing from halting oscillation to running oscillation by power-on reset or register setting (AUDIO_X1)
- Changing from halting oscillation to running oscillation by register setting (RTC_X1)
- Changing the gain of the on-chip crystal oscillator by RES# pin (EXTAL)

6.7.2 Oscillation Stabilizing Time of the PLL circuit

The clock from EXTAL is supplied to the PLL circuit. So, regardless of whether using a crystal resonator or inputting an external clock from EXTAL, wait for at least the oscillation stabilizing time at the following cases, to keep the oscillation stabilizing time of the PLL circuit.

- Power on (in the case of using the crystal resonator)/start inputting external clock (in the case of inputting the external clock)
- Releasing the software standby mode or deep standby mode by RES# pin

Note: The oscillation stabilizing time is kept by the counter running in the LSI at the following cases.

- Releasing the software standby mode or deep standby mode by the other than RES# pin
- Changing the gain of the on-chip crystal oscillator by the register setting (EXTAL)

6.8 Notes on Board Design

6.8.1 Note on Using a PLL Oscillation Circuit

In the PLLVcc connection pattern for the PLL, signal lines from the board power supply pins must be as short as possible and pattern width must be as wide as possible to reduce inductive interferences.

Since the analog power supply pins of the PLL are sensitive to the noise, the system may malfunction due to inductive interference at the other power supply pins. To prevent such malfunction, the analog power supply pins and the digital power supply pins Vcc and PVcc should not supply the same resources on the board if at all possible.

6.9 Definition of Modulation Rate and Frequency in the SSCG Specification

The SSCG circuit can be used to decrease the peak value of electromagnetic interference noise by frequency modulation, i.e. by slightly modulating the output frequency. In this case, the rate of change in the frequency and the size of the change to the input clock frequency are defined as the modulation rate and modulation frequency, respectively.
Figure 6.4 shows the modulation rate and modulation frequency.

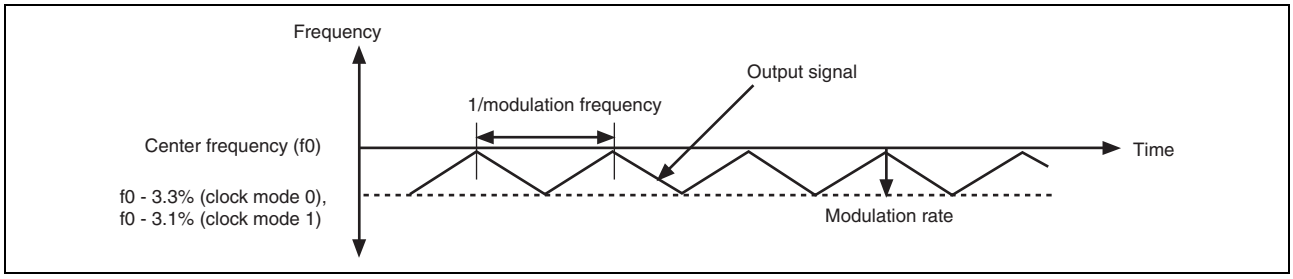


Figure 6.4 Definition of SSCG Modulation Rate and Frequency

6.10 Clock Signals

6.10.1 Clock Signals for the System and Realtime Clock

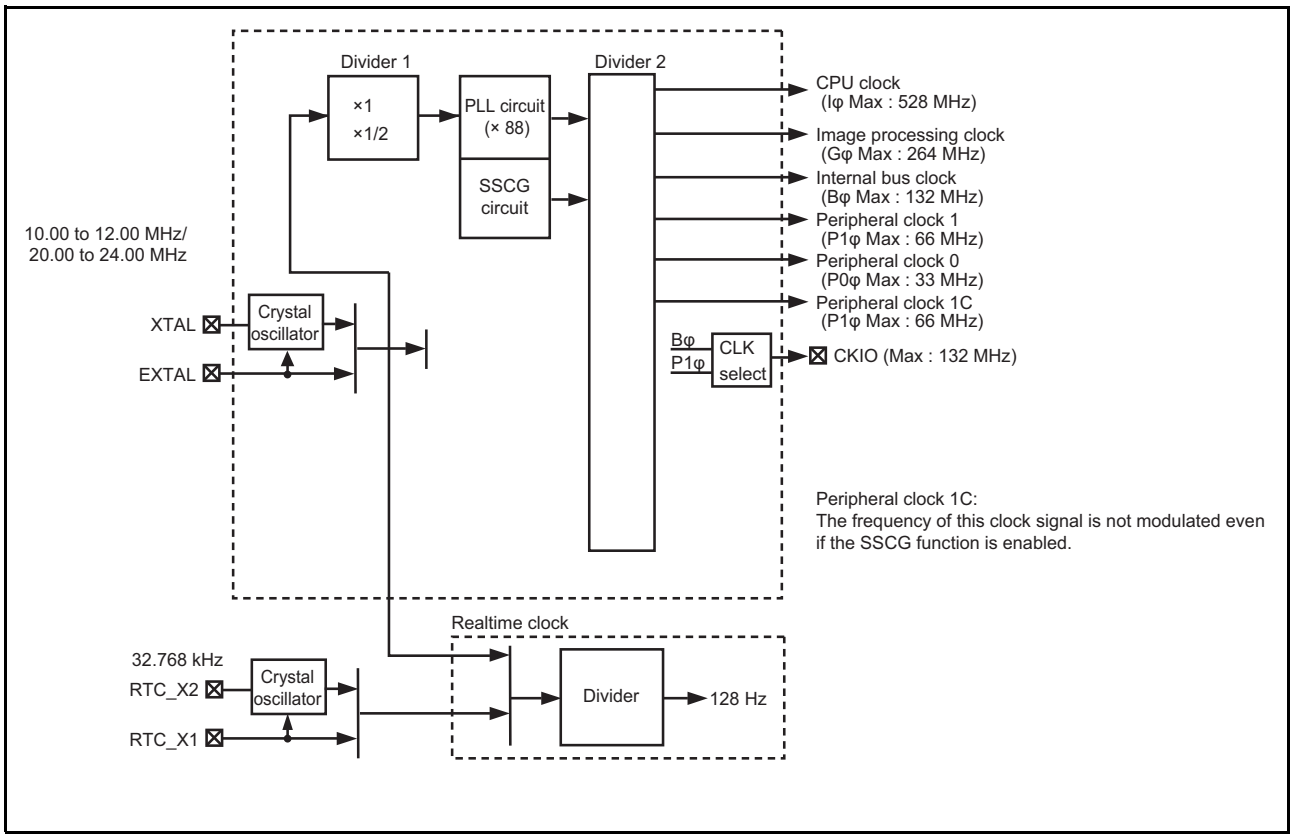


Figure 6.5 Clock Signals for the System and Realtime Clock

6.10.2 Audio and USB Clock Signals

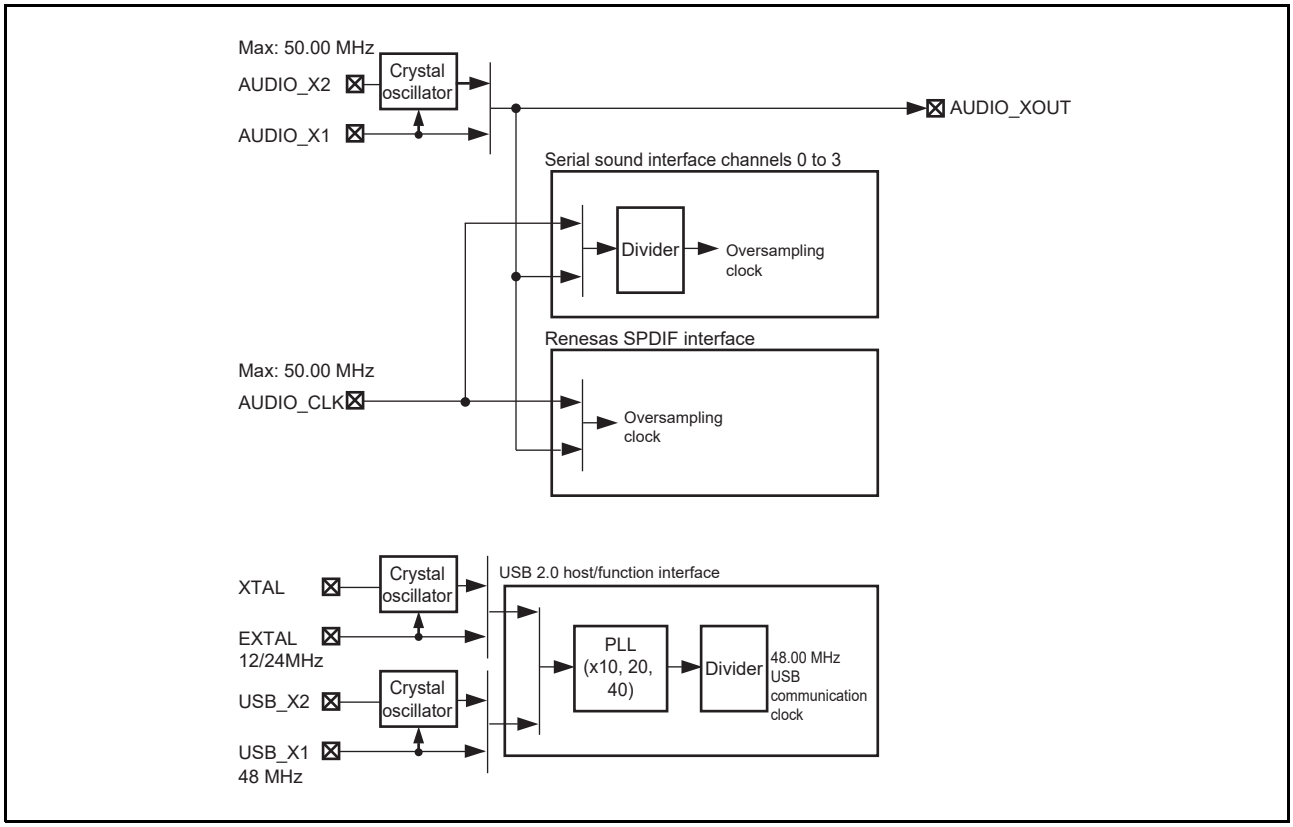


Figure 6.6 Audio and USB Clock Signals

6.10.3 Video Image Clock Signals (Channel 0)

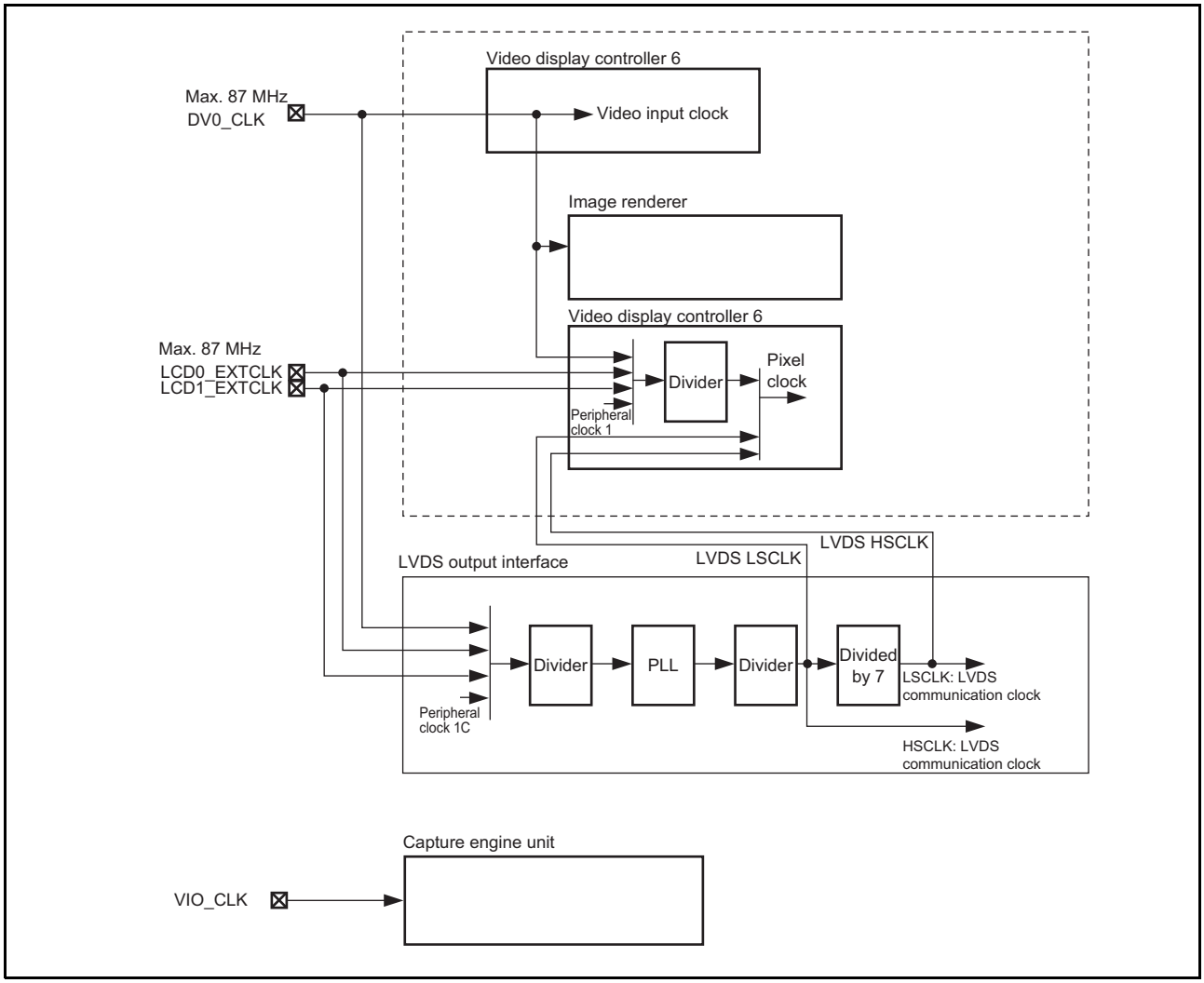


Figure 6.7 Video Image Clock Signals (Channel 0)

6.10.4 Other Clock Signals 1

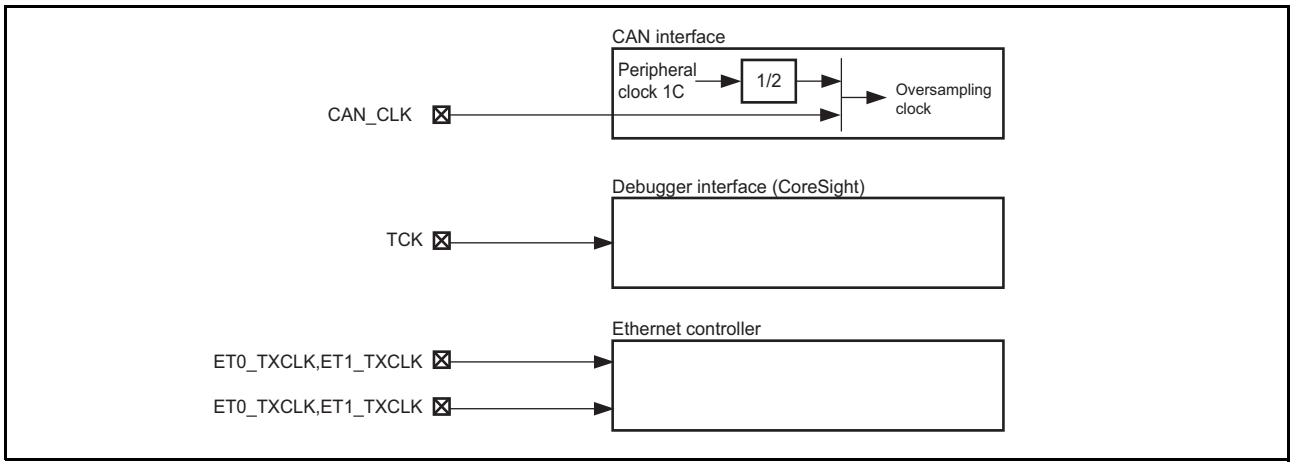


Figure 6.8 Clock Signals for Other Modules 1

6.10.5 Other Clock Signals 2

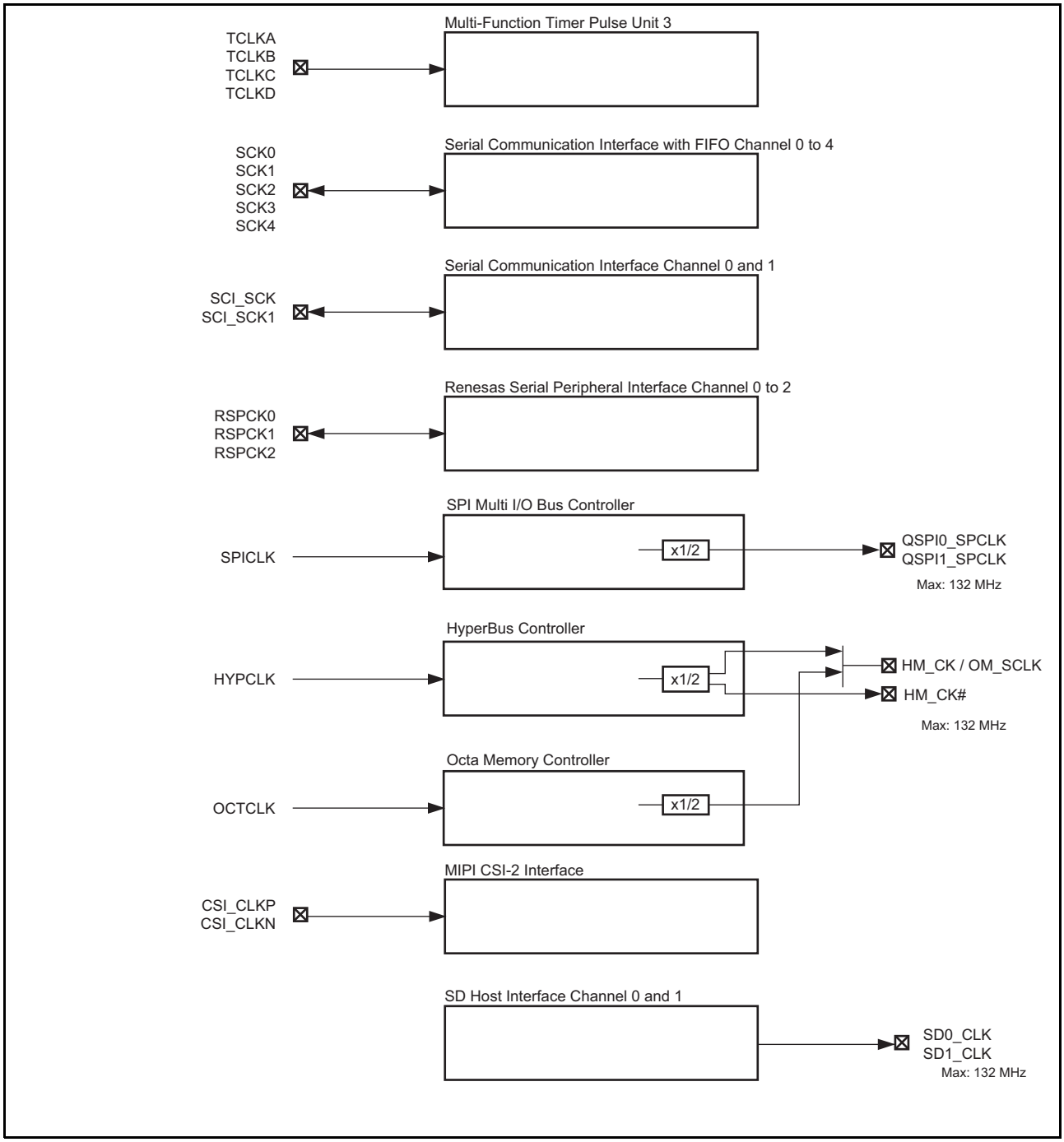


Figure 6.9 Clock Signals for Other Modules 2

6.10.6 Internal Clock Signals (1)

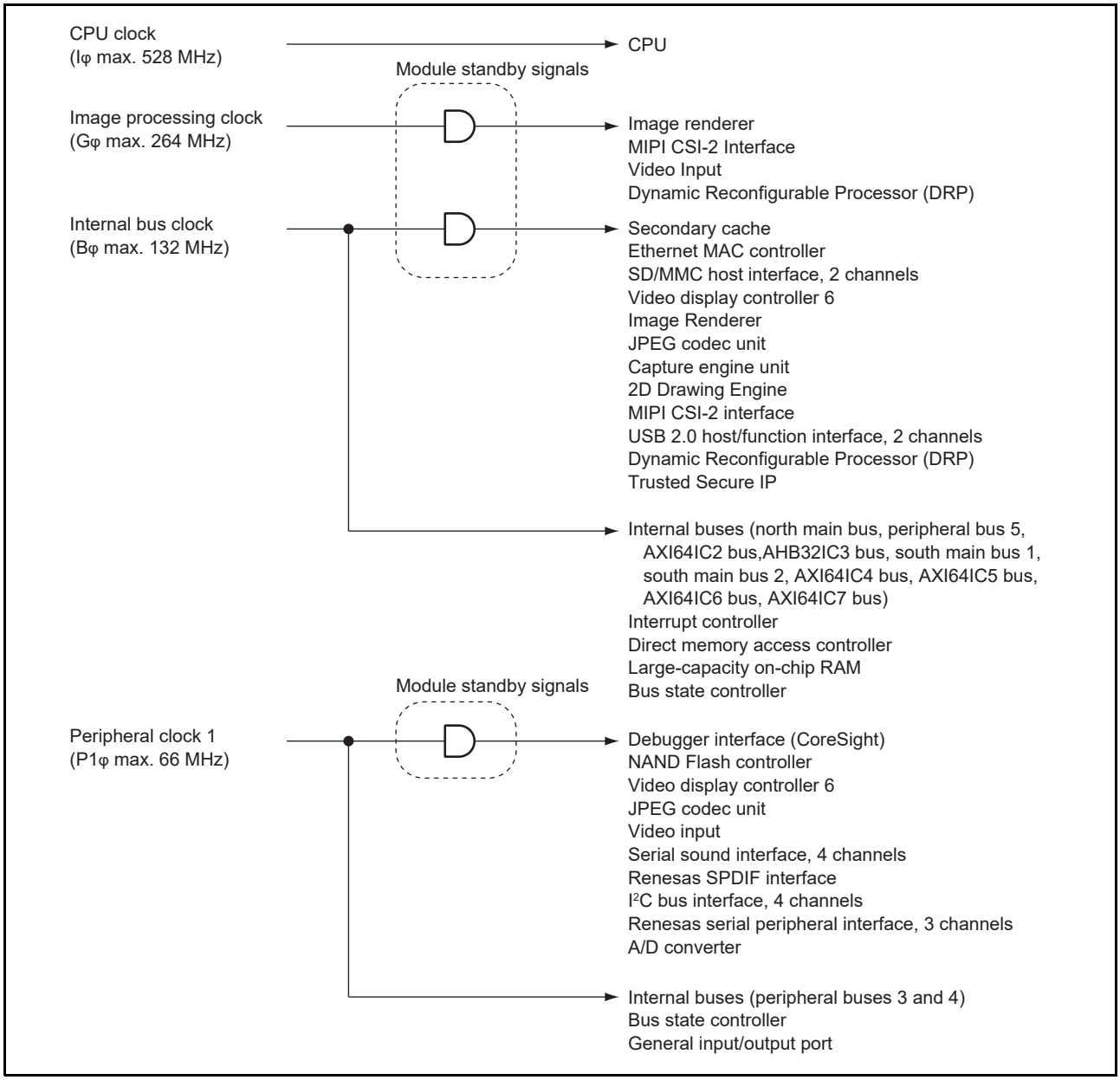


Figure 6.10 Distribution of Internal Clock Signals

6.10.7 Internal Clock Signals (2)

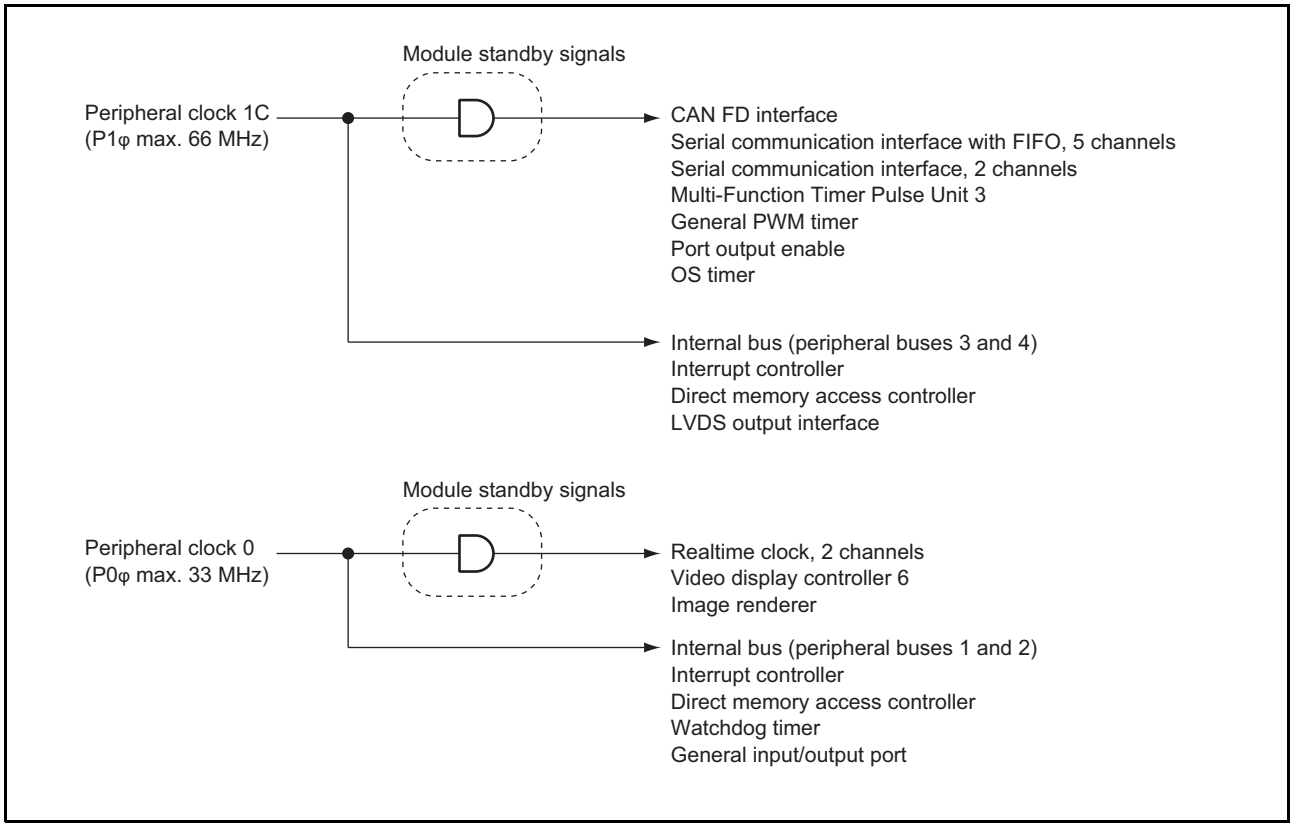


Figure 6.11 Distribution of Internal Clock Signals (2)

6.10.8 Internal Clock Signals (3)

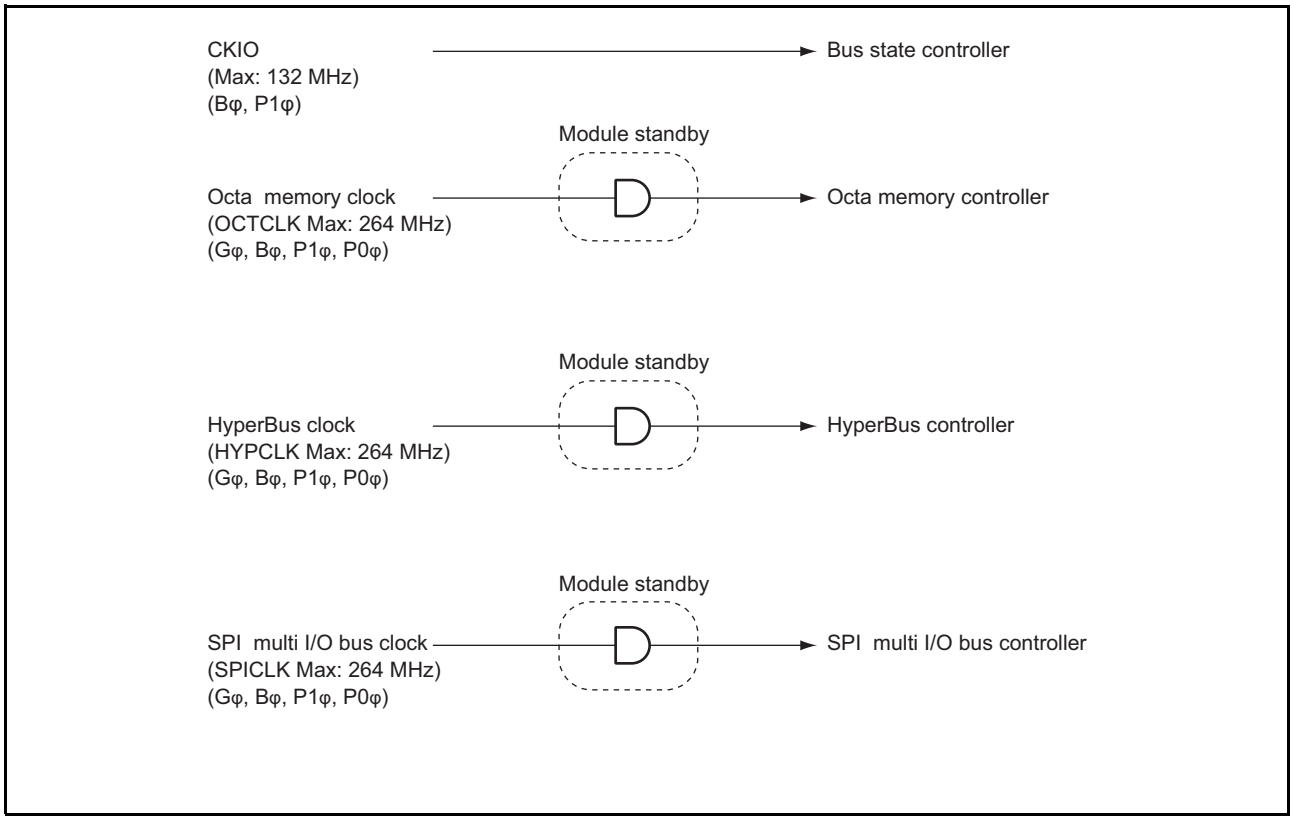


Figure 6.12 Distribution of Internal Clock Signals (3)

7. Interrupt Controller

The interrupt controller ascertains the priority of interrupt sources and controls interrupt requests to the CPU. The interrupt controller registers set the order of priority of each interrupt, allowing the user to process interrupt requests according to the user-set priority.

7.1 Features

- 32 levels of interrupt priority can be set.
  By setting the interrupt priority registers, the priorities of IRQ interrupts, on-chip peripheral module interrupts, and pin interrupts can be selected from 32 levels for request sources.
- NMI noise canceler function
  An NMI input-level bit indicates the NMI pin state. By reading this bit in the interrupt exception service routine, the pin state can be checked, enabling it to be used as the noise canceler function.
- Arm PrimeCell® generic interrupt controller (GIC-400)*

Note: * The GIC-400 supports version 2 of the specification for the architecture of the Arm generic interrupt controller (GIC).

Figure 7.1 shows a block diagram.

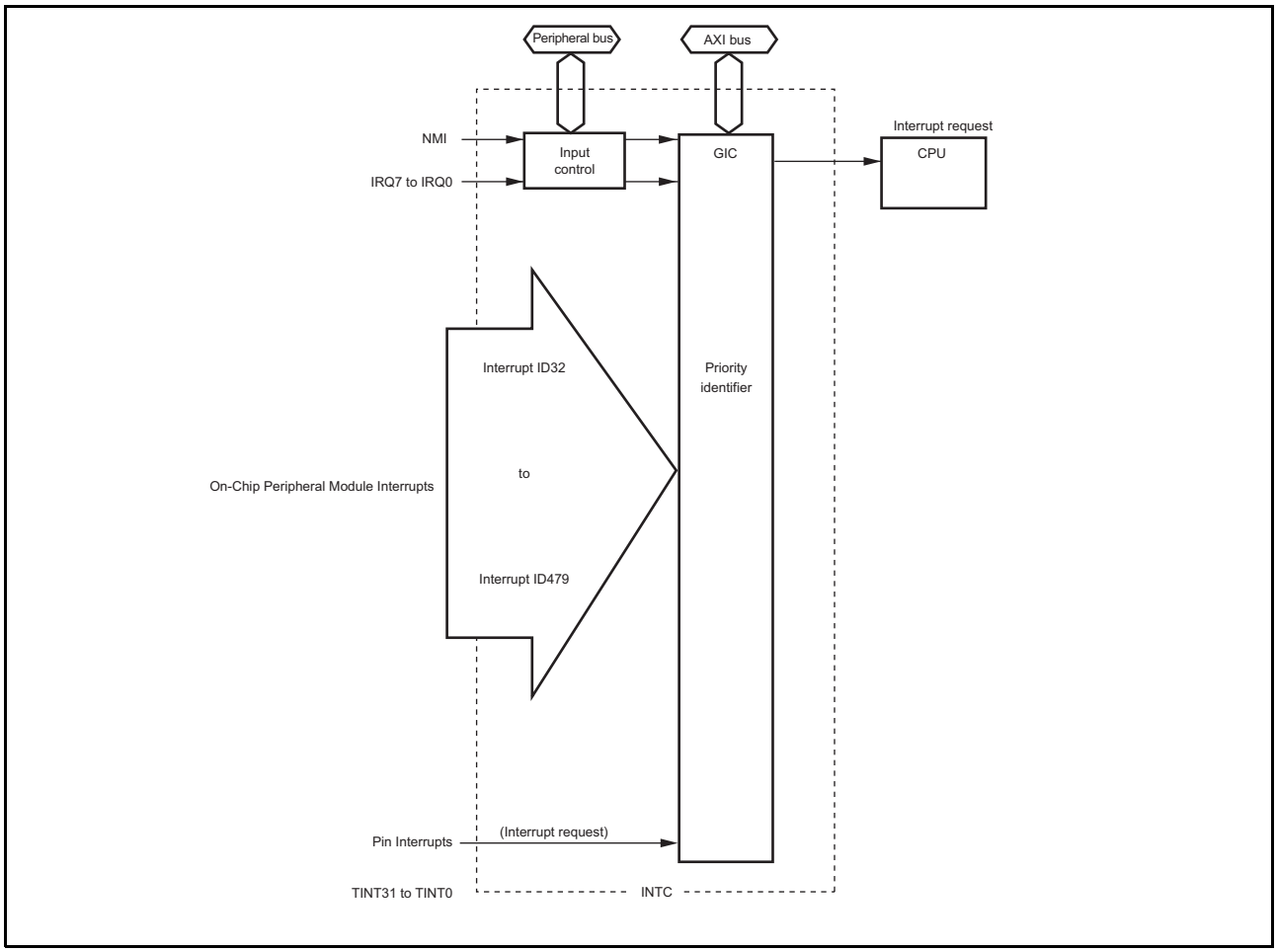


Figure 7.1 Block Diagram

7.2 Input/Output Pins

Table 7.1 shows the pin configuration.

Table 7.1 Pin Configuration

Pin Name	Symbol	I/O	Function
Nonmaskable interrupt input pin	NMI	Input	Input of nonmaskable interrupt request signal
Interrupt request input pins	IRQ7 to IRQ0	Input	Input of maskable interrupt request signals
	TINT31 to TINT0	Input	Input of maskable interrupt request signals

7.3 Register Descriptions

Table 7.2 shows the register configuration. These registers are used to set the interrupt priorities and control detection of the external interrupt input signal.
For a description of the registers other than interrupt control register 0, interrupt control register 1, and IRQ interrupt request register, see the Arm Generic Interrupt Controller Architecture Specification and the PrimeCell® Generic Interrupt Controller (GIC-400) Technical Reference Manual from Arm.

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt control register 0	ICR0	RW	*1	H'FCFE F800	16
Interrupt control register 1	ICR1	RW	H'0000	H'FCFE F802	16
IRQ Interrupt Request Register	IRQRR	RW *2	H'0000	H'FCFE F804	16
Divider Controller Register	GICD_CTLR	RW	H'00000000	H'E822 1000	32
Interrupt controller type register	GICD_TYPER	RO	H'0000FC0F *3	H'E822 1004	32
Distributor implementer identification register	GICD_IIDR	RO	H'0200143B	H'E822 1008	32
Interrupt group register 0	GICD_IGROUPR0	RW	H'00000000	H'E822 1080	32
Interrupt group register 1	GICD_IGROUPR1	RW	H'00000000	H'E822 1084	32
Interrupt group register 2	GICD_IGROUPR2	RW	H'00000000	H'E822 1088	32
Interrupt group register 3	GICD_IGROUPR3	RW	H'00000000	H'E822 108C	32
Interrupt group register 4	GICD_IGROUPR4	RW	H'00000000	H'E822 1090	32
Interrupt group register 5	GICD_IGROUPR5	RW	H'00000000	H'E822 1094	32
Interrupt group register 6	GICD_IGROUPR6	RW	H'00000000	H'E822 1098	32
Interrupt group register 7	GICD_IGROUPR7	RW	H'00000000	H'E822 109C	32
Interrupt group register 8	GICD_IGROUPR8	RW	H'00000000	H'E822 10A0	32
Interrupt group register 9	GICD_IGROUPR9	RW	H'00000000	H'E822 10A4	32
Interrupt group register 10	GICD_IGROUPR10	RW	H'00000000	H'E822 10A8	32
Interrupt group register 11	GICD_IGROUPR11	RW	H'00000000	H'E822 10AC	32
Interrupt group register 12	GICD_IGROUPR12	RW	H'00000000	H'E822 10B0	32
Interrupt group register 13	GICD_IGROUPR13	RW	H'00000000	H'E822 10B4	32
Interrupt group register 14	GICD_IGROUPR14	RW	H'00000000	H'E822 10B8	32
Interrupt group register 15	GICD_IGROUPR15	RW	H'00000000	H'E822 10BC	32
Interrupt set-enable register 0	GICD_ISENABLER0	RW	H'0000FFFF	H'E822 1100	32
Interrupt set-enable register 1	GICD_ISENABLER1	RW	H'00000000	H'E822 1104	32
Interrupt set-enable register 2	GICD_ISENABLER2	RW	H'00000000	H'E822 1108	32
Interrupt set-enable register 3	GICD_ISENABLER3	RW	H'00000000	H'E822 110C	32
Interrupt set-enable register 4	GICD_ISENABLER4	RW	H'00000000	H'E822 1110	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt set-enable register 5	GICD_ISENABLER5	RW	H'00000000	H'E822 1114	32
Interrupt set-enable register 6	GICD_ISENABLER6	RW	H'00000000	H'E822 1118	32
Interrupt set-enable register 7	GICD_ISENABLER7	RW	H'00000000	H'E822 111C	32
Interrupt set-enable register 8	GICD_ISENABLER8	RW	H'00000000	H'E822 1120	32
Interrupt set-enable register 9	GICD_ISENABLER9	RW	H'00000000	H'E822 1124	32
Interrupt set-enable register 10	GICD_ISENABLER10	RW	H'00000000	H'E822 1128	32
Interrupt set-enable register 11	GICD_ISENABLER11	RW	H'00000000	H'E822 112C	32
Interrupt set-enable register 12	GICD_ISENABLER12	RW	H'00000000	H'E822 1130	32
Interrupt set-enable register 13	GICD_ISENABLER13	RW	H'00000000	H'E822 1134	32
Interrupt set-enable register 14	GICD_ISENABLER14	RW	H'00000000	H'E822 1138	32
Interrupt set-enable register 15	GICD_ISENABLER15	RW	H'00000000	H'E822 113C	32
Interrupt clear-enable register 0	GICD_ICENABLER0	RW	H'00000000	H'E822 1180	32
Interrupt clear-enable register 1	GICD_ICENABLER1	RW	H'00000000	H'E822 1184	32
Interrupt clear-enable register 2	GICD_ICENABLER2	RW	H'00000000	H'E822 1188	32
Interrupt clear-enable register 3	GICD_ICENABLER3	RW	H'00000000	H'E822 118C	32
Interrupt clear-enable register 4	GICD_ICENABLER4	RW	H'00000000	H'E822 1190	32
Interrupt clear-enable register 5	GICD_ICENABLER5	RW	H'00000000	H'E822 1194	32
Interrupt clear-enable register 6	GICD_ICENABLER6	RW	H'00000000	H'E822 1198	32
Interrupt clear-enable register 7	GICD_ICENABLER7	RW	H'00000000	H'E822 119C	32
Interrupt clear-enable register 8	GICD_ICENABLER8	RW	H'00000000	H'E822 11A0	32
Interrupt clear-enable register 9	GICD_ICENABLER9	RW	H'00000000	H'E822 11A4	32
Interrupt clear-enable register 10	GICD_ICENABLER10	RW	H'00000000	H'E822 11A8	32
Interrupt clear-enable register 11	GICD_ICENABLER11	RW	H'00000000	H'E822 11AC	32
Interrupt clear-enable register 12	GICD_ICENABLER12	RW	H'00000000	H'E822 11B0	32
Interrupt clear-enable register 13	GICD_ICENABLER13	RW	H'00000000	H'E822 11B4	32
Interrupt clear-enable register 14	GICD_ICENABLER14	RW	H'00000000	H'E822 11B8	32
Interrupt clear-enable register 15	GICD_ICENABLER15	RW	H'00000000	H'E822 11BC	32
Interrupt set-pending register 0	GICD_ISPENDR0	RW	H'00000000	H'E822 1200	32
Interrupt set-pending register 1	GICD_ISPENDR1	RW	H'00000000	H'E822 1204	32
Interrupt set-pending register 2	GICD_ISPENDR2	RW	H'00000000	H'E822 1208	32
Interrupt set-pending register 3	GICD_ISPENDR3	RW	H'00000000	H'E822 120C	32
Interrupt set-pending register 4	GICD_ISPENDR4	RW	H'00000000	H'E822 1210	32
Interrupt set-pending register 5	GICD_ISPENDR5	RW	H'00000000	H'E822 1214	32
Interrupt set-pending register 6	GICD_ISPENDR6	RW	H'00000000	H'E822 1218	32
Interrupt set-pending register 7	GICD_ISPENDR7	RW	H'00000000	H'E822 121C	32
Interrupt set-pending register 8	GICD_ISPENDR8	RW	H'00000000	H'E822 1220	32
Interrupt set-pending register 9	GICD_ISPENDR9	RW	H'00000000	H'E822 1224	32
Interrupt set-pending register 10	GICD_ISPENDR10	RW	H'00000000	H'E822 1228	32
Interrupt set-pending register 11	GICD_ISPENDR11	RW	H'00000000	H'E822 122C	32
Interrupt set-pending register 12	GICD_ISPENDR12	RW	H'00000000	H'E822 1230	32
Interrupt set-pending register 13	GICD_ISPENDR13	RW	H'00000000	H'E822 1234	32
Interrupt set-pending register 14	GICD_ISPENDR14	RW	H'00000000	H'E822 1238	32
Interrupt set-pending register 15	GICD_ISPENDR15	RW	H'00000000	H'E822 123C	32
Interrupt clear-pending register 0	GICD_ICPENDR0	RW	H'00000000	H'E822 1280	32
Interrupt clear-pending register 1	GICD_ICPENDR1	RW	H'00000000	H'E822 1284	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt clear-pending register 2	GICD_ICPENDR2	RW	H'00000000	H'E822 1288	32
Interrupt clear-pending register 3	GICD_ICPENDR3	RW	H'00000000	H'E822 128C	32
Interrupt clear-pending register 4	GICD_ICPENDR4	RW	H'00000000	H'E822 1290	32
Interrupt clear-pending register 5	GICD_ICPENDR5	RW	H'00000000	H'E822 1294	32
Interrupt clear-pending register 6	GICD_ICPENDR6	RW	H'00000000	H'E822 1298	32
Interrupt clear-pending register 7	GICD_ICPENDR7	RW	H'00000000	H'E822 129C	32
Interrupt clear-pending register 8	GICD_ICPENDR8	RW	H'00000000	H'E822 12A0	32
Interrupt clear-pending register 9	GICD_ICPENDR9	RW	H'00000000	H'E822 12A4	32
Interrupt clear-pending register 10	GICD_ICPENDR10	RW	H'00000000	H'E822 12A8	32
Interrupt clear-pending register 11	GICD_ICPENDR11	RW	H'00000000	H'E822 12AC	32
Interrupt clear-pending register 12	GICD_ICPENDR12	RW	H'00000000	H'E822 12B0	32
Interrupt clear-pending register 13	GICD_ICPENDR13	RW	H'00000000	H'E822 12B4	32
Interrupt clear-pending register 14	GICD_ICPENDR14	RW	H'00000000	H'E822 12B8	32
Interrupt clear-pending register 15	GICD_ICPENDR15	RW	H'00000000	H'E822 12BC	32
Active bit set register 0	GICD_ISACTIVER0	RW	H'00000000	H'E822 1300	32
Active bit set register 1	GICD_ISACTIVER1	RW	H'00000000	H'E822 1304	32
Active bit set register 2	GICD_ISACTIVER2	RW	H'00000000	H'E822 1308	32
Active bit set register 3	GICD_ISACTIVER3	RW	H'00000000	H'E822 130C	32
Active bit set register 4	GICD_ISACTIVER4	RW	H'00000000	H'E822 1310	32
Active bit set register 5	GICD_ISACTIVER5	RW	H'00000000	H'E822 1314	32
Active bit set register 6	GICD_ISACTIVER6	RW	H'00000000	H'E822 1318	32
Active bit set register 7	GICD_ISACTIVER7	RW	H'00000000	H'E822 131C	32
Active bit set register 8	GICD_ISACTIVER8	RW	H'00000000	H'E822 1320	32
Active bit set register 9	GICD_ISACTIVER9	RW	H'00000000	H'E822 1324	32
Active bit set register 10	GICD_ISACTIVER10	RW	H'00000000	H'E822 1328	32
Active bit set register 11	GICD_ISACTIVER11	RW	H'00000000	H'E822 132C	32
Active bit set register 12	GICD_ISACTIVER12	RW	H'00000000	H'E822 1330	32
Active bit set register 13	GICD_ISACTIVER13	RW	H'00000000	H'E822 1334	32
Active bit set register 14	GICD_ISACTIVER14	RW	H'00000000	H'E822 1338	32
Active bit set register 15	GICD_ISACTIVER15	RW	H'00000000	H'E822 133C	32
Active bit clear register 0	GICD_ICACTIVER0	RW	H'00000000	H'E822 1380	32
Active bit clear register 1	GICD_ICACTIVER1	RW	H'00000000	H'E822 1384	32
Active bit clear register 2	GICD_ICACTIVER2	RW	H'00000000	H'E822 1388	32
Active bit clear register 3	GICD_ICACTIVER3	RW	H'00000000	H'E822 138C	32
Active bit clear register 4	GICD_ICACTIVER4	RW	H'00000000	H'E822 1390	32
Active bit clear register 5	GICD_ICACTIVER5	RW	H'00000000	H'E822 1394	32
Active bit clear register 6	GICD_ICACTIVER6	RW	H'00000000	H'E822 1398	32
Active bit clear register 7	GICD_ICACTIVER7	RW	H'00000000	H'E822 139C	32
Active bit clear register 8	GICD_ICACTIVER8	RW	H'00000000	H'E822 13A0	32
Active bit clear register 9	GICD_ICACTIVER9	RW	H'00000000	H'E822 13A4	32
Active bit clear register 10	GICD_ICACTIVER10	RW	H'00000000	H'E822 13A8	32
Active bit clear register 11	GICD_ICACTIVER11	RW	H'00000000	H'E822 13AC	32
Active bit clear register 12	GICD_ICACTIVER12	RW	H'00000000	H'E822 13B0	32
Active bit clear register 13	GICD_ICACTIVER13	RW	H'00000000	H'E822 13B4	32
Active bit clear register 14	GICD_ICACTIVER14	RW	H'00000000	H'E822 13B8	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Active bit clear register 15	GICD_ICACTIVER15	RW	H'00000000	H'E822 13BC	32
Interrupt priority register 0	GICD_IPRIORITYR0	RW	H'00000000	H'E822 1400	32
Interrupt priority register 1	GICD_IPRIORITYR1	RW	H'00000000	H'E822 1404	32
Interrupt priority register 2	GICD_IPRIORITYR2	RW	H'00000000	H'E822 1408	32
Interrupt priority register 3	GICD_IPRIORITYR3	RW	H'00000000	H'E822 140C	32
Interrupt priority register 4	GICD_IPRIORITYR4	RW	H'00000000	H'E822 1410	32
Interrupt priority register 5	GICD_IPRIORITYR5	RW	H'00000000	H'E822 1414	32
Interrupt priority register 6	GICD_IPRIORITYR6	RW	H'00000000	H'E822 1418	32
Interrupt priority register 7	GICD_IPRIORITYR7	RW	H'00000000	H'E822 141C	32
Interrupt priority register 8	GICD_IPRIORITYR8	RW	H'00000000	H'E822 1420	32
Interrupt priority register 9	GICD_IPRIORITYR9	RW	H'00000000	H'E822 1424	32
Interrupt priority register 10	GICD_IPRIORITYR10	RW	H'00000000	H'E822 1428	32
Interrupt priority register 11	GICD_IPRIORITYR11	RW	H'00000000	H'E822 142C	32
Interrupt priority register 12	GICD_IPRIORITYR12	RW	H'00000000	H'E822 1430	32
Interrupt priority register 13	GICD_IPRIORITYR13	RW	H'00000000	H'E822 1434	32
Interrupt priority register 14	GICD_IPRIORITYR14	RW	H'00000000	H'E822 1438	32
Interrupt priority register 15	GICD_IPRIORITYR15	RW	H'00000000	H'E822 143C	32
Interrupt priority register 16	GICD_IPRIORITYR16	RW	H'00000000	H'E822 1440	32
Interrupt priority register 17	GICD_IPRIORITYR17	RW	H'00000000	H'E822 1444	32
Interrupt priority register 18	GICD_IPRIORITYR18	RW	H'00000000	H'E822 1448	32
Interrupt priority register 19	GICD_IPRIORITYR19	RW	H'00000000	H'E822 144C	32
Interrupt priority register 20	GICD_IPRIORITYR20	RW	H'00000000	H'E822 1450	32
Interrupt priority register 21	GICD_IPRIORITYR21	RW	H'00000000	H'E822 1454	32
Interrupt priority register 22	GICD_IPRIORITYR22	RW	H'00000000	H'E822 1458	32
Interrupt priority register 23	GICD_IPRIORITYR23	RW	H'00000000	H'E822 145C	32
Interrupt priority register 24	GICD_IPRIORITYR24	RW	H'00000000	H'E822 1460	32
Interrupt priority register 25	GICD_IPRIORITYR25	RW	H'00000000	H'E822 1464	32
Interrupt priority register 26	GICD_IPRIORITYR26	RW	H'00000000	H'E822 1468	32
Interrupt priority register 27	GICD_IPRIORITYR27	RW	H'00000000	H'E822 146C	32
Interrupt priority register 28	GICD_IPRIORITYR28	RW	H'00000000	H'E822 1470	32
Interrupt priority register 29	GICD_IPRIORITYR29	RW	H'00000000	H'E822 1474	32
Interrupt priority register 30	GICD_IPRIORITYR30	RW	H'00000000	H'E822 1478	32
Interrupt priority register 31	GICD_IPRIORITYR31	RW	H'00000000	H'E822 147C	32
Interrupt priority register 32	GICD_IPRIORITYR32	RW	H'00000000	H'E822 1480	32
Interrupt priority register 33	GICD_IPRIORITYR33	RW	H'00000000	H'E822 1484	32
Interrupt priority register 34	GICD_IPRIORITYR34	RW	H'00000000	H'E822 1488	32
Interrupt priority register 35	GICD_IPRIORITYR35	RW	H'00000000	H'E822 148C	32
Interrupt priority register 36	GICD_IPRIORITYR36	RW	H'00000000	H'E822 1490	32
Interrupt priority register 37	GICD_IPRIORITYR37	RW	H'00000000	H'E822 1494	32
Interrupt priority register 38	GICD_IPRIORITYR38	RW	H'00000000	H'E822 1498	32
Interrupt priority register 39	GICD_IPRIORITYR39	RW	H'00000000	H'E822 149C	32
Interrupt priority register 40	GICD_IPRIORITYR40	RW	H'00000000	H'E822 14A0	32
Interrupt priority register 41	GICD_IPRIORITYR41	RW	H'00000000	H'E822 14A4	32
Interrupt priority register 42	GICD_IPRIORITYR42	RW	H'00000000	H'E822 14A8	32
Interrupt priority register 43	GICD_IPRIORITYR43	RW	H'00000000	H'E822 14AC	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt priority register 44	GICD_IPRIORITYR44	RW	H'00000000	H'E822 14B0	32
Interrupt priority register 45	GICD_IPRIORITYR45	RW	H'00000000	H'E822 14B4	32
Interrupt priority register 46	GICD_IPRIORITYR46	RW	H'00000000	H'E822 14B8	32
Interrupt priority register 47	GICD_IPRIORITYR47	RW	H'00000000	H'E822 14BC	32
Interrupt priority register 48	GICD_IPRIORITYR48	RW	H'00000000	H'E822 14C0	32
Interrupt priority register 49	GICD_IPRIORITYR49	RW	H'00000000	H'E822 14C4	32
Interrupt priority register 50	GICD_IPRIORITYR50	RW	H'00000000	H'E822 14C8	32
Interrupt priority register 51	GICD_IPRIORITYR51	RW	H'00000000	H'E822 14CC	32
Interrupt priority register 52	GICD_IPRIORITYR52	RW	H'00000000	H'E822 14D0	32
Interrupt priority register 53	GICD_IPRIORITYR53	RW	H'00000000	H'E822 14D4	32
Interrupt priority register 54	GICD_IPRIORITYR54	RW	H'00000000	H'E822 14D8	32
Interrupt priority register 55	GICD_IPRIORITYR55	RW	H'00000000	H'E822 14DC	32
Interrupt priority register 56	GICD_IPRIORITYR56	RW	H'00000000	H'E822 14E0	32
Interrupt priority register 57	GICD_IPRIORITYR57	RW	H'00000000	H'E822 14E4	32
Interrupt priority register 58	GICD_IPRIORITYR58	RW	H'00000000	H'E822 14E8	32
Interrupt priority register 59	GICD_IPRIORITYR59	RW	H'00000000	H'E822 14EC	32
Interrupt priority register 60	GICD_IPRIORITYR60	RW	H'00000000	H'E822 14F0	32
Interrupt priority register 61	GICD_IPRIORITYR61	RW	H'00000000	H'E822 14F4	32
Interrupt priority register 62	GICD_IPRIORITYR62	RW	H'00000000	H'E822 14F8	32
Interrupt priority register 63	GICD_IPRIORITYR63	RW	H'00000000	H'E822 14FC	32
Interrupt priority register 64	GICD_IPRIORITYR64	RW	H'00000000	H'E822 1500	32
Interrupt priority register 65	GICD_IPRIORITYR65	RW	H'00000000	H'E822 1504	32
Interrupt priority register 66	GICD_IPRIORITYR66	RW	H'00000000	H'E822 1508	32
Interrupt priority register 67	GICD_IPRIORITYR67	RW	H'00000000	H'E822 150C	32
Interrupt priority register 68	GICD_IPRIORITYR68	RW	H'00000000	H'E822 1510	32
Interrupt priority register 69	GICD_IPRIORITYR69	RW	H'00000000	H'E822 1514	32
Interrupt priority register 70	GICD_IPRIORITYR70	RW	H'00000000	H'E822 1518	32
Interrupt priority register 71	GICD_IPRIORITYR71	RW	H'00000000	H'E822 151C	32
Interrupt priority register 72	GICD_IPRIORITYR72	RW	H'00000000	H'E822 1520	32
Interrupt priority register 73	GICD_IPRIORITYR73	RW	H'00000000	H'E822 1524	32
Interrupt priority register 74	GICD_IPRIORITYR74	RW	H'00000000	H'E822 1528	32
Interrupt priority register 75	GICD_IPRIORITYR75	RW	H'00000000	H'E822 152C	32
Interrupt priority register 76	GICD_IPRIORITYR76	RW	H'00000000	H'E822 1530	32
Interrupt priority register 77	GICD_IPRIORITYR77	RW	H'00000000	H'E822 1534	32
Interrupt priority register 78	GICD_IPRIORITYR78	RW	H'00000000	H'E822 1538	32
Interrupt priority register 79	GICD_IPRIORITYR79	RW	H'00000000	H'E822 153C	32
Interrupt priority register 80	GICD_IPRIORITYR80	RW	H'00000000	H'E822 1540	32
Interrupt priority register 81	GICD_IPRIORITYR81	RW	H'00000000	H'E822 1544	32
Interrupt priority register 82	GICD_IPRIORITYR82	RW	H'00000000	H'E822 1548	32
Interrupt priority register 83	GICD_IPRIORITYR83	RW	H'00000000	H'E822 154C	32
Interrupt priority register 84	GICD_IPRIORITYR84	RW	H'00000000	H'E822 1550	32
Interrupt priority register 85	GICD_IPRIORITYR85	RW	H'00000000	H'E822 1554	32
Interrupt priority register 86	GICD_IPRIORITYR86	RW	H'00000000	H'E822 1558	32
Interrupt priority register 87	GICD_IPRIORITYR87	RW	H'00000000	H'E822 155C	32
Interrupt priority register 88	GICD_IPRIORITYR88	RW	H'00000000	H'E822 1560	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt priority register 89	GICD_IPRIORITYR89	RW	H'00000000	H'E822 1564	32
Interrupt priority register 90	GICD_IPRIORITYR90	RW	H'00000000	H'E822 1568	32
Interrupt priority register 91	GICD_IPRIORITYR91	RW	H'00000000	H'E822 156C	32
Interrupt priority register 92	GICD_IPRIORITYR92	RW	H'00000000	H'E822 1570	32
Interrupt priority register 93	GICD_IPRIORITYR93	RW	H'00000000	H'E822 1574	32
Interrupt priority register 94	GICD_IPRIORITYR94	RW	H'00000000	H'E822 1578	32
Interrupt priority register 95	GICD_IPRIORITYR95	RW	H'00000000	H'E822 157C	32
Interrupt priority register 96	GICD_IPRIORITYR96	RW	H'00000000	H'E822 1580	32
Interrupt priority register 97	GICD_IPRIORITYR97	RW	H'00000000	H'E822 1584	32
Interrupt priority register 98	GICD_IPRIORITYR98	RW	H'00000000	H'E822 1588	32
Interrupt priority register 99	GICD_IPRIORITYR99	RW	H'00000000	H'E822 158C	32
Interrupt priority register 100	GICD_IPRIORITYR100	RW	H'00000000	H'E822 1590	32
Interrupt priority register 101	GICD_IPRIORITYR101	RW	H'00000000	H'E822 1594	32
Interrupt priority register 102	GICD_IPRIORITYR102	RW	H'00000000	H'E822 1598	32
Interrupt priority register 103	GICD_IPRIORITYR103	RW	H'00000000	H'E822 159C	32
Interrupt priority register 104	GICD_IPRIORITYR104	RW	H'00000000	H'E822 15A0	32
Interrupt priority register 105	GICD_IPRIORITYR105	RW	H'00000000	H'E822 15A4	32
Interrupt priority register 106	GICD_IPRIORITYR106	RW	H'00000000	H'E822 15A8	32
Interrupt priority register 107	GICD_IPRIORITYR107	RW	H'00000000	H'E822 15AC	32
Interrupt priority register 108	GICD_IPRIORITYR108	RW	H'00000000	H'E822 15B0	32
Interrupt priority register 109	GICD_IPRIORITYR109	RW	H'00000000	H'E822 15B4	32
Interrupt priority register 110	GICD_IPRIORITYR110	RW	H'00000000	H'E822 15B8	32
Interrupt priority register 111	GICD_IPRIORITYR111	RW	H'00000000	H'E822 15BC	32
Interrupt priority register 112	GICD_IPRIORITYR112	RW	H'00000000	H'E822 15C0	32
Interrupt priority register 113	GICD_IPRIORITYR113	RW	H'00000000	H'E822 15C4	32
Interrupt priority register 114	GICD_IPRIORITYR114	RW	H'00000000	H'E822 15C8	32
Interrupt priority register 115	GICD_IPRIORITYR115	RW	H'00000000	H'E822 15CC	32
Interrupt priority register 116	GICD_IPRIORITYR116	RW	H'00000000	H'E822 15D0	32
Interrupt priority register 117	GICD_IPRIORITYR117	RW	H'00000000	H'E822 15D4	32
Interrupt priority register 118	GICD_IPRIORITYR118	RW	H'00000000	H'E822 15D8	32
Interrupt priority register 119	GICD_IPRIORITYR119	RW	H'00000000	H'E822 15DC	32
Interrupt priority register 120	GICD_IPRIORITYR120	RW	H'00000000	H'E822 15E0	32
Interrupt priority register 121	GICD_IPRIORITYR121	RW	H'00000000	H'E822 15E4	32
Interrupt priority register 122	GICD_IPRIORITYR122	RW	H'00000000	H'E822 15E8	32
Interrupt priority register 123	GICD_IPRIORITYR123	RW	H'00000000	H'E822 15EC	32
Interrupt priority register 124	GICD_IPRIORITYR124	RW	H'00000000	H'E822 15F0	32
Interrupt priority register 125	GICD_IPRIORITYR125	RW	H'00000000	H'E822 15F4	32
Interrupt priority register 126	GICD_IPRIORITYR126	RW	H'00000000	H'E822 15F8	32
Interrupt priority register 127	GICD_IPRIORITYR127	RW	H'00000000	H'E822 15FC	32
Interrupt processor target register 0	GICD_ITARGETSR0	RO	H'00000000	H'E822 1800	32
Interrupt processor target register 1	GICD_ITARGETSR1	RO	H'00000000	H'E822 1804	32
Interrupt processor target register 2	GICD_ITARGETSR2	RO	H'00000000	H'E822 1808	32
Interrupt processor target register 3	GICD_ITARGETSR3	RO	H'00000000	H'E822 180C	32
Interrupt processor target register 4	GICD_ITARGETSR4	RO	H'00000000	H'E822 1810	32
Interrupt processor target register 5	GICD_ITARGETSR5	RO	H'00000000	H'E822 1814	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt processor target register 6	GICD_ITARGETSR6	RO	H'00000000	H'E822 1818	32
Interrupt processor target register 7	GICD_ITARGETSR7	RO	H'00000000	H'E822 181C	32
Interrupt processor target register 8	GICD_ITARGETSR8	RW	H'00000000	H'E822 1820	32
Interrupt processor target register 9	GICD_ITARGETSR9	RW	H'00000000	H'E822 1824	32
Interrupt processor target register 10	GICD_ITARGETSR10	RW	H'00000000	H'E822 1828	32
Interrupt processor target register 11	GICD_ITARGETSR11	RW	H'00000000	H'E822 182C	32
Interrupt processor target register 12	GICD_ITARGETSR12	RW	H'00000000	H'E822 1830	32
Interrupt processor target register 13	GICD_ITARGETSR13	RW	H'00000000	H'E822 1834	32
Interrupt processor target register 14	GICD_ITARGETSR14	RW	H'00000000	H'E822 1838	32
Interrupt processor target register 15	GICD_ITARGETSR15	RW	H'00000000	H'E822 183C	32
Interrupt processor target register 16	GICD_ITARGETSR16	RW	H'00000000	H'E822 1840	32
Interrupt processor target register 17	GICD_ITARGETSR17	RW	H'00000000	H'E822 1844	32
Interrupt processor target register 18	GICD_ITARGETSR18	RW	H'00000000	H'E822 1848	32
Interrupt processor target register 19	GICD_ITARGETSR19	RW	H'00000000	H'E822 184C	32
Interrupt processor target register 20	GICD_ITARGETSR20	RW	H'00000000	H'E822 1850	32
Interrupt processor target register 21	GICD_ITARGETSR21	RW	H'00000000	H'E822 1854	32
Interrupt processor target register 22	GICD_ITARGETSR22	RW	H'00000000	H'E822 1858	32
Interrupt processor target register 23	GICD_ITARGETSR23	RW	H'00000000	H'E822 185C	32
Interrupt processor target register 24	GICD_ITARGETSR24	RW	H'00000000	H'E822 1860	32
Interrupt processor target register 25	GICD_ITARGETSR25	RW	H'00000000	H'E822 1864	32
Interrupt processor target register 26	GICD_ITARGETSR26	RW	H'00000000	H'E822 1868	32
Interrupt processor target register 27	GICD_ITARGETSR27	RW	H'00000000	H'E822 186C	32
Interrupt processor target register 28	GICD_ITARGETSR28	RW	H'00000000	H'E822 1870	32
Interrupt processor target register 29	GICD_ITARGETSR29	RW	H'00000000	H'E822 1874	32
Interrupt processor target register 30	GICD_ITARGETSR30	RW	H'00000000	H'E822 1878	32
Interrupt processor target register 31	GICD_ITARGETSR31	RW	H'00000000	H'E822 187C	32
Interrupt processor target register 32	GICD_ITARGETSR32	RW	H'00000000	H'E822 1880	32
Interrupt processor target register 33	GICD_ITARGETSR33	RW	H'00000000	H'E822 1884	32
Interrupt processor target register 34	GICD_ITARGETSR34	RW	H'00000000	H'E822 1888	32
Interrupt processor target register 35	GICD_ITARGETSR35	RW	H'00000000	H'E822 188C	32
Interrupt processor target register 36	GICD_ITARGETSR36	RW	H'00000000	H'E822 1890	32
Interrupt processor target register 37	GICD_ITARGETSR37	RW	H'00000000	H'E822 1894	32
Interrupt processor target register 38	GICD_ITARGETSR38	RW	H'00000000	H'E822 1898	32
Interrupt processor target register 39	GICD_ITARGETSR39	RW	H'00000000	H'E822 189C	32
Interrupt processor target register 40	GICD_ITARGETSR40	RW	H'00000000	H'E822 18A0	32
Interrupt processor target register 41	GICD_ITARGETSR41	RW	H'00000000	H'E822 18A4	32
Interrupt processor target register 42	GICD_ITARGETSR42	RW	H'00000000	H'E822 18A8	32
Interrupt processor target register 43	GICD_ITARGETSR43	RW	H'00000000	H'E822 18AC	32
Interrupt processor target register 44	GICD_ITARGETSR44	RW	H'00000000	H'E822 18B0	32
Interrupt processor target register 45	GICD_ITARGETSR45	RW	H'00000000	H'E822 18B4	32
Interrupt processor target register 46	GICD_ITARGETSR46	RW	H'00000000	H'E822 18B8	32
Interrupt processor target register 47	GICD_ITARGETSR47	RW	H'00000000	H'E822 18BC	32
Interrupt processor target register 48	GICD_ITARGETSR48	RW	H'00000000	H'E822 18C0	32
Interrupt processor target register 49	GICD_ITARGETSR49	RW	H'00000000	H'E822 18C4	32
Interrupt processor target register 50	GICD_ITARGETSR50	RW	H'00000000	H'E822 18C8	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt processor target register 51	GICD_ITARGETSR51	RW	H'00000000	H'E822 18CC	32
Interrupt processor target register 52	GICD_ITARGETSR52	RW	H'00000000	H'E822 18D0	32
Interrupt processor target register 53	GICD_ITARGETSR53	RW	H'00000000	H'E822 18D4	32
Interrupt processor target register 54	GICD_ITARGETSR54	RW	H'00000000	H'E822 18D8	32
Interrupt processor target register 55	GICD_ITARGETSR55	RW	H'00000000	H'E822 18DC	32
Interrupt processor target register 56	GICD_ITARGETSR56	RW	H'00000000	H'E822 18E0	32
Interrupt processor target register 57	GICD_ITARGETSR57	RW	H'00000000	H'E822 18E4	32
Interrupt processor target register 58	GICD_ITARGETSR58	RW	H'00000000	H'E822 18E8	32
Interrupt processor target register 59	GICD_ITARGETSR59	RW	H'00000000	H'E822 18EC	32
Interrupt processor target register 60	GICD_ITARGETSR60	RW	H'00000000	H'E822 18F0	32
Interrupt processor target register 61	GICD_ITARGETSR61	RW	H'00000000	H'E822 18F4	32
Interrupt processor target register 62	GICD_ITARGETSR62	RW	H'00000000	H'E822 18F8	32
Interrupt processor target register 63	GICD_ITARGETSR63	RW	H'00000000	H'E822 18FC	32
Interrupt processor target register 64	GICD_ITARGETSR64	RW	H'00000000	H'E822 1900	32
Interrupt processor target register 65	GICD_ITARGETSR65	RW	H'00000000	H'E822 1904	32
Interrupt processor target register 66	GICD_ITARGETSR66	RW	H'00000000	H'E822 1908	32
Interrupt processor target register 67	GICD_ITARGETSR67	RW	H'00000000	H'E822 190C	32
Interrupt processor target register 68	GICD_ITARGETSR68	RW	H'00000000	H'E822 1910	32
Interrupt processor target register 69	GICD_ITARGETSR69	RW	H'00000000	H'E822 1914	32
Interrupt processor target register 70	GICD_ITARGETSR70	RW	H'00000000	H'E822 1918	32
Interrupt processor target register 71	GICD_ITARGETSR71	RW	H'00000000	H'E822 191C	32
Interrupt processor target register 72	GICD_ITARGETSR72	RW	H'00000000	H'E822 1920	32
Interrupt processor target register 73	GICD_ITARGETSR73	RW	H'00000000	H'E822 1924	32
Interrupt processor target register 74	GICD_ITARGETSR74	RW	H'00000000	H'E822 1928	32
Interrupt processor target register 75	GICD_ITARGETSR75	RW	H'00000000	H'E822 192C	32
Interrupt processor target register 76	GICD_ITARGETSR76	RW	H'00000000	H'E822 1930	32
Interrupt processor target register 77	GICD_ITARGETSR77	RW	H'00000000	H'E822 1934	32
Interrupt processor target register 78	GICD_ITARGETSR78	RW	H'00000000	H'E822 1938	32
Interrupt processor target register 79	GICD_ITARGETSR79	RW	H'00000000	H'E822 193C	32
Interrupt processor target register 80	GICD_ITARGETSR80	RW	H'00000000	H'E822 1940	32
Interrupt processor target register 81	GICD_ITARGETSR81	RW	H'00000000	H'E822 1944	32
Interrupt processor target register 82	GICD_ITARGETSR82	RW	H'00000000	H'E822 1948	32
Interrupt processor target register 83	GICD_ITARGETSR83	RW	H'00000000	H'E822 194C	32
Interrupt processor target register 84	GICD_ITARGETSR84	RW	H'00000000	H'E822 1950	32
Interrupt processor target register 85	GICD_ITARGETSR85	RW	H'00000000	H'E822 1954	32
Interrupt processor target register 86	GICD_ITARGETSR86	RW	H'00000000	H'E822 1958	32
Interrupt processor target register 87	GICD_ITARGETSR87	RW	H'00000000	H'E822 195C	32
Interrupt processor target register 88	GICD_ITARGETSR88	RW	H'00000000	H'E822 1960	32
Interrupt processor target register 89	GICD_ITARGETSR89	RW	H'00000000	H'E822 1964	32
Interrupt processor target register 90	GICD_ITARGETSR90	RW	H'00000000	H'E822 1968	32
Interrupt processor target register 91	GICD_ITARGETSR91	RW	H'00000000	H'E822 196C	32
Interrupt processor target register 92	GICD_ITARGETSR92	RW	H'00000000	H'E822 1970	32
Interrupt processor target register 93	GICD_ITARGETSR93	RW	H'00000000	H'E822 1974	32
Interrupt processor target register 94	GICD_ITARGETSR94	RW	H'00000000	H'E822 1978	32
Interrupt processor target register 95	GICD_ITARGETSR95	RW	H'00000000	H'E822 197C	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt processor target register 96	GICD_ITARGETSR96	RW	H'00000000	H'E822 1980	32
Interrupt processor target register 97	GICD_ITARGETSR97	RW	H'00000000	H'E822 1984	32
Interrupt processor target register 98	GICD_ITARGETSR98	RW	H'00000000	H'E822 1988	32
Interrupt processor target register 99	GICD_ITARGETSR99	RW	H'00000000	H'E822 198C	32
Interrupt processor target register 100	GICD_ITARGETSR100	RW	H'00000000	H'E822 1990	32
Interrupt processor target register 101	GICD_ITARGETSR101	RW	H'00000000	H'E822 1994	32
Interrupt processor target register 102	GICD_ITARGETSR102	RW	H'00000000	H'E822 1998	32
Interrupt processor target register 103	GICD_ITARGETSR103	RW	H'00000000	H'E822 199C	32
Interrupt processor target register 104	GICD_ITARGETSR104	RW	H'00000000	H'E822 19A0	32
Interrupt processor target register 105	GICD_ITARGETSR105	RW	H'00000000	H'E822 19A4	32
Interrupt processor target register 106	GICD_ITARGETSR106	RW	H'00000000	H'E822 19A8	32
Interrupt processor target register 107	GICD_ITARGETSR107	RW	H'00000000	H'E822 19AC	32
Interrupt processor target register 108	GICD_ITARGETSR108	RW	H'00000000	H'E822 19B0	32
Interrupt processor target register 109	GICD_ITARGETSR109	RW	H'00000000	H'E822 19B4	32
Interrupt processor target register 110	GICD_ITARGETSR110	RW	H'00000000	H'E822 19B8	32
Interrupt processor target register 111	GICD_ITARGETSR111	RW	H'00000000	H'E822 19BC	32
Interrupt processor target register 112	GICD_ITARGETSR112	RW	H'00000000	H'E822 19C0	32
Interrupt processor target register 113	GICD_ITARGETSR113	RW	H'00000000	H'E822 19C4	32
Interrupt processor target register 114	GICD_ITARGETSR114	RW	H'00000000	H'E822 19C8	32
Interrupt processor target register 115	GICD_ITARGETSR115	RW	H'00000000	H'E822 19CC	32
Interrupt processor target register 116	GICD_ITARGETSR116	RW	H'00000000	H'E822 19D0	32
Interrupt processor target register 117	GICD_ITARGETSR117	RW	H'00000000	H'E822 19D4	32
Interrupt processor target register 118	GICD_ITARGETSR118	RW	H'00000000	H'E822 19D8	32
Interrupt processor target register 119	GICD_ITARGETSR119	RW	H'00000000	H'E822 19DC	32
Interrupt processor target register 120	GICD_ITARGETSR120	RW	H'00000000	H'E822 19E0	32
Interrupt processor target register 121	GICD_ITARGETSR121	RW	H'00000000	H'E822 19E4	32
Interrupt processor target register 122	GICD_ITARGETSR122	RW	H'00000000	H'E822 19E8	32
Interrupt processor target register 123	GICD_ITARGETSR123	RW	H'00000000	H'E822 19EC	32
Interrupt processor target register 124	GICD_ITARGETSR124	RW	H'00000000	H'E822 19F0	32
Interrupt processor target register 125	GICD_ITARGETSR125	RW	H'00000000	H'E822 19F4	32
Interrupt processor target register 126	GICD_ITARGETSR126	RW	H'00000000	H'E822 19F8	32
Interrupt processor target register 127	GICD_ITARGETSR127	RW	H'00000000	H'E822 19FC	32
Interrupt configuration register 0	GICD_ICFGR0	RO	H'AAAAAAAA	H'E822 1C00	32
Interrupt configuration register 1	GICD_ICFGR1	RO	H'55540000	H'E822 1C04	32
Interrupt configuration register 2	GICD_ICFGR2	RW	H'55555555	H'E822 1C08	32
Interrupt configuration register 3	GICD_ICFGR3	RW	H'55555555	H'E822 1C0C	32
Interrupt configuration register 4	GICD_ICFGR4	RW	H'55555555	H'E822 1C10	32
Interrupt configuration register 5	GICD_ICFGR5	RW	H'55555555	H'E822 1C14	32
Interrupt configuration register 6	GICD_ICFGR6	RW	H'55555555	H'E822 1C18	32
Interrupt configuration register 7	GICD_ICFGR7	RW	H'55555555	H'E822 1C1C	32
Interrupt configuration register 8	GICD_ICFGR8	RW	H'55555555	H'E822 1C20	32
Interrupt configuration register 9	GICD_ICFGR9	RW	H'55555555	H'E822 1C24	32
Interrupt configuration register 10	GICD_ICFGR10	RW	H'55555555	H'E822 1C28	32
Interrupt configuration register 11	GICD_ICFGR11	RW	H'55555555	H'E822 1C2C	32
Interrupt configuration register 12	GICD_ICFGR12	RW	H'55555555	H'E822 1C30	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt configuration register 13	GICD_ICFGR13	RW	H'55555555	H'E822 1C34	32
Interrupt configuration register 14	GICD_ICFGR14	RW	H'55555555	H'E822 1C38	32
Interrupt configuration register 15	GICD_ICFGR15	RW	H'55555555	H'E822 1C3C	32
Interrupt configuration register 16	GICD_ICFGR16	RW	H'55555555	H'E822 1C40	32
Interrupt configuration register 17	GICD_ICFGR17	RW	H'55555555	H'E822 1C44	32
Interrupt configuration register 18	GICD_ICFGR18	RW	H'55555555	H'E822 1C48	32
Interrupt configuration register 19	GICD_ICFGR19	RW	H'55555555	H'E822 1C4C	32
Interrupt configuration register 20	GICD_ICFGR20	RW	H'55555555	H'E822 1C50	32
Interrupt configuration register 21	GICD_ICFGR21	RW	H'55555555	H'E822 1C54	32
Interrupt configuration register 22	GICD_ICFGR22	RW	H'55555555	H'E822 1C58	32
Interrupt configuration register 23	GICD_ICFGR23	RW	H'55555555	H'E822 1C5C	32
Interrupt configuration register 24	GICD_ICFGR24	RW	H'55555555	H'E822 1C60	32
Interrupt configuration register 25	GICD_ICFGR25	RW	H'55555555	H'E822 1C64	32
Interrupt configuration register 26	GICD_ICFGR26	RW	H'55555555	H'E822 1C68	32
Interrupt configuration register 27	GICD_ICFGR27	RW	H'55555555	H'E822 1C6C	32
Interrupt configuration register 28	GICD_ICFGR28	RW	H'55555555	H'E822 1C70	32
Interrupt configuration register 29	GICD_ICFGR29	RW	H'55555555	H'E822 1C74	32
Interrupt configuration register 30	GICD_ICFGR30	RW	H'55555555	H'E822 1C78	32
Interrupt configuration register 31	GICD_ICFGR31	RW	H'55555555	H'E822 1C7C	32
PPI status register	GICD_PPISR	RO	H'00000000	H'E822 1D00	32
SPI status register 0	GICD_SPISR0	RO	H'00000000	H'E822 1D04	32
SPI status register 1	GICD_SPISR1	RO	H'00000000	H'E822 1D08	32
SPI status register 2	GICD_SPISR2	RO	H'00000000	H'E822 1D0C	32
SPI status register 3	GICD_SPISR3	RO	H'00000000	H'E822 1D10	32
SPI status register 4	GICD_SPISR4	RO	H'00000000	H'E822 1D14	32
SPI status register 5	GICD_SPISR5	RO	H'00000000	H'E822 1D18	32
SPI status register 6	GICD_SPISR6	RO	H'00000000	H'E822 1D1C	32
SPI status register 7	GICD_SPISR7	RO	H'00000000	H'E822 1D20	32
SPI status register 8	GICD_SPISR8	RO	H'00000000	H'E822 1D24	32
SPI status register 9	GICD_SPISR9	RO	H'00000000	H'E822 1D28	32
SPI status register 10	GICD_SPISR10	RO	H'00000000	H'E822 1D2C	32
SPI status register 11	GICD_SPISR11	RO	H'00000000	H'E822 1D30	32
SPI status register 12	GICD_SPISR12	RO	H'00000000	H'E822 1D34	32
SPI status register 13	GICD_SPISR13	RO	H'00000000	H'E822 1D38	32
SPI status register 14	GICD_SPISR14	RO	H'00000000	H'E822 1D3C	32
Software generation interrupt register	GICD_SGIR	WO	H'00000000	H'E822 1F00	32
SGI clear-pending register 0	GICD_CPENDSGIR0	RW	H'00000000	H'E822 1F10	32
SGI clear-pending register 1	GICD_CPENDSGIR1	RW	H'00000000	H'E822 1F14	32
SGI clear-pending register 2	GICD_CPENDSGIR2	RW	H'00000000	H'E822 1F18	32
SGI clear-pending register 3	GICD_CPENDSGIR3	RW	H'00000000	H'E822 1F1C	32
SGI set-pending register 0	GICD_SPENDSGIR0	RW	H'00000000	H'E822 1F20	32
SGI set-pending register 1	GICD_SPENDSGIR1	RW	H'00000000	H'E822 1F24	32
SGI set-pending register 2	GICD_SPENDSGIR2	RW	H'00000000	H'E822 1F28	32
SGI set-pending register 3	GICD_SPENDSGIR3	RW	H'00000000	H'E822 1F2C	32
Peripheral ID4 register	GICD_PIDR4	RO	H'00000004	H'E822 1FD0	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Peripheral ID5 register	GICD_PIDR5	RO	H'00000000	H'E822 1FD4	32
Peripheral ID6 register	GICD_PIDR6	RO	H'00000000	H'E822 1FD8	32
Peripheral ID7 register	GICD_PIDR7	RO	H'00000000	H'E822 1FDC	32
Peripheral ID0 register	GICD_PIDR0	RO	H'00000090	H'E822 1FE0	32
Peripheral ID1 register	GICD_PIDR1	RO	H'000000B4	H'E822 1FE4	32
Peripheral ID2 register	GICD_PIDR2	RO	H'0000002B	H'E822 1FE8	32
Peripheral ID3 register	GICD_PIDR3	RO	H'00000000	H'E822 1FEC	32
Configuration ID0 register	GICD_CIDR0	RO	H'0000000D	H'E822 1FF0	32
Configuration ID1 register	GICD_CIDR1	RO	H'000000F0	H'E822 1FF4	32
Configuration ID2 register	GICD_CIDR2	RO	H'00000005	H'E822 1FF8	32
Configuration ID3 register	GICD_CIDR3	RO	H'000000B1	H'E822 1FFC	32
CPU interface control register	GICC_CTLR	RW	H'00000000	H'E822 2000	32
Interrupt priority mask register	GICC_PMR	RW	H'00000000	H'E822 2004	32
Binary point register	GICC_BPR	RW	H'00000002	H'E822 2008	32
Interrupt acknowledge register	GICC_IAR	RO	H'000003FF	H'E822 200C	32
End-of-interrupt register	GICC_EOIR	WO	H'00000000	H'E822 2010	32
Running priority register	GICC_RPR	RO	H'000000FF	H'E822 2014	32
Highest pending interrupt register	GICC_HPPIR	RO	H'000003FF	H'E822 2018	32
Aliased binary point register	GICC_ABPR	RW	H'00000003	H'E822 201C	32
Aliased interrupt acknowledge register	GICC_AIAR	RO	H'000003FF	H'E822 2020	32
Aliased end-of-interrupt register	GICC_AEOIR	WO	H'00000000	H'E822 2024	32
Aliased highest pending interrupt register	GICC_AHPPIR	RO	H'000003FF	H'E822 2028	32
Active priority register	GICC_APR0	RW	H'00000000	H'E822 20D0	32
Non-secure active priority register	GICC_NSAPR0	RW	H'00000000	H'E822 20E0	32
CPU interface implementer identification register	GICC_IIDR	RO	H'0202143B	H'E822 20FC	32
Interrupt disabled register	GICC_DIR	WO	H'00000000	H'E822 3000 *4	32

Note 1. When the NMI pin is high, becomes H'8000; when low, becomes H'0000.
Note 2. Only 0 can be written after reading 1, to clear the flag.
Note 3. Use the following expression to calculate the number of interrupts from the number of IT lines.
(15+1)×32: 512
Note 4. This interrupt controller does not support Virtual interface control blocks and Virtual CPU interfaces.
Address setting H'E822 3000 or more is invalid.

7.3.1 Interrupt Control Register 0 (ICR0)

ICR0 is a 16-bit register that sets the input signal detection mode for the external interrupt input pin NMI, and indicates the input level at the NMI pin.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NMIL	—	—	—	—	—	—	NMIE	—	—	—	—	—	—	NMIF	—
Initial Value:	*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R(W)*2	R

Note 1. 1 when the NMI pin is high, and 0 when the NMI pin is low.
Note 2. Only 0 can be written after reading 1, to clear the flag.

Bit	Bit Name	Initial Value	R/W	Description
15	NMIL	*1	R	NMI Input Level Sets the level of the signal input at the NMI pin. The NMI pin level can be obtained by reading this bit. This bit cannot be modified. 0: Low level is input to NMI pin 1: High level is input to NMI pin
14 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	NMIE	0	R/W	NMI Edge Select Selects whether the falling or rising edge of the interrupt request signal on the NMI pin is detected. 0: Interrupt request is detected on falling edge of NMI input 1: Interrupt request is detected on rising edge of NMI input
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	NMIF	0	R/(W)*2	NMI Interrupt Request This bit indicates the status of the NMI interrupt request. This bit cannot be modified. 0: NMI interrupt request has not occurred [Clearing conditions] • Cleared by changing NMIE of ICR0 • Cleared by reading NMIF while NMIF = 1, then writing 0 to NMIF 1: NMI interrupt request is detected [Setting condition] • Edge corresponding to NMIE of ICR0 has occurred at NMI pin
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

7.3.2 Interrupt Control Register 1 (ICR1)

ICR1 is a 16-bit register that specifies the detection mode for external interrupt input pins IRQ7 to IRQ0 individually: low level, falling edge, rising edge, or both edges.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ 71S	IRQ 70S	IRQ 61S	IRQ 60S	IRQ 51S	IRQ 50S	IRQ 41S	IRQ 40S	IRQ 31S	IRQ 30S	IRQ 21S	IRQ 20S	IRQ 11S	IRQ 10S	IRQ 01S	IRQ 00S
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	IRQ71S	0	R/W	IRQ Sense Select These bits select whether interrupt signals corresponding to pins IRQ7 to IRQ0 are detected by a low level, falling edge, rising edge, or both edges. 00: Interrupt request is detected on low level of IRQn input 01: Interrupt request is detected on falling edge of IRQn input 10: Interrupt request is detected on rising edge of IRQn input 11: Interrupt request is detected on both edges of IRQn input
14	IRQ70S	0	R/W	
13	IRQ61S	0	R/W	
12	IRQ60S	0	R/W	
11	IRQ51S	0	R/W	
10	IRQ50S	0	R/W	
9	IRQ41S	0	R/W	
8	IRQ40S	0	R/W	
7	IRQ31S	0	R/W	
6	IRQ30S	0	R/W	
5	IRQ21S	0	R/W	
4	IRQ20S	0	R/W	
3	IRQ11S	0	R/W	
2	IRQ10S	0	R/W	
1	IRQ01S	0	R/W	
0	IRQ00S	0	R/W	

[Legend]
n = 7 to 0

7.3.3 IRQ Interrupt Request Register (IRQRR)

IRQRR is a 16-bit register that indicates interrupt requests from external input pins IRQ7 to IRQ0. If edge detection is set for the IRQ7 to IRQ0 interrupts, writing 0 to the IRQ7F to IRQ0F bits after reading IRQ7F to IRQ0F = 1 cancels the retained interrupts.

IRQMSK is a bit to set release, when the IRQ signal is used as a software standby cancel source signal. For details, refer to section 52.3.2, Software Standby Mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ MSK	—	—	—	—	—	—	—	IRQ 7F	IRQ 6F	IRQ 5F	IRQ 4F	IRQ 3F	IRQ 2F	IRQ 1F	IRQ 0F
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R(W)*	R(W)*	R(W)*	R(W)*	R(W)*	R(W)*	R(W)*	R(W)*

Note: * Only 0 can be written to clear the flag after 1 is read.

Bit	Bit Name	Initial Value	R/W	Description
15	IRQMSK	0	R/W	IRQ Software standby cancel Source Mask bit *1 0: Masked 1: Unmasked
14 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	IRQ7F	0	R/(W)*	IRQ Interrupt Request These bits indicate the status of the IRQ7 to IRQ0 interrupt requests. Level detection: 0: IRQn interrupt request has not occurred [Clearing condition] • IRQn input is high 1: IRQn interrupt has occurred [Setting condition] • IRQn input is low Edge detection: 0: IRQn interrupt request is not detected [Clearing condition] • Cleared by reading IRQnF while IRQnF = 1, then writing 0 to IRQnF 1: IRQn interrupt request is detected [Setting condition] • Edge corresponding to IRQn1S or IRQn0S of ICR1 has occurred at IRQn pin
6	IRQ6F	0	R/(W)*	
5	IRQ5F	0	R/(W)*	
4	IRQ4F	0	R/(W)*	
3	IRQ3F	0	R/(W)*	
2	IRQ2F	0	R/(W)*	
1	IRQ1F	0	R/(W)*	
0	IRQ0F	0	R/(W)*	

[Legend]

n = 7 to 0

Note 1. When IRQMSK is 1 (Unmasked), the IRQ signal can be used as a software standby cancel source.

7.4 Interrupt Sources

There are four types of interrupt sources: NMI, IRQ, on-chip peripheral modules, and pin interrupts. Each interrupt has a priority level (0 to 31), with 0 the highest and 31 the lowest.

7.4.1 NMI Interrupt

The NMI interrupt with the highest priority is accepted by the CPU as an FIQ exception all times. NMI interrupt requests are edge-detected, and the NMI edge select bit (NMIE) in ICR0 selects whether the rising edge or falling edge is detected. The status of the interrupt request can be checked by reading the NMI interrupt request bit (NMIF) in the ICR0. When the NMIE bit is changed, the NMI interrupt request that is retained is cleared.
When deep standby mode is entered, deep standby mode is canceled by the NMI interrupt.

7.4.2 IRQ Interrupts

IRQ interrupts are input from pins IRQ7 to IRQ0. For the IRQ interrupts, low-level, falling-edge, rising-edge, or both-edge detection can be selected individually for each pin by the IRQ sense select bits (IRQ71S to IRQ01S and IRQ70S to IRQ00S) in interrupt control register 1 (ICR1).
When using low-level sensing for IRQ interrupts, an interrupt request signal is sent to the interrupt controller while the IRQ7 to IRQ0 pins are low. An interrupt request signal is stopped being sent to the interrupt controller when the IRQ7 to IRQ0 pins are driven high. The status of the interrupt requests can be checked by reading the IRQ interrupt request bits (IRQ7F to IRQ0F) in the IRQ interrupt request register (IRQRR).
When using edge-sensing for IRQ interrupts, an interrupt request is detected due to change of the IRQ7 to IRQ0 pin states, and an interrupt request signal is sent to the interrupt controller. The result of IRQ interrupt request detection is retained until that interrupt request is accepted. Whether IRQ interrupt requests have been detected or not can be checked by reading the IRQ7F to IRQ0F bits in the IRQ interrupt request register (IRQRR). Writing 0 to these bits after reading them as 1 clears the result of IRQ interrupt request detection.
When returning from IRQ interrupt exception service routine, execute the return instruction after confirming that the interrupt request has been cleared by the IRQ interrupt request register (IRQRR) so as not to accidentally receive the interrupt request again.

7.4.3 On-Chip Peripheral Module Interrupts

When returning from the interrupt exception service routine for an interrupt request at the peripheral-module level, execute the return instruction after clearing the source flag at the source of the request and reading the source flag so that the interrupt request is not accidentally received again.

7.4.4 Pin Interrupts

Pin interrupts are input from the general-purpose I/O port pins. Signals input on the general-purpose I/O port pins are conveyed as interrupt signals regardless of mode settings for the general-purpose I/O port pin.
Accordingly, if pin interrupts are to be used, set the pin as an input port. In cases where the general-purpose I/O port pins are used for the peripheral functions, pin interrupts can only be set when the pins are specified as input or input and output. For the pin interrupts, the general-purpose I/O port pins are divided into 32-group of TINT31 to TINT0. The pin interrupts select mask in each group can be set with the pin function control register. For the settings of general purpose I/O port pins and function control registers, see section 51, GPIO.
For the pin interrupts, high-level or rising-edge detection can be selected individually for each pin group by the interrupt configuration registers (GICD_ICFGRn).
For a description of the interrupt configuration registers (GICD_ICFGRn), see the GIC architecture specification.

7.5 Interrupt IDs

Table 7.3 lists the interrupt sources and their interrupt IDs, and the registers for setting the interrupt sources. Do not make settings other than those in Table 7.3, otherwise, the operation cannot be guaranteed.
Each interrupt source is allocated a different interrupt ID. To control notification of the interrupt source to the CPU and reference its status, it is necessary to set and reference the following registers which correspond to given interrupt IDs.

- Interrupt group registerGICD_IGROUPRn
- Interrupt set-enable registerGICD_ISENABLERn
- Interrupt clear-enable registerGICD_ICENABLERn
- Interrupt set-pending registerGICD_ISPENDRn
- Interrupt clear-pending register GICD_ICPENDRn
- Active bit set register GICD_ISACTIVERn
- Interrupt configuration register GICD_ICFGRn
- Interrupt priority register GICD_IPRIORITYRn
- Interrupt processor target register GICD_ITARGETSRn

For the procedure for the initial settings of the registers, see section 7.6.1, Interrupt Settings. For details on individual registers, see the Arm Generic Interrupt Controller Architecture Specification and the PrimeCell® Generic Interrupt Controller (GIC-400) Technical Reference Manual from Arm.

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation						
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
GIC software interrupt	—	—	—	0	0	0	0	1 to 0	0	7 to 0
				1		1		3 to 2		15 to 8
				2		2		5 to 4		23 to 16
				3		3		7 to 6		31 to 24
				4		4		9 to 8	1	7 to 0
				5		5		11 to 10		15 to 8
				6		6		13 to 12		23 to 16
				7		7		15 to 14		31 to 24
				8		8		17 to 16	2	7 to 0
				9		9		19 to 18		15 to 8
				10		10		21 to 20		23 to 16
				11		11		23 to 22		31 to 24
				12		12		25 to 24	3	7 to 0
				13		13		27 to 26		15 to 8
				14		14		29 to 28		23 to 16
				15		15		31 to 30		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation					
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
CA9 main	0	PMUIRQ0	Level	32	1	0	2	1 to 0	8	7 to 0
		COMMRX	Level	33		1		3 to 2		15 to 8
		COMMTX	Level	34		2		5 to 4		23 to 16
		CTIIRQ	Level	35		3		7 to 6		31 to 24
IRQ	—	IRQ0	Level	36		4		9 to 8	9	7 to 0
		IRQ1	Level	37		5		11 to 10		15 to 8
		IRQ2	Level	38		6		13 to 12		23 to 16
		IRQ3	Level	39		7		15 to 14		31 to 24
		IRQ4	Level	40		8		17 to 16	10	7 to 0
		IRQ5	Level	41		9		19 to 18		15 to 8
		IRQ6	Level	42		10		21 to 20		23 to 16
		IRQ7	Level	43		11		23 to 22		31 to 24
Secondary Cache	—	PL310ERR	Level	44		12		25 to 24	11	7 to 0
Direct memory access controller (Secure)	0	DMAINT0	Edge	45		13		27 to 26		15 to 8
	1	DMAINT1	Edge	46		14		29 to 28		23 to 16
	2	DMAINT2	Edge	47		15		31 to 30		31 to 24
	3	DMAINT3	Edge	48		16	3	1 to 0	12	7 to 0
	4	DMAINT4	Edge	49		17		3 to 2		15 to 8
	5	DMAINT5	Edge	50		18		5 to 4		23 to 16
	6	DMAINT6	Edge	51		19		7 to 6		31 to 24
	7	DMAINT7	Edge	52		20		9 to 8	13	7 to 0
	8	DMAINT8	Edge	53		21		11 to 10		15 to 8
	9	DMAINT9	Edge	54		22		13 to 12		23 to 16
	10	DMAINT10	Edge	55		23		15 to 14		31 to 24
	11	DMAINT11	Edge	56		24		17 to 16	14	7 to 0
	12	DMAINT12	Edge	57		25		19 to 18		15 to 8
	13	DMAINT13	Edge	58		26		21 to 20		23 to 16
	14	DMAINT14	Edge	59		27		23 to 22		31 to 24
	15	DMAINT15	Edge	60		28		25 to 24	15	7 to 0
Direct memory access controller (Non-secure)	0	DMAERR0	Edge	61		29		27 to 26		15 to 8
	1	DMAERR1	Edge	62		30		29 to 28		23 to 16
USB host / function	0	USBHI0	Level	63		31		31 to 30		31 to 24
		USBFI0	Level	64	2	0	4	1 to 0	16	7 to 0
		USBFDMA00	Level	65		1		3 to 2		15 to 8
		USBFDMA01	Level	66		2		5 to 4		23 to 16
		USBFDMAERR0	Level	67		3		7 to 6		31 to 24
	1	USBHI1	Level	68		4		9 to 8	17	7 to 0
		USBFI1	Level	69		5		11 to 10		15 to 8
		USBFDMA10	Level	70		6		13 to 12		23 to 16
		USBFDMA11	Level	71		7		15 to 14		31 to 24
		USBFDMAERR1	Level	72		8		17 to 16	18	7 to 0
Video Display Controller 6		S0_VI_VSYNC0	Level	73		9		19 to 18		15 to 8
		S0_LO_VSYNC0	Level	74		10		21 to 20		23 to 16
		S0_VSYNCERR0	Level	75		11		23 to 22		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation					
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
Video Display Controller 6		GR3_VLINE0	Level	76	2	12	4	25 to 24	19	7 to 0
		S0_VFIELD0	Level	77		13		27 to 26		15 to 8
		IV1_VBUFERR0	Level	78		14		29 to 28		23 to 16
		IV3_VBUFERR0	Level	79		15		31 to 30		31 to 24
		IV5_VBUFERR0	Level	80		16	5	1 to 0	20	7 to 0
		IV6_VBUFERR0	Level	81		17		3 to 2		15 to 8
		S0_WLINE0	Level	82		18		5 to 4		23 to 16
Image Renderer 2		IMR2I0	Level	83		19		7 to 6		31 to 24
JPEG codec unit		JEDI	Level	84		20		9 to 8	21	7 to 0
		JDTI	Level	85		21		11 to 10		15 to 8
2D Drawing Engine		DRWI	Level	86		22		13 to 12		23 to 16
MIPI CSI-2 Interface		CSII	Level	87		23		15 to 14		31 to 24
OS Timer	0	OSTMI0	Edge	88		24		17 to 16	22	7 to 0
	1	OSTMI1	Edge	89		25		19 to 18		15 to 8
	2	OSTMI2	Edge	90		26		21 to 20		23 to 16
Bus State Controller		CMI	Level	91		27		23 to 22		31 to 24
		WTOUT	Level	92		28		25 to 24	23	7 to 0
Watchdog Timer		ITI	Level	93		29		27 to 26		15 to 8
		CA9PEI	Level	94		30		29 to 28		23 to 16
Multi-Function Timer Pulse Unit 3	0	TGIA0	Edge	95		31		31 to 30		31 to 24
		TGIB0	Edge	96	3	0	6	1 to 0	24	7 to 0
		TGIC0	Edge	97		1		3 to 2		15 to 8
		TGID0	Edge	98		2		5 to 4		23 to 16
		TCIV0	Edge	99		3		7 to 6		31 to 24
		TGIE0	Edge	100		4		9 to 8	25	7 to 0
		TGIF0	Edge	101		5		11 to 10		15 to 8
	1	TGIA1	Edge	102		6		13 to 12		23 to 16
		TGIB1	Edge	103		7		15 to 14		31 to 24
		TCIV1	Edge	104		8		17 to 16	26	7 to 0
		TCIU1	Edge	105		9		19 to 18		15 to 8
	2	TGIA2	Edge	106		10		21 to 20		23 to 16
		TGIB2	Edge	107		11		23 to 22		31 to 24
		TCIV2	Edge	108		12		25 to 24	27	7 to 0
		TCIU2	Edge	109		13		27 to 26		15 to 8
	3	TGIA3	Edge	110		14		29 to 28		23 to 16
		TGIB3	Edge	111		15		31 to 30		31 to 24
		TGIC3	Edge	112		16	7	1 to 0	28	7 to 0
		TGID3	Edge	113		17		3 to 2		15 to 8
		TCIV3	Edge	114		18		5 to 4		23 to 16
	4	TGIA4	Edge	115		19		7 to 6		31 to 24
		TGIB4	Edge	116		20		9 to 8	29	7 to 0
		TGIC4	Edge	117		21		11 to 10		15 to 8
		TGID4	Edge	118		22		13 to 12		23 to 16
		TCIV4	Edge	119		23		15 to 14		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation					
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
Multi-Function Timer Pulse Unit 3	5	TGIU5	Edge	120	3	24	7	17 to 16	30	7 to 0
		TGIV5	Edge	121		25		19 to 18		15 to 8
		TGIW5	Edge	122		26		21 to 20		23 to 16
	6	TGIA6	Edge	123		27		23 to 22		31 to 24
		TGIB6	Edge	124		28		25 to 24	31	7 to 0
		TGIC6	Edge	125		29		27 to 26		15 to 8
		TGID6	Edge	126		30		29 to 28		23 to 16
		TCIV6	Edge	127		31		31 to 30		31 to 24
	7	TGIA7	Edge	128	4	0	8	1 to 0	32	7 to 0
		TGIB7	Edge	129		1		3 to 2		15 to 8
		TGIC7	Edge	130		2		5 to 4		23 to 16
		TGID7	Edge	131		3		7 to 6		31 to 24
		TCIV7	Edge	132		4		9 to 8	33	7 to 0
	8	TGIA8	Edge	133		5		11 to 10		15 to 8
		TGIB8	Edge	134		6		13 to 12		23 to 16
		TGIC8	Edge	135		7		15 to 14		31 to 24
		TGID8	Edge	136		8		17 to 16	34	7 to 0
		TCIV8	Edge	137		9		19 to 18		15 to 8
Reserved				138		10		21 to 20		23 to 16
General PWM Timer	0	CCMPA0	Edge	139		11		23 to 22		31 to 24
		CCMPB0	Edge	140		12		25 to 24	35	7 to 0
		CMPC0	Edge	141		13		27 to 26		15 to 8
		CMPD0	Edge	142		14		29 to 28		23 to 16
Reserved				143		15		31 to 30		31 to 24
				144		16	9	1 to 0	36	7 to 0
				145		17		3 to 2		15 to 8
General PWM Timer	0	CMPE0	Edge	146		18		5 to 4		23 to 16
		CMPF0	Edge	147		19		7 to 6		31 to 24
		ADTRGA0	Edge	148		20		9 to 8	37	7 to 0
		ADTRGB0	Edge	149		21		11 to 10		15 to 8
		OVF0	Edge	150		22		13 to 12		23 to 16
		UNF0	Edge	151		23		15 to 14		31 to 24
	1	CCMPA1	Edge	152		24		17 to 16	38	7 to 0
		CCMPB1	Edge	153		25		19 to 18		15 to 8
		CMPC1	Edge	154		26		21 to 20		23 to 16
		CMPD1	Edge	155		27		23 to 22		31 to 24
Reserved				156		28		25 to 24	39	7 to 0
				157		29		27 to 26		15 to 8
				158		30		29 to 28		23 to 16
General PWM Timer	1	CMPE1	Edge	159		31		31 to 30		31 to 24
		CMPF1	Edge	160	5	0	10	1 to 0	40	7 to 0
		ADTRGA1	Edge	161		1		3 to 2		15 to 8
		ADTRGB1	Edge	162		2		5 to 4		23 to 16
		OVF1	Edge	163		3		7 to 6		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation					
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
General PWM Timer	1	UNF1	Edge	164	5	4	10	9 to 8	41	7 to 0
	2	CCMPA2	Edge	165		5		11 to 10		15 to 8
		CCMPB2	Edge	166		6		13 to 12		23 to 16
		CMPC2	Edge	167		7		15 to 14		31 to 24
		CMPD2	Edge	168		8		17 to 16	42	7 to 0
Reserved				169		9		19 to 18		15 to 8
				170		10		21 to 20		23 to 16
				171		11		23 to 22		31 to 24
General PWM Timer	2	CMPE2	Edge	172		12		25 to 24	43	7 to 0
		CMPF2	Edge	173		13		27 to 26		15 to 8
		ADTRGA2	Edge	174		14		29 to 28		23 to 16
		ADTRGB2	Edge	175		15		31 to 30		31 to 24
		OVF2	Edge	176		16	11	1 to 0	44	7 to 0
		UNF2	Edge	177		17		3 to 2		15 to 8
	3	CCMPA3	Edge	178		18		5 to 4		23 to 16
		CCMPB3	Edge	179		19		7 to 6		31 to 24
		CMPC3	Edge	180		20		9 to 8	45	7 to 0
		CMPD3	Edge	181		21		11 to 10		15 to 8
Reserved				182		22		13 to 12		23 to 16
				183		23		15 to 14		31 to 24
				184		24		17 to 16	46	7 to 0
General PWM Timer	3	CMPE3	Edge	185		25		19 to 18		15 to 8
		CMPF3	Edge	186		26		21 to 20		23 to 16
		ADTRGA3	Edge	187		27		23 to 22		31 to 24
		ADTRGB3	Edge	188		28		25 to 24	47	7 to 0
		OVF3	Edge	189		29		27 to 26		15 to 8
		UNF3	Edge	190		30		29 to 28		23 to 16
	4	CCMPA4	Edge	191		31		31 to 30		31 to 24
		CCMPB4	Edge	192	6	0	12	1 to 0	48	7 to 0
		CMPC4	Edge	193		1		3 to 2		15 to 8
		CMPD4	Edge	194		2		5 to 4		23 to 16
Reserved				195		3		7 to 6		31 to 24
				196		4		9 to 8	49	7 to 0
				197		5		11 to 10		15 to 8
General PWM Timer	4	CMPE4	Edge	198		6		13 to 12		23 to 16
		CMPF4	Edge	199		7		15 to 14		31 to 24
		ADTRGA4	Edge	200		8		17 to 16	50	7 to 0
		ADTRGB4	Edge	201		9		19 to 18		15 to 8
		OVF4	Edge	202		10		21 to 20		23 to 16
		UNF4	Edge	203		11		23 to 22		31 to 24
	5	CCMPA5	Edge	204		12		25 to 24	51	7 to 0
		CCMPB5	Edge	205		13		27 to 26		15 to 8
		CMPC5	Edge	206		14		29 to 28		23 to 16
		CMPD5	Edge	207		15		31 to 30		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation					
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
Reserved				208	6	16	13	1 to 0	52	7 to 0
				209		17		3 to 2		15 to 8
				210		18		5 to 4		23 to 16
General PWM Timer	5	CMPE5	Edge	211		19		7 to 6		31 to 24
		CMPF5	Edge	212		20		9 to 8	53	7 to 0
		ADTRGA5	Edge	213		21		11 to 10		15 to 8
		ADTRGB5	Edge	214		22		13 to 12		23 to 16
		OVF5	Edge	215		23		15 to 14		31 to 24
		UNF5	Edge	216		24		17 to 16	54	7 to 0
	6	CCMPA6	Edge	217		25		19 to 18		15 to 8
		CCMPB6	Edge	218		26		21 to 20		23 to 16
		CMPC6	Edge	219		27		23 to 22		31 to 24
		CMPD6	Edge	220		28		25 to 24	55	7 to 0
Reserved				221		29		27 to 26		15 to 8
				222		30		29 to 28		23 to 16
				223		31		31 to 30		31 to 24
General PWM Timer	6	CMPE6	Edge	224	7	0	14	1 to 0	56	7 to 0
		CMPF6	Edge	225		1		3 to 2		15 to 8
		ADTRGA6	Edge	226		2		5 to 4		23 to 16
		ADTRGB6	Edge	227		3		7 to 6		31 to 24
		OVF6	Edge	228		4		9 to 8	57	7 to 0
		UNF6	Edge	229		5		11 to 10		15 to 8
	7	CCMPA7	Edge	230		6		13 to 12		23 to 16
		CCMPB7	Edge	231		7		15 to 14		31 to 24
		CMPC7	Edge	232		8		17 to 16	58	7 to 0
		CMPD7	Edge	233		9		19 to 18		15 to 8
Reserved				234		10		21 to 20		23 to 16
				235		11		23 to 22		31 to 24
				236		12		25 to 24	59	7 to 0
General PWM Timer	7	CMPE7	Edge	237		13		27 to 26		15 to 8
		CMPF7	Edge	238		14		29 to 28		23 to 16
		ADTRGA7	Edge	239		15		31 to 30		31 to 24
		ADTRGB7	Edge	240		16	15	1 to 0	60	7 to 0
		OVF7	Edge	241		17		3 to 2		15 to 8
		UNF7	Edge	242		18		5 to 4		23 to 16
Port Output Enable 3	1	OEI1	Level	243		19		7 to 6		31 to 24
	2	OEI2	Level	244		20		9 to 8	61	7 to 0
	3	OEI3	Level	245		21		11 to 10		15 to 8
	4	OEI4	Level	246		22		13 to 12		23 to 16
A/D converter		S12ADI0	Edge	247		23		15 to 14		31 to 24
		S12GBADI0	Edge	248		24		17 to 16	62	7 to 0
		S12GCADI0	Edge	249		25		19 to 18		15 to 8
		S12ADCMPAI0	Level	250		26		21 to 20		23 to 16
		S12ADCMPBI0	Level	251		27		23 to 22		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation					
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
Serial Sound Interface	0	INT_ssif_int_req_0	Level	252	7	28	15	25 to 24	63	7 to 0
		INT_ssif_dma_tx_0	Edge	253		29		27 to 26		15 to 8
		INT_ssif_dma_rx_0	Edge	254		30		29 to 28		23 to 16
	1	INT_ssif_int_req_1	Level	255		31		31 to 30		31 to 24
		INT_ssif_dma_tx_1	Edge	256	8	0	16	1 to 0	64	7 to 0
		INT_ssif_dma_rx_1	Edge	257		1		3 to 2		15 to 8
	2	INT_ssif_int_req_2	Level	258		2		5 to 4		23 to 16
		INT_ssif_dma_rt_2	Edge	259		3		7 to 6		31 to 24
	3	INT_ssif_int_req_3	Level	260		4		9 to 8	65	7 to 0
		INT_ssif_dma_tx_3	Edge	261		5		11 to 10		15 to 8
		INT_ssif_dma_rx_3	Edge	262		6		13 to 12		23 to 16
Renesas SPDIF Interface		SPDIFI	Level	263		7		15 to 14		31 to 24
I²C Bus Interface	0	INTRIICTEI0	Level	264		8		17 to 16	66	7 to 0
		INTRIICRI0	Edge	265		9		19 to 18		15 to 8
		INTRIICTI0	Edge	266		10		21 to 20		23 to 16
		INTRIICSPI0	Level	267		11		23 to 22		31 to 24
		INTRIICSTI0	Level	268		12		25 to 24	67	7 to 0
		INTRIICNAKI0	Level	269		13		27 to 26		15 to 8
		INTRIICALI0	Level	270		14		29 to 28		23 to 16
		INTRIICTMOI0	Level	271		15		31 to 30		31 to 24
	1	INTRIICTEI1	Level	272		16	17	1 to 0	68	7 to 0
		INTRIICRI1	Edge	273		17		3 to 2		15 to 8
		INTRIICTI1	Edge	274		18		5 to 4		23 to 16
		INTRIICSPI1	Level	275		19		7 to 6		31 to 24
		INTRIICSTI1	Level	276		20		9 to 8	69	7 to 0
		INTRIICNAKI1	Level	277		21		11 to 10		15 to 8
		INTRIICALI1	Level	278		22		13 to 12		23 to 16
		INTRIICTMOI1	Level	279		23		15 to 14		31 to 24
	2	INTRIICTEI2	Level	280		24		17 to 16	70	7 to 0
		INTRIICRI2	Edge	281		25		19 to 18		15 to 8
		INTRIICTI2	Edge	282		26		21 to 20		23 to 16
		INTRIICSPI2	Level	283		27		23 to 22		31 to 24
		INTRIICSTI2	Level	284		28		25 to 24	71	7 to 0
		INTRIICNAKI2	Level	285		29		27 to 26		15 to 8
		INTRIICALI2	Level	286		30		29 to 28		23 to 16
		INTRIICTMOI2	Level	287		31		31 to 30		31 to 24
	3	INTRIICTEI3	Level	288	9	0	18	1 to 0	72	7 to 0
		INTRIICRI3	Edge	289		1		3 to 2		15 to 8
		INTRIICTI3	Edge	290		2		5 to 4		23 to 16
		INTRIICSPI3	Level	291		3		7 to 6		31 to 24
		INTRIICSTI3	Level	292		4		9 to 8	73	7 to 0
		INTRIICNAKI3	Level	293		5		11 to 10		15 to 8
		INTRIICALI3	Level	294		6		13 to 12		23 to 16
		INTRIICTMOI3	Level	295		7		15 to 14		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation					
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
Reserved				296	9	8	18	17 to 16	74	7 to 0
Serial Communication Interface with FIFO	0	ERI0/BRI0	Level	297		9		19 to 18		15 to 8
		RXI0	Level	298		10		21 to 20		23 to 16
		TXI0	Level	299		11		23 to 22		31 to 24
		TEI0/DRI0	Level	300		12		25 to 24	75	7 to 0
Reserved				301		13		27 to 26		15 to 8
				302		14		29 to 28		23 to 16
Serial Communication Interface with FIFO	1	ERI1/BRI1	Level	303		15		31 to 30		31 to 24
		RXI1	Level	304		16	19	1 to 0	76	7 to 0
		TXI1	Level	305		17		3 to 2		15 to 8
		TEI1/DRI1	Level	306		18		5 to 4		23 to 16
Reserved				307		19		7 to 6		31 to 24
				308		20		9 to 8	77	7 to 0
Serial Communication Interface with FIFO	2	ERI2/BRI2	Level	309		21		11 to 10		15 to 8
		RXI2	Level	310		22		13 to 12		23 to 16
		TXI2	Level	311		23		15 to 14		31 to 24
		TEI2/DRI2	Level	312		24		17 to 16	78	7 to 0
Reserved				313		25		19 to 18		15 to 8
				314		26		21 to 20		23 to 16
Serial Communication Interface with FIFO	3	ERI3/BRI3	Level	315		27		23 to 22		31 to 24
		RXI3	Level	316		28		25 to 24	79	7 to 0
		TXI3	Level	317		29		27 to 26		15 to 8
		TEI3/DRI3	Level	318		30		29 to 28		23 to 16
Reserved				319		31		31 to 30		31 to 24
				320	10	0	20	1 to 0	80	7 to 0
Serial Communication Interface with FIFO	4	ERI4/BRI4	Level	321		1		3 to 2		15 to 8
		RXI4	Level	322		2		5 to 4		23 to 16
		TXI4	Level	323		3		7 to 6		31 to 24
		TEI4/DRI4	Level	324		4		9 to 8	81	7 to 0
Reserved				325		5		11 to 10		15 to 8
CANFD Interface		GERI	Level	326		6		13 to 12		23 to 16
		RFI	Level	327		7		15 to 14		31 to 24
		CFRXI0	Level	328		8		17 to 16	82	7 to 0
		CERI0	Level	329		9		19 to 18		15 to 8
		CTXI0	Level	330		10		21 to 20		23 to 16
		CFRXI1	Level	331		11		23 to 22		31 to 24
		CERI1	Level	332		12		25 to 24	83	7 to 0
		CTXI1	Level	333		13		27 to 26		15 to 8
Direct memory access controller (Non-secure)	0	DMAINT0	Edge	334		14		29 to 28		23 to 16
	1	DMAINT1	Edge	335		15		31 to 30		31 to 24
	2	DMAINT2	Edge	336		16	21	1 to 0	84	7 to 0
	3	DMAINT3	Edge	337		17		3 to 2		15 to 8
	4	DMAINT4	Edge	338		18		5 to 4		23 to 16
	5	DMAINT5	Edge	339		19		7 to 6		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation						
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
Direct memory access controller (Non-secure)	6	DMAINT6	Edge	340	10	20	21	9 to 8	85	7 to 0
	7	DMAINT7	Edge	341		21		11 to 10		15 to 8
	8	DMAINT8	Edge	342		22		13 to 12		23 to 16
	9	DMAINT9	Edge	343		23		15 to 14		31 to 24
Renesas Serial Peripheral Interface	0	SPEI0	Level	344		24		17 to 16	86	7 to 0
		SPRI0	Level	345		25		19 to 18		15 to 8
		SPTI0	Level	346		26		21 to 20		23 to 16
	1	SPEI1	Level	347		27		23 to 22		31 to 24
		SPRI1	Level	348		28		25 to 24	87	7 to 0
		SPTI1	Level	349		29		27 to 26		15 to 8
	2	SPEI2	Level	350		30		29 to 28		23 to 16
		SPRI2	Level	351		31		31 to 30		31 to 24
		SPTI2	Level	352	11	0	22	1 to 0	88	7 to 0
NAND flash controller		NAND	Level	353		1		3 to 2		15 to 8
SD/MMC host interface	0	SDHI0_0	Level	354		2		5 to 4		23 to 16
Reserved				355		3		7 to 6		31 to 24
SD/MMC host interface	1	SDHI1_0	Level	356		4		9 to 8	89	7 to 0
Reserved				357		5		11 to 10		15 to 8
HyperBus™ controller		HYPER	Level	358		6		13 to 12		23 to 16
Realtime Clock	0	ALM	Level	359		7		15 to 14		31 to 24
		PRD	Level	360		8		17 to 16	90	7 to 0
		CUP	Level	361		9		19 to 18		15 to 8
	1	ALM_S	Level	362		10		21 to 20		23 to 16
		PRD_S	Level	363		11		23 to 22		31 to 24
		CUP_S	Level	364		12		25 to 24	91	7 to 0
Serial Communication Interface	0	ERI0	Level	365		13		27 to 26		15 to 8
		RXI0	Edge	366		14		29 to 28		23 to 16
		TXI0	Edge	367		15		31 to 30		31 to 24
		TEI0	Level	368		16	23	1 to 0	92	7 to 0
	1	ERI1	Level	369		17		3 to 2		15 to 8
		RXI1	Edge	370		18		5 to 4		23 to 16
		TXI1	Edge	371		19		7 to 6		31 to 24
		TEI1	Level	372		20		9 to 8	93	7 to 0
Ethernet Controller	—	EINT0	Level	373		21		11 to 10		15 to 8
		EINT1	Level	374		22		13 to 12		23 to 16
		PINT	Level	375		23		15 to 14		31 to 24
		MINT	Level	376		24		17 to 16	94	7 to 0
		IPLS	Edge	377		25		19 to 18		15 to 8
Capture Engine Unit		CEUI	Level	378		26		21 to 20		23 to 16
Internal bus		H2XUSB00_ERRINT	Edge	379		27		23 to 22		31 to 24
		H2XUSB01_ERRINT	Edge	380		28		25 to 24	95	7 to 0
		H2XUSB10_ERRINT	Edge	381		29		27 to 26		15 to 8
		H2XUSB11_ERRINT	Edge	382		30		29 to 28		23 to 16
		H2XETH_ERRINT	Edge	383		31		31 to 30		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation					
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
Internal bus		X2HPERI12_ERRINT	Edge	384	12	0	24	1 to 0	96	7 to 0
		X2HPERI34_ERRINT	Edge	385		1		3 to 2		15 to 8
		X2HPERI5_ERRINT	Edge	386		2		5 to 4		23 to 16
		X2HPERI67_ERRINT	Edge	387		3		7 to 6		31 to 24
		H2XDBG_ERRINT	Edge	388		4		9 to 8	97	7 to 0
		X2HDBGR_ERRINT	Edge	389		5		11 to 10		15 to 8
Direct memory access controller (Non-secure)	10	DMAINT10	Edge	390		6		13 to 12		23 to 16
	11	DMAINT11	Edge	391		7		15 to 14		31 to 24
	12	DMAINT12	Edge	392		8		17 to 16	98	7 to 0
	13	DMAINT13	Edge	393		9		19 to 18		15 to 8
	14	DMAINT14	Edge	394		10		21 to 20		23 to 16
	15	DMAINT15	Edge	395		11		23 to 22		31 to 24
Internal bus		H2XDAV0_ERRINT	Edge	396		12		25 to 24	99	7 to 0
		H2XDAV1_ERRINT	Edge	397		13		27 to 26		15 to 8
Reserved				398		14		29 to 28		23 to 16
				399		15		31 to 30		31 to 24
				400		16	25	1 to 0	100	7 to 0
				401		17		3 to 2		15 to 8
				402		18		5 to 4		23 to 16
				403		19		7 to 6		31 to 24
				404		20		9 to 8	101	7 to 0
				405		21		11 to 10		15 to 8
				406		22		13 to 12		23 to 16
				407		23		15 to 14		31 to 24
				408		24		17 to 16	102	7 to 0
				409		25		19 to 18		15 to 8
				410		26		21 to 20		23 to 16
				411		27		23 to 22		31 to 24
				412		28		25 to 24	103	7 to 0
				413		29		27 to 26		15 to 8
				414		30		29 to 28		23 to 16
				415		31		31 to 30		31 to 24
				416	13	0	26	1 to 0	104	7 to 0
				417		1		3 to 2		15 to 8
				418		2		5 to 4		23 to 16
				419		3		7 to 6		31 to 24
				420		4		9 to 8	105	7 to 0
				421		5		11 to 10		15 to 8
				422		6		13 to 12		23 to 16
				423		7		15 to 14		31 to 24
				424		8		17 to 16	106	7 to 0
				425		9		19 to 18		15 to 8
				426		10		21 to 20		23 to 16
				427		11		23 to 22		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation					
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
Reserved				428	13	12	26	25 to 24	107	7 to 0
				429		13		27 to 26		15 to 8
				430		14		29 to 28		23 to 16
				431		15		31 to 30		31 to 24
				432		16	27	1 to 0	108	7 to 0
				433		17		3 to 2		15 to 8
Internal bus		PRRI	Level	434		18		5 to 4		23 to 16
Trusted Secure IP *1		ROMOK	Edge	435		19		7 to 6		31 to 24
		LONG_PLG	Edge	436		20		9 to 8	109	7 to 0
		PROC_BUSY	Edge	437		21		11 to 10		15 to 8
		RDRDY1	Edge	438		22		13 to 12		23 to 16
		RDRDY0	Edge	439		23		15 to 14		31 to 24
		WRRDY4	Edge	440		24		17 to 16	110	7 to 0
		Reserved		441		25		19 to 18		15 to 8
		WRRDY1	Edge	442		26		21 to 20		23 to 16
		WRRDY0	Edge	443		27		23 to 22		31 to 24
		IRDRDY	Edge	444		28		25 to 24	111	7 to 0
		IWRRDY	Edge	445		29		27 to 26		15 to 8
Dynamic Reconfigurable Processor (DRP) *2		ERRINT	Level	446		30		29 to 28		23 to 16
		NMLINT	Level	447		31		31 to 30		31 to 24
		PAF5	Level	448	14	0	28	1 to 0	112	7 to 0
		PAE5	Level	449		1		3 to 2		15 to 8
		INTB5	Level	450		2		5 to 4		23 to 16
		INTA5	Level	451		3		7 to 6		31 to 24
		PAF4	Level	452		4		9 to 8	113	7 to 0
		PAE4	Level	453		5		11 to 10		15 to 8
		INTB4	Level	454		6		13 to 12		23 to 16
		INTA4	Level	455		7		15 to 14		31 to 24
		PAF3	Level	456		8		17 to 16	114	7 to 0
		PAE3	Level	457		9		19 to 18		15 to 8
		INTB3	Level	458		10		21 to 20		23 to 16
		INTA3	Level	459		11		23 to 22		31 to 24
		PAF2	Level	460		12		25 to 24	115	7 to 0
		PAE2	Level	461		13		27 to 26		15 to 8
		INTB2	Level	462		14		29 to 28		23 to 16
		INTA2	Level	463		15		31 to 30		31 to 24
		PAF1	Level	464		16	29	1 to 0	116	7 to 0
		PAE1	Level	465		17		3 to 2		15 to 8
		INTB1	Level	466		18		5 to 4		23 to 16
		INTA1	Level	467		19		7 to 6		31 to 24
		PAF0	Level	468		20		9 to 8	117	7 to 0
		PAE0	Level	469		21		11 to 10		15 to 8
		INTB0	Level	470		22		13 to 12		23 to 16
		INTA0	Level	471		23		15 to 14		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation					
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
Video Input Module		VINI	Level	472	14	24	29	17 to 16	118	7 to 0
Port Output Enable for GPT		GROUP0	Level	473		25		19 to 18		15 to 8
		GROUP1	Level	474		26		21 to 20		23 to 16
		GROUP2	Level	475		27		23 to 22		31 to 24
		GROUP3	Level	476		28		25 to 24	119	7 to 0
SPI Multi I/O Bus Controller		SPIHF	Level	477		29		27 to 26		15 to 8
Reserved				478		30		29 to 28		23 to 16
				479		31		31 to 30		31 to 24
Pin interrupts		TINT0	Edge/ Level	480	15	0	30	1 to 0	120	7 to 0
		TINT1	Edge/ Level	481		1		3 to 2		15 to 8
		TINT2	Edge/ Level	482		2		5 to 4		23 to 16
		TINT3	Edge/ Level	483		3		7 to 6		31 to 24
		TINT4	Edge/ Level	484		4		9 to 8	121	7 to 0
		TINT5	Edge/ Level	485		5		11 to 10		15 to 8
		TINT6	Edge/ Level	486		6		13 to 12		23 to 16
		TINT7	Edge/ Level	487		7		15 to 14		31 to 24
		TINT8	Edge/ Level	488		8		17 to 16	122	7 to 0
		TINT9	Edge/ Level	489		9		19 to 18		15 to 8
		TINT10	Edge/ Level	490		10		21 to 20		23 to 16
		TINT11	Edge/ Level	491		11		23 to 22		31 to 24
		TINT12	Edge/ Level	492		12		25 to 24	123	7 to 0
		TINT13	Edge/ Level	493		13		27 to 26		15 to 8
		TINT14	Edge/ Level	494		14		29 to 28		23 to 16
		TINT15	Edge/ Level	495		15		31 to 30		31 to 24
		TINT16	Edge/ Level	496		16	31	1 to 0	124	7 to 0
		TINT17	Edge/ Level	497		17		3 to 2		15 to 8
		TINT18	Edge/ Level	498		18		5 to 4		23 to 16
		TINT19	Edge/ Level	499		19		7 to 6		31 to 24
		TINT20	Edge/ Level	500		20		9 to 8	125	7 to 0

Table 7.3 List of Interrupt IDs

Interrupt Source					Register Allocation					
Module	Channel	Request Source Name	Interrupt Request Edge/ Level	Interrupt ID	GICD_IGROUPRn GICD_ISENABLERn GICD_ICENABLERn GICD_ISPENDRn GICD_ICPENDRn GICD_ISACTIVERn	Bit	GICD_ ICFGRn	Bit	GICD_ IPRIORITYRn GICD_ ITARGETSRn	Bit
Pin interrupts		TINT21	Edge/ Level	501	15	21	31	11 to 10	125	15 to 8
		TINT22	Edge/ Level	502		22		13 to 12		23 to 16
		TINT23	Edge/ Level	503		23		15 to 14		31 to 24
		TINT24	Edge/ Level	504		24		17 to 16	126	7 to 0
		TINT25	Edge/ Level	505		25		19 to 18		15 to 8
		TINT26	Edge/ Level	506		26		21 to 20		23 to 16
		TINT27	Edge/ Level	507		27		23 to 22		31 to 24
		TINT28	Edge/ Level	508		28		25 to 24	127	7 to 0
		TINT29	Edge/ Level	509		29		27 to 26		15 to 8
		TINT30	Edge/ Level	510		30		29 to 28		23 to 16
		TINT31	Edge/ Level	511		31		31 to 30		31 to 24

Note 1. Only in products with a Trusted Secure IP
Note 2. Only in products with a DRP

7.6 Operation

7.6.1 Interrupt Settings

Figure 7.2 shows the procedure of the initial settings for the interrupt controller.
Figure 7.3 shows the procedure of the settings when a pin interrupt or IRQ interrupt is to be used, and the procedure for setting the senses of NMI and IRQ interrupts.

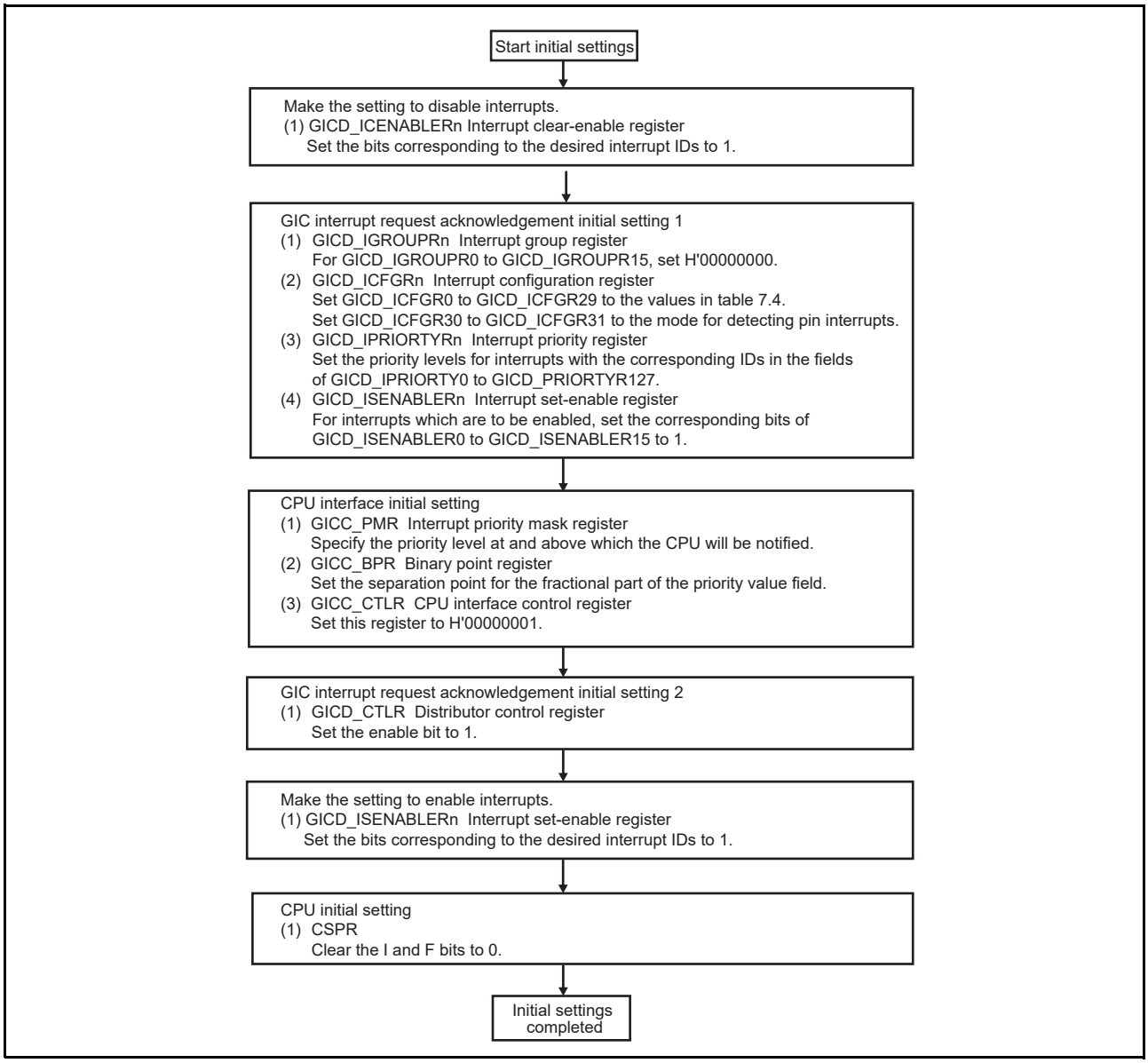


Figure 7.2 Flow of Initial Settings

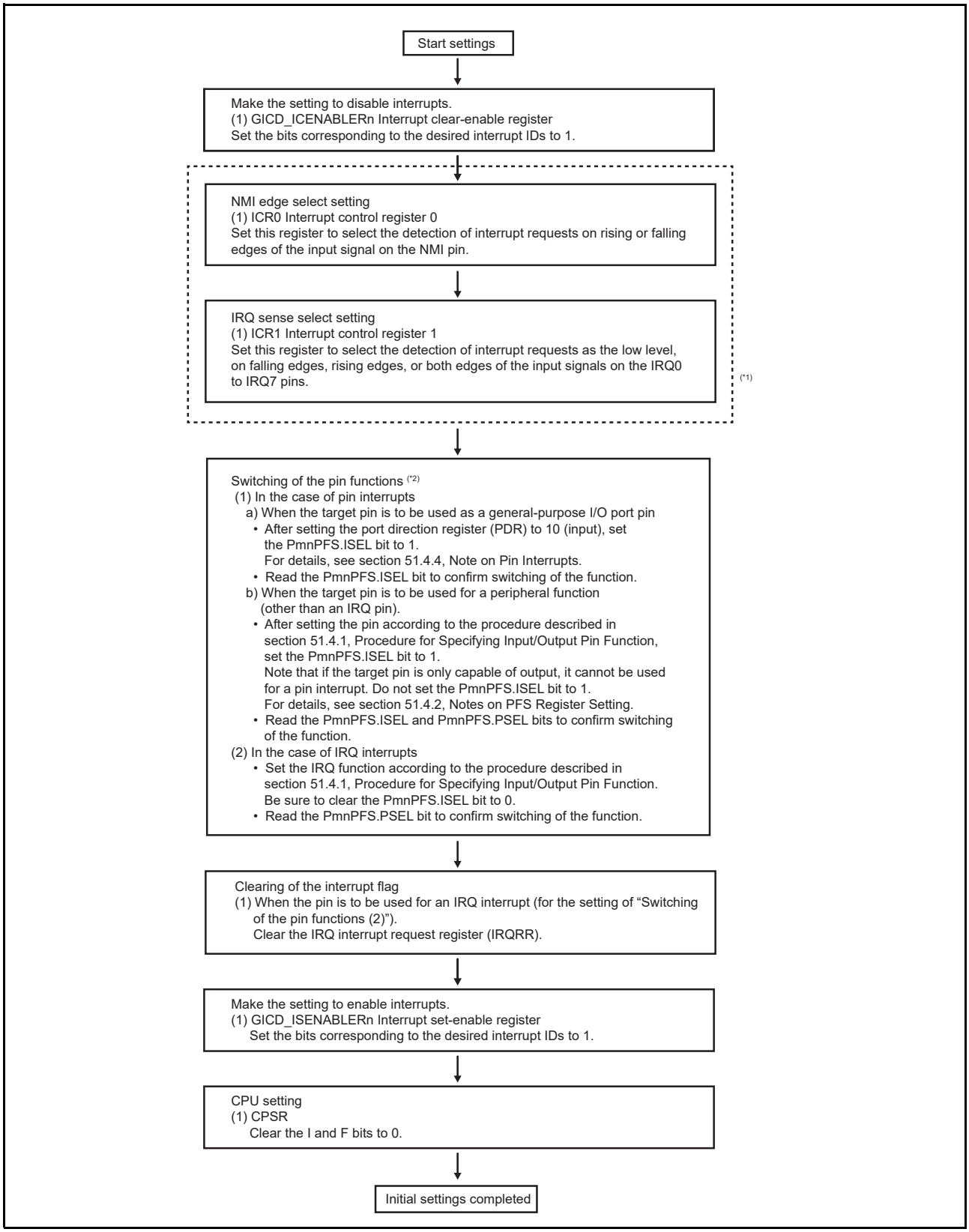


Figure 7.3 Flow of Switching Pin Functions *3

Note 1. If the settings in the broken lines have already been completed when the pin function is switched, that part of the procedure can be omitted.
Note 2. For the register of the general-purpose input/output port, see section 51.3, Register Description.
Note 3. From the start of the settings to the end of the settings, keep pins for use as pin interrupts at the low level, and pins for use as IRQ interrupts at the high level.

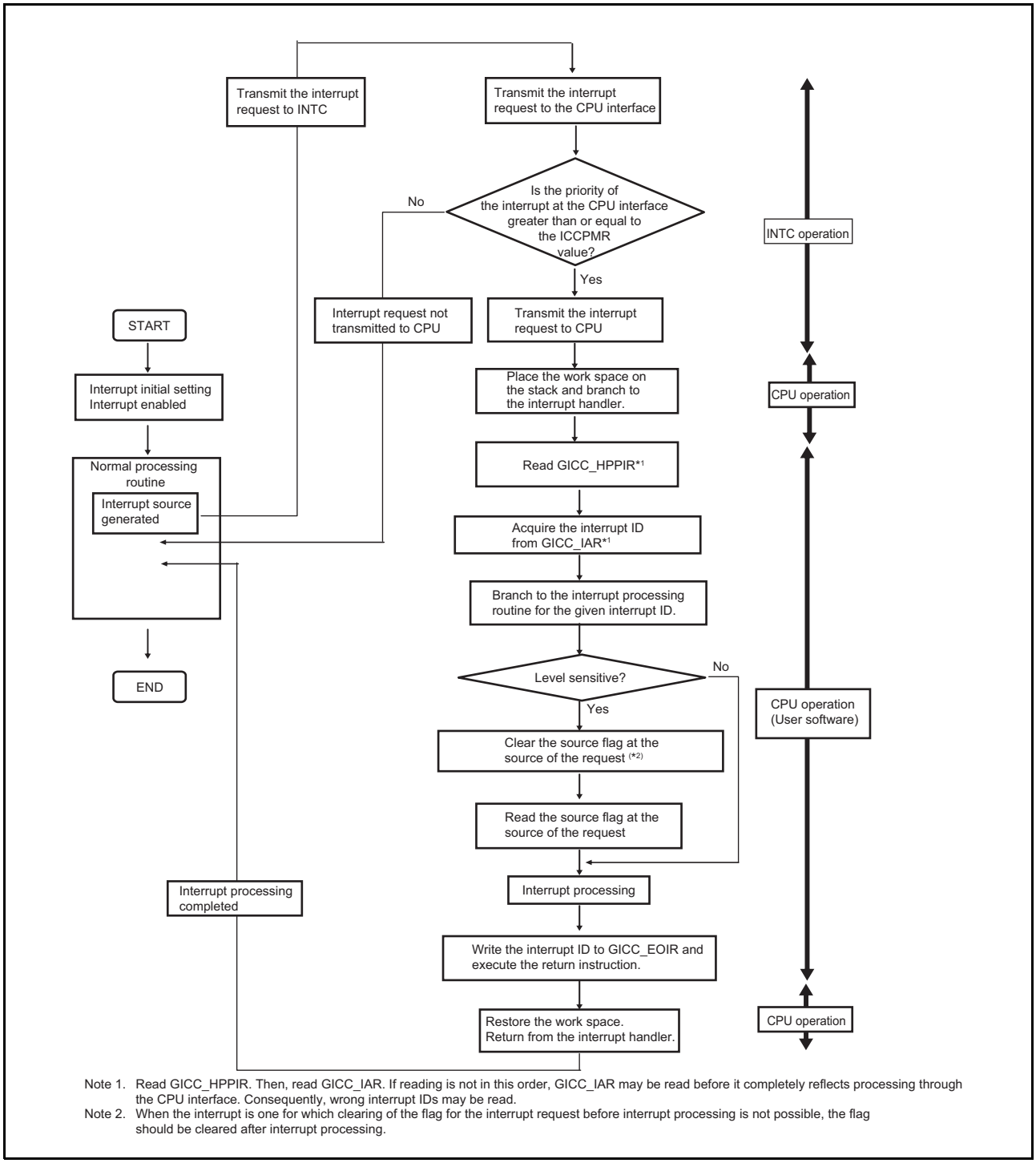
Table 7.4 GICD_ICFGRn Interrupt Configuration Register Settings

Register Name	Setting	Interrupt ID
GICD_ICFGR0	H'AAAA_AAAA	15 to 0
GICD_ICFGR1	H'5554_0000	31 to 16
GICD_ICFGR2	H'FD55_5555	47 to 32
GICD_ICFGR3	H'7FFF_FFFF	63 to 48
GICD_ICFGR4	H'5555_5555	79 to 64
GICD_ICFGR5	H'D57F_5555	95 to 80
GICD_ICFGR6	H'FFFF_FFFF	111 to 96
GICD_ICFGR7	H'FFFF_FFFF	127 to 112
GICD_ICFGR8	H'FFFF_FFFF	143 to 128
GICD_ICFGR9	H'FFFF_FFFF	159 to 144
GICD_ICFGR10	H'FFFF_FFFF	175 to 160
GICD_ICFGR11	H'FFFF_FFFF	191 to 176
GICD_ICFGR12	H'FFFF_FFFF	207 to 192
GICD_ICFGR13	H'FFFF_FFFF	223 to 208
GICD_ICFGR14	H'FFFF_FFFF	239 to 224
GICD_ICFGR15	H'7D5F_D57F	255 to 240
GICD_ICFGR16	H'557D_7DDF	271 to 256
GICD_ICFGR17	H'557D_557D	287 to 272
GICD_ICFGR18	H'5555_557D	303 to 288
GICD_ICFGR19	H'5555_5555	319 to 304
GICD_ICFGR20	H'F555_5555	335 to 320
GICD_ICFGR21	H'5555_FFFF	351 to 336
GICD_ICFGR22	H'F555_5555	367 to 352
GICD_ICFGR23	H'FFDD_55F5	383 to 368
GICD_ICFGR24	H'FFFF_FFFF	399 to 384
GICD_ICFGR25	H'FFFF_FFFF	415 to 400
GICD_ICFGR26	H'FFFF_FFFF	431 to 416
GICD_ICFGR27	H'5FFF_FFDF	447 to 432
GICD_ICFGR28	H'5555_5555	463 to 448
GICD_ICFGR29	H'5555_5555	479 to 464
GICD_ICFGR30*1	H'5555_5555	495 to 480
GICD_ICFGR31*1	H'5555_5555	511 to 496

Note 1. Edge or level detection can be selected for IDs corresponding to pin interrupts. The settings in the above table select level detection.

7.6.2 Flow of Interrupt Operations

For details on operation involved in interrupt generation, see the Arm Generic Interrupt Controller Architecture Specification and the PrimeCell® Generic Interrupt Controller (GIC-400) Technical Reference Manual from Arm. Figure 7.4 shows the flow of interrupt operations.



Note 1. Read GICC_HPPIR. Then, read GICC_IAR. If reading is not in this order, GICC_IAR may be read before it completely reflects processing through the CPU interface. Consequently, wrong interrupt IDs may be read.
Note 2. When the interrupt is one for which clearing of the flag for the interrupt request before interrupt processing is not possible, the flag should be cleared after interrupt processing.

Figure 7.4 Flow of Interrupt Operations

7.7 Data Transfer with Interrupt Request Signals

Interrupt request signals can be used to activate the direct memory access controller and transfer data.
Interrupt sources for which the direct memory access controller is designated as the destination by DMA extension resource selectors 0 to 7 are masked and requests from them are not input to the interrupt controller.
Figure 7.5 shows a block diagram of interrupt control. For details, see section 9, Direct Memory Access Controller.

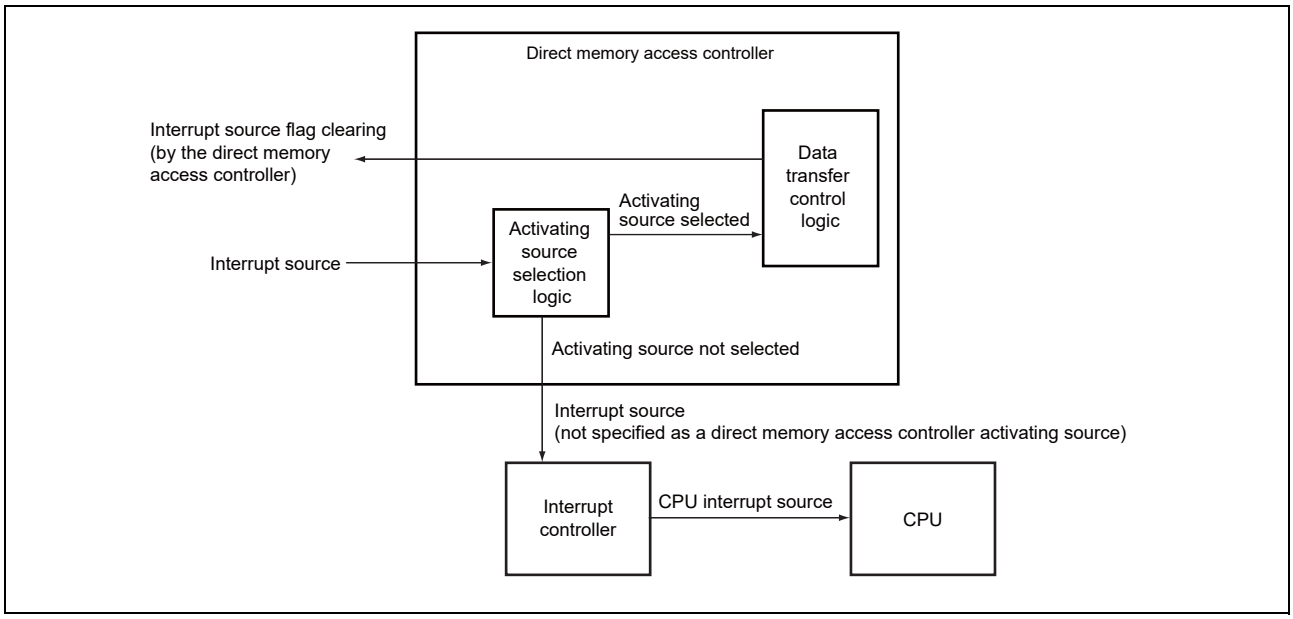


Figure 7.5 Interrupt Control Block Diagram

7.7.1 Handling Interrupt Request Signals as Sources for CPU Interrupt but Not Direct Memory Access Controller Activating

1. Do not select direct memory access controller activating sources.
2. When interrupts occur, interrupt requests are sent to the CPU.
3. The CPU clears the interrupt source and performs the necessary processing in the interrupt exception service routine.

7.7.2 Handling Interrupt Request Signals as Sources for Activating Direct Memory Access Controller but Not CPU Interrupt

1. Select direct memory access controller activating sources.
2. Activating sources are applied to the direct memory access controller when interrupts occur.
3. The direct memory access controller clears the interrupt sources when starting transfer.

7.8 Usage Note

7.8.1 Timing to Clear Interrupt Source

Clear the interrupt source flag to 0 in the interrupt exception handler. It takes some time to clear an interrupt in the CPU after clearing the interrupt source flag to 0. Read the interrupt source flag after clearing it to ensure that the interrupt request that should have been cleared is not received again erroneously. After that, execute the return instruction.

7.8.2 Notes on Selecting IRQ Interrupt Pin Functions

While the interrupt control register 1 (ICR1) is set so that an interrupt request is detected on the falling edge of an IRQn input and the current input on the pin is at the low level, the relevant edge will be detected when the pin function is switched to the IRQ interrupt function.

8. Bus State Controller

The bus state controller outputs control signals for various types of memory and external devices that are connected to the external address space. The functions of this module enable this LSI to connect directly with SRAM, SDRAM, and other memory storage devices, and external devices.

8.1 Features

1. External address space
   - A maximum of 64 Mbytes for each of areas CS0 to CS5.
   - Can specify the normal space interface, SRAM interface with byte selection, burst ROM (clocked synchronous or asynchronous), MPX-I/O, and SDRAM memory type for each address space.
   - Can select the data bus width (8 or 16 bits) for each of address spaces.
   - Controls insertion of wait cycles for each address space.
   - Controls insertion of wait cycles for each read access and write access.
   - Can set independent idle cycles during the continuous access for five cases: read-write (in same space/different spaces), read-read (in same space/different spaces), the first cycle is a write access.
2. Normal space interface
   - Supports the interface that can directly connect to the SRAM.
3. Burst ROM interface (clocked asynchronous)
   - High-speed access to the ROM that has the page mode function.
4. MPX-I/O interface
   - Can directly connect to a peripheral LSI that needs an address/data multiplexing.
5. SDRAM interface
   - Can set the SDRAM in up to two areas.
   - Multiplex output for row address/column address.
   - Efficient access by single read/single write.
   - High-speed access in bank-active mode.
   - Supports an auto-refresh and self-refresh.
   - Supports a power-down mode.
   - Issues MRS and EMRS commands.
6. SRAM interface with byte selection
   - Can connect directly to a SRAM with byte selection.
7. Burst ROM interface (clocked synchronous)
   - Can connect directly to a burst ROM of the clocked synchronous type.
8. Refresh function
   - Supports the auto-refresh and self-refresh functions.
   - Specifies the refresh interval using the refresh counter and clock selection.
   - Can execute concentrated refresh by specifying the refresh counts (1, 2, 4, 6, or 8).
9. Usage as interval timer for refresh counter
   - Generates an interrupt request at compare match.
10. Detection of long wait state for access by the signal on the external WAIT# pin.
    - A timeout detection condition is specifiable per CS space.
    - Once timeout is detected, the external WAIT function is disabled and a timeout detection interrupt request is issued.

Figure 8.1 shows a block diagram of this module.

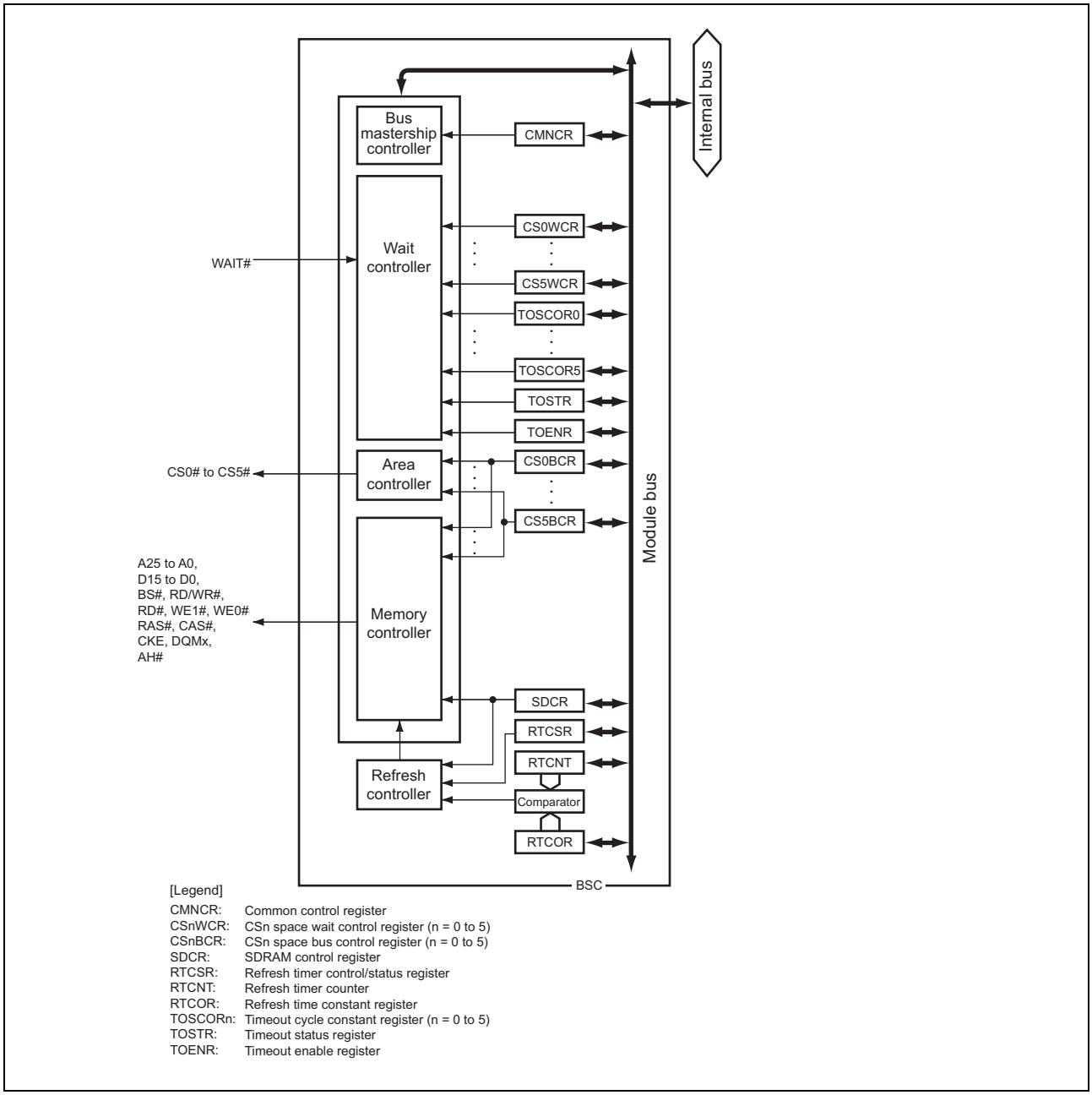


Figure 8.1 Block Diagram of Bus State Controller

8.2 Input/Output Pins

Table 8.1 shows the pin configuration.

Table 8.1 Pin Configuration

Name	I/O	Function
A25 to A0	Output	Address bus
D15 to D0	I/O	Data bus
BS#	Output	Bus cycle start
CS0# to CS5#	Output	Chip select
RD/WR#	Output	Read/write Connects to WE# pins when SDRAM or SRAM with byte selection is connected.
RD#	Output	Read pulse signal (read data output enable signal)
AH#	Output	Functions as the address hold signal when the MPX-I/O is used.
WE1#/DQMU	Output	Indicates that D15 to D8 are being written to. Connected to the byte select signal when a SRAM with byte selection is connected. Functions as the select signals for D15 to D8 when SDRAM is connected.
WE0#/DQML	Output	Indicates that D7 to D0 are being written to. Connected to the byte select signal when a SRAM with byte selection is connected. Functions as the select signals for D7 to D0 when SDRAM is connected.
RAS#	Output	Connects to RAS# pin when SDRAM is connected.
CAS#	Output	Connects to CAS# pin when SDRAM is connected.
CKE	Output	Connects to CKE pin when SDRAM is connected.
WAIT#	Input	External wait input

8.3 Area Overview

8.3.1 Address Map

In the architecture, this LSI has a 32-bit address space, which is divided into external memory spaces (SPI multi I/O bus space, on-chip large-capacity RAM, on-chip data retention RAM, on-chip peripheral modules, and reserved areas) according to the upper bits of the address.
See section 5, LSI Internal Bus for how to enable or disable caching for the CS0 to CS5 external address spaces.
The kind of memory to be connected and the data bus width are specified in each partial space. The address map for the external address space is listed below.

Table 8.2 Address Map

Internal Address	Space	Memory to be Connected
H'00000000 to H'03FFFFFF	CS0	Normal space, SRAM with byte selection, burst ROM (asynchronous or synchronous)
H'04000000 to H'07FFFFFF	CS1	Normal space, SRAM with byte selection
H'08000000 to H'0BFFFFFF	CS2	Normal space, SRAM with byte selection, SDRAM
H'0C000000 to H'0FFFFFFF	CS3	Normal space, SRAM with byte selection, SDRAM
H'10000000 to H'13FFFFFF	CS4	Normal space, SRAM with byte selection, burst ROM (asynchronous)
H'14000000 to H'17FFFFFF	CS5	Normal space, SRAM with byte selection, MPX-I/O
H'1F000000 to H'8FFFFFFF	Others	On-chip peripheral module, SPI multi I/O bus space, HyperFlash™ / HyperRAM™ space, OctaFlash™ / OctaRAM™ space, on-chip large-capacity RAM, on-chip data retention RAM, reserved area*1

Note 1. For the on-chip large-capacity RAM space and on-chip data retention RAM space, access the addresses shown in section 50, On-Chip RAM. For the on-chip peripheral module space, access the addresses shown in section 55, Register States. Do not access addresses which are not described in these sections. Otherwise, the correct operation cannot be guaranteed.

8.3.2 Data Bus Width and Related Pin Setting for Each Area Depending on Boot Mode

The initial state of data bus width and settings of the pins related to this module depends on boot mode. For boot mode, refer to section 3, Boot Mode.

In boot mode 0, the state of area 0 is fixed to the state with bus width of 16 bits, because this LSI is started up by the program stored in the ROM connected to area 0. The initial states of areas 1 to 5 are the same as that of area 0, but the bus width can be changed by the program. Immediately after a power-on reset in these modes, some of the address signals and the data-bus and the CS0# and RD# signals are automatically selected by default as the functions of the corresponding pins, since these signals are required to read ROM data from area 0. With the exception of these pins, the general purpose pin function is selected by default, and other required pin functions must be specified by the program.

Read access to area 0 is only permitted before the pin settings are completed.

In boot modes 1 to 7, the state of areas 0 to 5 can be changed from the initial state by the program, because the LSI is started by the program stored in the SPI serial memory, the NAND flash memory with the SD controller, or the NAND flash memory with the MMC controller. Since pin functions related to this module are not set automatically, they need to be set by the program. Do not access external address spaces before the pin settings are completed.

Table 8.3 shows the initial state by areas 0 to 5 in boot modes 0, and 1 to 7.

The sample access waveforms shown in this section include the pins such as BS#, RD/WR#, and WEn#. They are the waveforms when pin functions are assigned to the general I/O ports. For example, when 8-bit bus width is used in boot mode 0, setting for pin A0 is needed.

For details on pin function settings, see section 51, GPIO.

Table 8.3 Initial States by Areas in Boot Modes 0, and 1 to 7

Boot Mode	Item	Area 0	Areas 1 to 5
0	Data bus width	Fixed to 16 bits. Not changeable.	16 bits. Can be changed by program.
	Settings of pins related to this module	Pins A20 to A1, D15 to D0, CS0#, and RD# are set automatically. Other pins need to be set by program.	
1 to 7	Data bus width	16 bits. Can be changed by program.	
	Settings of pins related to this module	General I/O function. For external bus access, all the necessary pins need to be set by program.	

Note 1. In boot mode 0, if a boot ROM that uses A21 and higher-order address lines is connected, the circuit board must include pull-down resistors for those address lines.

Note 2. The data-bus width may be limited by the type of memory in use. For details, see section 8.4.2, CSn Space Bus Control Register (CSnBCR) (n = 0 to 5).

8.4 Register Descriptions

Table 8.4 shows the register configuration of this module.
Do not access the areas until settings of the connected memory interface are completed.

Table 8.4 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Common control register	CMNCR	R/W	H'00001018	H'1F000000	32
CS0 space bus control register	CS0BCR	R/W	H'36DB0C00	H'1F000004	32
CS1 space bus control register	CS1BCR	R/W	H'36DB0C00	H'1F000008	32
CS2 space bus control register	CS2BCR	R/W	H'36DB0C00	H'1F00000C	32
CS3 space bus control register	CS3BCR	R/W	H'36DB0C00	H'1F000010	32
CS4 space bus control register	CS4BCR	R/W	H'36DB0C00	H'1F000014	32
CS5 space bus control register	CS5BCR	R/W	H'36DB0C00	H'1F000018	32
CS0 space wait control register	CS0WCR	R/W	H'00000500	H'1F000028	32
CS1 space wait control register	CS1WCR	R/W	H'00000500	H'1F00002C	32
CS2 space wait control register	CS2WCR	R/W	H'00000500	H'1F000030	32
CS3 space wait control register	CS3WCR	R/W	H'00000500	H'1F000034	32
CS4 space wait control register	CS4WCR	R/W	H'00000500	H'1F000038	32
CS5 space wait control register	CS5WCR	R/W	H'00000500	H'1F00003C	32
SDRAM control register	SDCR	R/W	H'00000000	H'1F00004C	32
Refresh timer control/status register	RTCSR	R/W	H'00000000	H'1F000050	32
Refresh timer counter	RTCNT	R/W	H'00000000	H'1F000054	32
Refresh time constant register	RTCOR	R/W	H'00000000	H'1F000058	32
Timeout cycle constant register 0	TOSCOR0	R/W	H'00000000	H'1F000060	32
Timeout cycle constant register 1	TOSCOR1	R/W	H'00000000	H'1F000064	32
Timeout cycle constant register 2	TOSCOR2	R/W	H'00000000	H'1F000068	32
Timeout cycle constant register 3	TOSCOR3	R/W	H'00000000	H'1F00006C	32
Timeout cycle constant register 4	TOSCOR4	R/W	H'00000000	H'1F000070	32
Timeout cycle constant register 5	TOSCOR5	R/W	H'00000000	H'1F000074	32
Timeout status register	TOSTR	R/W	H'00000000	H'1F000080	32
Timeout enable register	TOENR	R/W	H'00000000	H'1F000084	32
AC characteristics adjustment register	ACADJ	R/W	H'00000100	H'1F000090	32

8.4.1 Common Control Register (CMNCR)

CMNCR is a 32-bit register that controls the common items for each area.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	TL0	—	—	—	AL0	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DPRTY[1:0]		—	—	—	—	—	—	—	HIZ MEM	HIZ CNT*
Initial value:	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	TL0	0	R/W	Transfer End Level Specifies the TEND0 signal output is high active or low active. 0: Low-active output from TEND0 1: High-active output from TEND0
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	AL0	0	R/W	Specifies the DACK0 (acknowledge) signal output is high active or low active. 0: Low-active output from DACK0 1: High-active output from DACK0
23 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10, 9	DPRTY[1:0]	00	R/W	DMA Burst Transfer Priority Specify the priority for a refresh request during DMA burst transfer. 0*: Accepts a refresh request during DMA burst transfer. 10: Does not accept a refresh request during DMA burst transfer. 11: Reserved (setting prohibited)
8 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4, 3	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1	HIZMEM	0	R/W	High-Z Memory Control Specifies the pin state in software standby mode or deep standby mode for A25 to A0, BS#, CSn#, RD/WR#, WEn# /DQMx/AH#, and RD#. 0: High impedance in software standby mode or deep standby mode. 1: Driven in software standby mode or deep standby mode
0	HIZCNT*	0	R/W	High-Z Control Specifies the state in software standby mode or deep standby mode for CKE, RAS#, and CAS#. 0: High impedance in software standby mode or deep standby mode for CKE, RAS#, and CAS#. 1: Driven in software standby mode or deep standby mode for CKE, RAS#, and CAS#.

Note: * For High-Z control of CKIO, see section 6, Clock Pulse Generator.

8.4.2 CSn Space Bus Control Register (CSnBCR) (n = 0 to 5)

CSnBCR is a 32-bit readable/writable register that specifies the memory connected to each space, the number of idle cycles between bus cycles, and the bus width.

Do not access external memory for the corresponding area until CSnBCR initial setting and pin setting are completed. Idle cycles may be inserted even when they are not specified. For details, see section 8.5.10, Wait between Access Cycles.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	IWW[2:0]			IWRWD[2:0]			IWRWS[2:0]			IWRRD[2:0]			IWRRS[2:0]		
Initial value:	0	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TYPE[2:0]			—	BSZ[1:0]		—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	1	1*	0*	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 28	IWW[2:0]	011	R/W	Idle Cycles between Write-Read Cycles and Write-Write Cycles These bits specify the number of idle cycles to be inserted after the access to a memory that is connected to the space. The target access cycles are the write-read cycle and write-write cycle. 000: No idle cycle inserted 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted
27 to 25	IWRWD[2:0]	011	R/W	Idle Cycles for Another Space Read-Write Specify the number of idle cycles to be inserted after the access to a memory that is connected to the space. The target access cycle is a read-write one in which continuous access cycles switch between different spaces. 000: No idle cycle inserted 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted
24 to 22	IWRWS[2:0]	011	R/W	Idle Cycles for Read-Write in the Same Space Specify the number of idle cycles to be inserted after the access to a memory that is connected to the space. The target cycle is a read-write cycle of which continuous access cycles are for the same space. 000: No idle cycle inserted 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted

Bit	Bit Name	Initial Value	R/W	Description
21 to 19	IWRRD[2:0]	011	R/W	Idle Cycles for Read-Read in Another Space Specify the number of idle cycles to be inserted after the access to a memory that is connected to the space. The target cycle is a read-read cycle of which continuous access cycles switch between different space. 000: No idle cycle inserted 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted
18 to 16	IWRRS[2:0]	011	R/W	Idle Cycles for Read-Read in the Same Space Specify the number of idle cycles to be inserted after the access to a memory that is connected to the space. The target cycle is a read-read cycle of which continuous access cycles are for the same space. 000: No idle cycle inserted 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 12	TYPE[2:0]	000	R/W	Specify the type of memory connected to a space. 000: Normal space 001: Burst ROM (clock asynchronous) 010: MPX-I/O 011: SRAM with byte selection 100: SDRAM 101: Reserved (setting prohibited) 110: Reserved (setting prohibited) 111: Burst ROM (clock synchronous) For details for memory type in each area, see Table 8.2. Note: When connecting the burst ROM to the CS0 space in boot mode 0, change the CS0WCR register to the settings by the burst ROM CS0WCR uses and then set TYPE[2:0] to the burst ROM setting. In boot modes 1 to 7, first set the CS0BCR register that includes the TYPE[2:0] bits and then set the CS0WCR register.
11	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
10, 9	BSZ[1:0]	10*	R/W	Data Bus Width Specification Specify the data bus widths of spaces. 00: Reserved (setting prohibited) 01: 8-bit size 10: 16-bit size 11: In MPX-I/O, the bus width can be specified by the address. Other than in MPX-I/O, setting is prohibited. Note 1. If area 5 is specified as MPX-I/O, the bus width can be specified as 8 bits or 16 bits by the address according to the SZSEL bit in CS5WCR by specifying the BSZ[1:0] bits to 11. The fixed bus width can be specified as 8 bits or 16 bits Note 2. In boot mode 0, the BSZ[1:0] bits settings in CS0BCR are ignored. Note 3. If area 2 or area 3 is specified as SDRAM space, the bus width can be specified as 16 bits only. Note 4. If area 0 is specified as clocked synchronous burst ROM space, the bus width can be specified as 16 bits only.
8 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

8.4.3 CSn Space Wait Control Register (CSnWCR) (n = 0 to 5)

CSnWCR specifies various wait cycles for memory access. The bit configuration of this register varies as shown below according to the memory type (TYPE2 to TYPE0) specified by the CSn space bus control register (CSnBCR). Specify CSnWCR before accessing the target area. Specify CSnBCR first, then specify CSnWCR.

(1) Normal Space, SRAM with Byte Selection, and MPX-I/O

- CS0WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—*	BAS	—	—	—*	—*
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		WR[3:0]				WM	—	—	—	—	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	—*	0	R/W	Reserved Set this bit to 0 when the interfaces for normal space or for SRAM with byte selection are used.
20	BAS	0	R/W	SRAM with Byte Selection Byte Access Select Specifies the WEn# and RD/WR# signal timing when the SRAM interface with byte selection is used. 0: Asserts the WEn# signal at the read/write timing and asserts the RD/WR# signal during the write access cycle. 1: Asserts the WEn# signal during the read/write access cycle and asserts the RD/WR# signal at the write timing.
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	—*	All 0	R/W	Reserved Set these bits to 0 when the interfaces for normal space or for SRAM with byte selection are used.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, CS0# Assertion to RD#, WEn# Assertion Specify the number of delay cycles from address and CS0# assertion to RD# and WEn# assertion. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	WR[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of cycles that are necessary for read/write access. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from RD# , WEn# Negation to Address, CS0# Negation Specify the number of delay cycles from RD# and WEn# negation to address and CS0# negation. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Note: * In boot mode 0, to connect the burst ROM to the CS0 space and switch to burst ROM interface after activation, set the TYPE[2:0] bits in CS0BCR after setting the burst number by the bits 20 and 21 and the burst wait cycle number by the bits 16 and 17. Do not write 1 to the reserved bits other than above bits.

- CS1WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	BAS	—	WW[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		WR[3:0]				WM	—	—	—	—	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	BAS	0	R/W	SRAM with Byte Selection Byte Access Select Specifies the WEn# and RD/WR# signal timing when the SRAM interface with byte selection is used. 0: Asserts the WEn# signal at the read/write timing and asserts the RD/WR# signal during the write access cycle. 1: Asserts the WEn#signal during the read/write access cycle and asserts the RD/WR# signal at the write timing.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	WW[2:0]	000	R/W	Number of Write Access Wait Cycles Specify the number of cycles that are necessary for write access. 000: The same cycles as WR[3:0] setting (number of read access wait cycles) 001: No cycle 010: 1 cycle 011: 2 cycles 100: 3 cycles 101: 4 cycles 110: 5 cycles 111: 6 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, CSn# Assertion to RD#, WEn# Assertion Specify the number of delay cycles from address and CSn# assertion to RD# and WEn# assertion. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	WR[3:0]	1010	R/W	Number of Read Access Wait Cycles Specify the number of cycles that are necessary for read access. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from RD#, WEn# Negation to Address, CSn# Negation Specify the number of delay cycles from RD# and WEn# negation to address and CSn# negation. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

- CS2WCR, CS3WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	BAS	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	WR[3:0]				WM	—	—	—	—	—	—
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	BAS	0	R/W	SRAM with Byte Selection Byte Access Select Specifies the WEn# and RD/WR# signal timing when the SRAM interface with byte selection is used. 0: Asserts the WEn# signal at the read timing and asserts the RD/WR# signal during the write access cycle. 1: Asserts the WEn# signal during the read access cycle and asserts the RD/WR# signal at the write timing.
19 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 7	WR[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of cycles that are necessary for read/write access. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- CS4WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	BAS	—	WW[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		WR[3:0]				WM	—	—	—	—	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	BAS	0	R/W	SRAM with Byte Selection Byte Access Select Specifies the WEn# and RD/WR# signal timing when the SRAM interface with byte selection is used. 0: Asserts the WEn# signal at the read timing and asserts the RD/WR# signal during the write access cycle. 1: Asserts the WEn# signal during the read access cycle and asserts the RD/WR# signal at the write timing.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	WW[2:0]	000	R/W	Number of Write Access Wait Cycles Specify the number of cycles that are necessary for write access. 000: The same cycles as WR[3:0] setting (number of read access wait cycles) 001: No cycle 010: 1 cycle 011: 2 cycles 100: 3 cycles 101: 4 cycles 110: 5 cycles 111: 6 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, CS4# Assertion to RD# , WE# Assertion Specify the number of delay cycles from address and CS4# assertion to RD# and WE# assertion. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	WR[3:0]	1010	R/W	Number of Read Access Wait Cycles Specify the number of cycles that are necessary for read access. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from RD# , WEn# Negation to Address, CS4# Negation Specify the number of delay cycles from RD# and WEn# negation to address and CS4# negation. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

- CS5WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	SZSEL	MPXW/ BAS	—	WW[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		WR[3:0]				WM	—	—	—	—	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	SZSEL	0	R/W	MPX-I/O Interface Bus Width Specification Specifies an address to select the bus width when the BSZ[1:0] of CS5BCR are specified as 11. This bit is valid only when area 5 is specified as MPX-I/O. 0: Selects the bus width by address A14 1: Selects the bus width by address A21 The relationship between the SZSEL bit and bus width selected by A14 or A21 are summarized below.
20	MPXW	0	R/W	MPX-I/O Interface Address Wait This bit setting is valid only when area 5 is specified as MPX-I/O. Specifies the address cycle insertion wait for MPX-I/O interface. 0: Inserts no wait cycle 1: Inserts 1 wait cycle
	BAS	0	R/W	SRAM with Byte Selection Byte Access Select This bit setting is valid only when area 5 is specified as SRAM with byte selection. Specifies the WEn# and RD/WR# signal timing when the SRAM interface with byte selection is used. 0: Asserts the WEn# signal at the read timing and asserts the RD/WR# signal during the write access cycle. 1: Asserts the WEn# signal during the read access cycle and asserts the RD/WR# signal at the write timing.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	WW[2:0]	000	R/W	Number of Write Access Wait Cycles Specify the number of cycles that are necessary for write access. 000: The same cycles as WR[3:0] setting (number of read access wait cycles) 001: No cycle 010: 1 cycle 011: 2 cycles 100: 3 cycles 101: 4 cycles 110: 5 cycles 111: 6 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

SZSEL	A14	A21	Bus Width
0	0	Not affected	8 bits
0	1	Not affected	16 bits
1	Not affected	0	8 bits
1	Not affected	1	16 bits

Bit	Bit Name	Initial Value	R/W	Description
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, CS5# Assertion to RD#, WE# Assertion Specify the number of delay cycles from address and CS5# assertion to RD# and WEn# assertion when area 5 is specified as normal space or SRAM with byte selection. Specify the number of delay cycles from the end of address cycle (Ta3) to RD# and WEn# assertion when area 5 is specified as MPx-I/O. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles
10 to 7	WR[3:0]	1010	R/W	Number of Read Access Wait Cycles Specify the number of cycles that are necessary for read access. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from RD#, WEn# Negation to Address, CS5# Negation Specify the number of delay cycles from RD# and WEn# negation to address and CS5# negation when area 5 is specified as normal space or SRAM with byte selection. Specify the number of delay cycles from RD# and WEn# negation to CS5# negation when area 5 is specified as MPx-I/O. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

(2) Burst ROM (Clocked Asynchronous)

- CS0WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	BST[1:0]		—	—	BW[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	W[3:0]				WM	—	—	—	—	—	—
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	BST[1:0]	00	R/W	Burst Count Specification Specify the burst count for 16-byte or more access. These bits must not be set to B'11, because B'11 setting is reserved.

Bus Width	BST[1:0]	Burst count (16-byte access)
8 bits	00	16 burst × one time
	01	4 burst × four times
16 bits	00	8 burst × one time
	01	2 burst × four times
	10	4-4 or 2-4-2 burst

Note: For details, see Table 8.13, Relationship between Bus Width, Access Size, and Number of Bursts.

19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	BW[1:0]	00	R/W	Number of Burst Wait Cycles Specify the number of wait cycles to be inserted between the second or subsequent access cycles in burst access. 00: No cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	W[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted in the first access cycle. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- CS4WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	BST[1:0]		—	—	BW[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		W[3:0]				WM	—	—	—	—	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	BST[1:0]	00	R/W	Burst Count Specification Specify the burst count for 16-byte or more access. These bits must not be set to B'11, because B'11 setting is reserved.

Bus Width	BST[1:0]	Burst count (16-byte access)
8 bits	00	16 burst × one time
	01	4 burst × four times
16 bits	00	8 burst × one time
	01	2 burst × four times
	10	4-4 or 2-4-2 burst

Note: For details, see Table 8.13, Relationship between Bus Width, Access Size, and Number of Bursts.

Bit	Bit Name	Initial Value	R/W	Description
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	BW[1:0]	00	R/W	Number of Burst Wait Cycles Specify the number of wait cycles to be inserted between the second or subsequent access cycles in burst access. 00: No cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, CS4# Assertion to RD#, WEn# Assertion Specify the number of delay cycles from address and CS4# assertion to RD# and WEn# assertion. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	W[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted in the first access cycle. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from RD#, WEn# Negation to Address, CS4# Negation Specify the number of delay cycles from RD# and WEn# negation to address and CS4# negation. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

(3) SDRAM*

- CS2WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	A2CL[1:0]		—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8, 7	A2CL[1:0]	10	R/W	CAS Latency for Area 2 Specify the CAS latency for area 2. 00: 1 cycle 01: 2 cycles 10: 3 cycles 11: 4 cycles
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: * If only one area is connected to the SDRAM, specify area 3. In this case, specify area 2 as normal space or SRAM with byte selection.

- CS3WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	WTRP[1:0]*		—	WTRCD[1:0]*		—	A3CL[1:0]		—	—	TRWL[1:0]*		—	WTRC[1:0]*	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14, 13	WTRP[1:0]*	00	R/W	Number of Auto-Precharge Completion Wait Cycles Specify the number of minimum precharge completion wait cycles as shown below. • From the start of auto-precharge and issuing of ACTV command for the same bank • From issuing of the PRE/PALL command to issuing of the ACTV command for the same bank • Till entering the power-down mode or deep power-down mode • From the issuing of PALL command to issuing REF command in auto refresh mode • From the issuing of PALL command to issuing SELF command in self refresh mode The setting for areas 2 and 3 is common. 00: No cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles
12	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
11, 10	WTRCD [1:0]*	01	R/W	Number of Wait Cycles between ACTV Command and READ(A)/WRIT(A) Command Specify the minimum number of wait cycles from issuing the ACTV command to issuing the READ(A)/WRIT(A) command. The setting for areas 2 and 3 is common. 00: No cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8, 7	A3CL[1:0]	10	R/W	CAS Latency for Area 3 Specify the CAS latency for area 3. 00: 1 cycle 01: 2 cycles 10: 3 cycles 11: 4 cycles
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4, 3	TRWL[1:0]*	00	R/W	Number of Auto-Precharge Startup Wait Cycles Specify the number of minimum auto-precharge startup wait cycles as shown below. • Cycle number from the issuance of the WRITA command by this LSI until the completion of auto-precharge in the SDRAM. Equivalent to the cycle number from the issuance of the WRITA command until the issuance of the ACTV command. Confirm that how many cycles are required between the WRITA command receive in the SDRAM and the auto-precharge activation, referring to each SDRAM data sheet. And set the cycle number so as not to exceed the cycle number specified by these bits. • Cycle number from the issuance of the WRIT command until the issuance of the PRE command. This is the case when accessing another low address in the same bank in bank active mode. The setting for areas 2 and 3 is common. 00: No cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1, 0	WTRC[1:0]*	00	R/W	Number of Idle Cycles from REF Command/Self-Refresh Release to ACTV/REF/MRS Command Specify the number of minimum idle cycles in the periods shown below. • From the issuance of the REF command until the issuance of the ACTV/REF/MRS command • From releasing self-refresh until the issuance of the ACTV/REF/MRS command. The setting for areas 2 and 3 is common. 00: 2 cycles 01: 3 cycles 10: 5 cycles 11: 8 cycles

Note: * If both areas 2 and 3 are specified as SDRAM, WTRP[1:0], WTRCD[1:0], TRWL[1:0], and WTRC[1:0] bit settings are used in both areas in common.
If only one area is connected to the SDRAM, specify area 3. In this case, specify area 2 as normal space or SRAM with byte selection.

(4) Burst ROM (Clocked Synchronous)

- CS0WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BW[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	W[3:0]				WM	—	—	—	—	—	—
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	BW[1:0]	00	R/W	Number of Burst Wait Cycles Specify the number of wait cycles to be inserted between the second or subsequent access cycles in burst access. 00: No cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 7	W[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted in the first access cycle. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

8.4.4 SDRAM Control Register (SDCR)

SDCR specifies the method to refresh and access SDRAM, and the types of SDRAMs to be connected.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	A2ROW[1:0]		—	A2COL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	DEEP	—	RFSH	RMODE	PDOWN	BACTV	—	—	—	A3ROW[1:0]		—	A3COL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20, 19	A2ROW[1:0]	00	R/W	Number of Bits of Row Address for Area 2 Specify the number of bits of row address for area 2. 00: 11 bits 01: 12 bits 10: 13 bits 11: Reserved (setting prohibited)
18	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
17, 16	A2COL[1:0]	00	R/W	Number of Bits of Column Address for Area 2 Specify the number of bits of column address for area 2. 00: 8 bits 01: 9 bits 10: 10 bits 11: Reserved (setting prohibited)
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	DEEP	0	R/W	Deep Power-Down Mode This bit is valid for low-power SDRAM. If the RFSH or RMODE bit is set to 1 while this bit is set to 1, the deep power-down entry command is issued and the low-power SDRAM enters the deep power-down mode. 0: Self-refresh mode 1: Deep power-down mode
12	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
11	RFSH	0	R/W	Refresh Control Specifies whether or not the refresh operation of the SDRAM is performed. 0: No refresh 1: Refresh
10	RMODE	0	R/W	Refresh Mode Specifies whether to perform auto-refresh or self-refresh when the RFSH bit is 1. When the RFSH bit is 1 and this bit is 1, self-refresh starts immediately. When the RFSH bit is 1 and this bit is 0, auto-refresh starts according to the contents that are set in registers RTCSR, RTCNT, and RTCOR. 0: Auto-refresh is performed 1: Self-refresh is performed
9	PDOWN	0	R/W	Power-Down Mode Specifies whether the SDRAM will enter the power-down mode after the access to the SDRAM. With this bit being set to 1, after the SDRAM is accessed, the CKE signal is driven low and the SDRAM enters the power-down mode. 0: The SDRAM does not enter the power-down mode after being accessed. 1: The SDRAM enters the power-down mode after being accessed.

Bit	Bit Name	Initial Value	R/W	Description
8	BACTV	0	R/W	Bank Active Mode Specifies to access whether in auto-precharge mode (using READA and WRITA commands) or in bank active mode (using READ and WRIT commands). 0: Auto-precharge mode (using READA and WRITA commands) 1: Bank active mode (using READ and WRIT commands) Note: Bank active mode can be set only for area 3. When both areas 2 and 3 are set to SDRAM, specify the auto-precharge mode.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4, 3	A3ROW[1:0]	00	R/W	Number of Bits of Row Address for Area 3 Specify the number of bits of the row address for area 3. 00: 11 bits 01: 12 bits 10: 13 bits 11: Reserved (setting prohibited)
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1, 0	A3COL[1:0]	00	R/W	Number of Bits of Column Address for Area 3 Specify the number of bits of the column address for area 3. 00: 8 bits 01: 9 bits 10: 10 bits 11: Reserved (setting prohibited)

8.4.5 Refresh Timer Control/Status Register (RTCSR)

RTCSR specifies various items about refresh for SDRAM.
When RTCSR is written, the upper 16 bits of the write data must be H'A55A to cancel write protection.
The phase of the clock for incrementing the count in the refresh timer counter (RTCNT) is adjusted only by a power-on reset. Note that there is an error in the time until the compare match flag is set for the first time after the timer is started with the CKS[2:0] bits being set to a value other than B'000.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CMF	CMIE	CKS[2:0]			RRC[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0.
7	CMF	0	R/W	Compare Match Flag Indicates that a compare match occurs between the refresh timer counter (RTCNT) and refresh time constant register (RTCOR). This bit is set or cleared in the following conditions. 0: Clearing condition: When 0 is written in CMF after reading out RTCSR during CMF = 1. 1: Setting condition: When the condition RTCNT = RTCOR is satisfied.
6	CMIE	0	R/W	Compare Match Interrupt Enable Enables or disables CMF interrupt requests when the CMF bit in RTCSR is set to 1. 0: Disables CMF interrupt requests. 1: Enables CMF interrupt requests.
5 to 3	CKS[2:0]	000	R/W	Clock Select Select the clock input to count-up the refresh timer counter (RTCNT). 000: Stop the counting-up 001: CKIOφ/4 010: CKIOφ/16 011: CKIOφ/64 100: CKIOφ/256 101: CKIOφ/1024 110: CKIOφ/2048 111: CKIOφ/4096
2 to 0	RRC[2:0]	000	R/W	Refresh Count Specify the number of continuous refresh cycles, when the refresh request occurs after the coincidence of the values of the refresh timer counter (RTCNT) and the refresh time constant register (RTCOR). These bits can make the period of occurrence of refresh long. 000: 1 time 001: 2 times 010: 4 times 011: 6 times 100: 8 times 101: Reserved (setting prohibited) 110: Reserved (setting prohibited) 111: Reserved (setting prohibited)

8.4.6 Refresh Timer Counter (RTCNT)

RTCNT is an 8-bit counter that increments using the clock selected by bits CKS[2:0] in RTCSR. When RTCNT matches RTCOR, RTCNT is cleared to 0. The value in RTCNT returns to 0 after counting up to 255. When the RTCNT is written, the upper 16 bits of the write data must be H'A55A to cancel write protection.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0.
7 to 0		All 0	R/W	8-Bit Counter

8.4.7 Refresh Time Constant Register (RTCOR)

RTCOR is an 8-bit register. When RTCOR matches RTCNT, the CMF bit in RTCSR is set to 1 and RTCNT is cleared to 0. When the RFSH bit in SDCR is 1, a memory refresh request is issued by this matching signal. This request is maintained until the refresh operation is performed. If the request is not processed when the next matching occurs, the previous request is ignored.

When the CMIE bit in RTCSR is set to 1, an interrupt request is issued by this matching signal. The request continues to be output until the CMF bit in RTCSR is cleared. Clearing the CMF bit only affects the interrupt request and does not clear the refresh request. Therefore, a combination of refresh request and interval timer interrupt can be specified so that the number of refresh requests are counted by using timer interrupts while refresh is performed periodically.

When RTCOR is written, the upper 16 bits of the write data must be H'A55A to cancel write protection.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0.
7 to 0		All 0	R/W	8-Bit Counter

8.4.8 Timeout Cycle Constant Register (TOSCORn) (n = 0 to 5)

TOSCORn is a 16-bit register the value of which is effective when the WM bit in the CSn space wait control register (CSnWCR) is 0 and the corresponding bit in the timeout enable register (TOENR) is 1. When the number of cycles of waiting due to the signal on the eternal wait input pin matches the setting of TOSCORn, external wait input is disabled to end the cycle of access, the timeout status flag for the corresponding space in the timeout status register (TOSTR) is set, and a timeout detection interrupt request is generated. The timeout detection interrupt request is retained until the corresponding bit in TOENR is set to 0 or 0 is written to the timeout status flag for the corresponding space. Note that timeout detection is enabled even while the timeout status flag for the corresponding space in TOSTR is 1, and external wait input is disabled in response to a further timeout.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0.
15 to 0		All 0	R/W	16-Bit Register H'0000: 65536 cycles H'0001: 1 cycle : H'FFFF: 65535 cycles

8.4.9 Timeout Status Register (TOSTR)

TOSTOR is an 8-bit register that holds the timeout status flags for the CS spaces. When the WM bit in the CSn space wait control register (CSnWCR) is 0 and the corresponding bit in the timeout enable register (TOENR) is 1 and the number of cycles of waiting in response to the signal on the eternal wait input matches the setting of TOSCORn, the timeout status flag for the corresponding space is set and a timeout detection interrupt request is generated. The only writable value for the timeout status flags is 0, which clears the flag. Writing 1 to a flag is ignored.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CS5 TOSTF	CS4 TOSTF	CS3 TOSTF	CS2 TOSTF	CS1 TOSTF	CS0 TOSTF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0.
5	CS5TOSTF	0	R/W	CS5 Space Timeout Status Flag Status flag that indicates that the number of cycles of waiting due to the input on the external wait pin during access to the CS5 space has matched the setting of the CS5 space timeout cycle constant register (TOSCOR5). This bit is set or cleared in the following conditions. 0: Clearing condition When 0 is written in CS5TOSTF. 1: Setting condition When the WM bit in the CS5 space wait control register (CS5WCR) is 0 and the CS5TOEN bit in the timeout enable register (TOENR) is 1, the number of cycles of waiting due to the input on the external wait pin during access to the CS5 space has matched the setting of TOSCOR5.
4	CS4TOSTF	0	R/W	CS4 Space Timeout Status Flag Status flag that indicates that the number of cycles of waiting due to the input on the external wait pin during access to the CS4 space has matched the setting of the CS4 space timeout cycle constant register (TOSCOR4). For the condition to set or clear this bit, refer to the description of CS5TOSTF.
3	CS3TOSTF	0	R/W	CS3 Space Timeout Status Flag Status flag that indicates that the number of cycles of waiting due to the input on the external wait pin during access to the CS3 space has matched the setting of the CS3 space timeout cycle constant register (TOSCOR3). For the condition to set or clear this bit, refer to the description of CS5TOSTF.
2	CS2TOSTF	0	R/W	CS2 Space Timeout Status Flag Status flag that indicates that the number of cycles of waiting due to the input on the external wait pin during access to the CS2 space has matched the setting of the CS2 space timeout cycle constant register (TOSCOR2). For the condition to set or clear this bit, refer to the description of CS5TOSTF.
1	CS1TOSTF	0	R/W	CS1 Space Timeout Status Flag Status flag that indicates that the number of cycles of waiting due to the input on the external wait pin during access to the CS1 space has matched the setting of the CS1 space timeout cycle constant register (TOSCOR1). For the condition to set or clear this bit, refer to the description of CS5TOSTF.
0	CS0TOSTF	0	R/W	CS0 Space Timeout Status Flag Status flag that indicates that the number of cycles of waiting due to the input on the external wait pin during access to the CS0 space has matched the setting of the CS0 space timeout cycle constant register (TOSCOR0). For the condition to set or clear this bit, refer to the description of CS5TOSTF.

8.4.10 Timeout Enable Register (TOENR)

TOENR is an 8-bit register that specifies enabling or disabling the detection of timeout for waiting in each of the CS spaces.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CS5 TOEN	CS4 TOEN	CS3 TOEN	CS2 TOEN	CS1 TOEN	CS0 TOEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0.
5	CS5TOEN	0	R/W	CS5 Space Timeout Detection Enable Specifies enabling or disabling the detection of timeout for waiting in the CS5 space. 0: The timeout detection is disabled. 1: The timeout detection is enabled.
4	CS4TOEN	0	R/W	CS4 Space Timeout Detection Enable Specifies enabling or disabling the detection of timeout for waiting in the CS4 space. 0: The timeout detection is disabled. 1: The timeout detection is enabled.
3	CS3TOEN	0	R/W	CS3 Space Timeout Detection Enable Specifies enabling or disabling the detection of timeout for waiting in the CS3 space. 0: The timeout detection is disabled. 1: The timeout detection is enabled.
2	CS2TOEN	0	R/W	CS2 Space Timeout Detection Enable Specifies enabling or disabling the detection of timeout for waiting in the CS2 space. 0: The timeout detection is disabled. 1: The timeout detection is enabled.
1	CS1TOEN	0	R/W	CS1 Space Timeout Detection Enable Specifies enabling or disabling the detection of timeout for waiting in the CS1 space. 0: The timeout detection is disabled. 1: The timeout detection is enabled.
0	CS0TOEN	0	R/W	CS0 Space Timeout Detection Enable Specifies enabling or disabling the detection of timeout for waiting in the CS0 space. 0: The timeout detection is disabled. 1: The timeout detection is enabled.

8.4.11 AC Characteristics Adjustment Register (ACADJ)

ACADJ is a register that adjusts the input/output timing of the bus state controller. It is used for AC characteristics adjustment for SDRAM interface.

Do not change for the initial values when SDRAM is not used.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	SDRODLY[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SDRIDLY[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	SDRODLY[3:0]	All 0	R/W	Output characteristics adjustment bit Adjust the output delay time. H'0: Initial setting. The SDRAM is not to be connected. H'2: The SDRAM is to be connected. Other than above: Setting prohibited
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	SDRIDLY[3:0]	All 0	R/W	Input characteristics adjustment bit Adjust the input data capture timing. H'0: Initial setting. The SDRAM is not to be connected. H'F: The SDRAM is to be connected. Other than above: Setting prohibited

8.5 Operation

8.5.1 Access Size and Data Alignment

This LSI supports little endian, in which the least significant byte (LSB) is that in the direction of the 0th address.
Data bus width can be selected from 8 bits and 16 bits for the normal memory and SRAM with byte selection. Data bus width can be selected from 16 bits for SDRAM. For MPX-I/O, the data bus width is fixed to either 8 or 16 bits, or made selectable as 8 bits or 16 bits by one of the address lines.
Data bus width varies depending on boot mode. For details, refer to section 8.3.2, Data Bus Width and Related Pin Setting for Each Area Depending on Boot Mode.
Data alignment is performed in accordance with the data bus width selected for the device. This also means that four read operations are required to read 32-bit data from a byte-width device. In this LSI, data alignment and conversion of data length is performed automatically between the respective interfaces.
Table 8.5 and Table 8.6 show the relationship between device data width and access unit.

Table 8.5 16-Bit External Device Access and Data Alignment in Little Endian

		Data Bus		Strobe Signals	
Operation		D15 to D8	D7 to D0	WE1#, DQMU	WE0#, DQML
8-bit access at address 0		—	Data 7 to 0	—	Assert
8-bit access at address 1		Data 7 to 0	—	Assert	—
8-bit access at address 2		—	Data 7 to 0	—	Assert
8-bit access at address 3		Data 7 to 0	—	Assert	—
16-bit access at address 0		Data 15 to 8	Data 7 to 0	Assert	Assert
16-bit access at address 2		Data 15 to 8	Data 7 to 0	Assert	Assert
32-bit access at address 0	1st access at address 0	Data 15 to 8	Data 7 to 0	Assert	Assert
	2nd access at address 2	Data 31 to 24	Data 23 to 16	Assert	Assert

Table 8.6 8-Bit External Device Access and Data Alignment in Little Endian

		Data Bus		Strobe Signals	
Operation		D15 to D8	D7 to D0	WE1# DQMU	WE0#, DQML
8-bit access at address 0		—	Data 7 to 0	—	Assert
8-bit access at address 1		—	Data 7 to 0	—	Assert
8-bit access at address 2		—	Data 7 to 0	—	Assert
8-bit access at address 3		—	Data 7 to 0	—	Assert
16-bit access at address 0	1st access at address 0	—	Data 7 to 0	—	Assert
	2nd access at address 1	—	Data 15 to 8	—	Assert
16-bit access at address 2	1st access at address 0	—	Data 7 to 0	—	Assert
	2nd access at address 1	—	Data 15 to 8	—	Assert
32-bit access at address 0	1st access at address 0	—	Data 7 to 0	—	Assert
	2nd access at address 1	—	Data 15 to 8	—	Assert
	3rd access at address 2	—	Data 23 to 16	—	Assert
	4th access at address 3	—	Data 31 to 24	—	Assert

8.5.2 Normal Space Interface

(1) Basic Timing

For access to a normal space, this LSI uses strobe signal output in consideration of the fact that mainly static RAM will be directly connected. When using SRAM with a byte-selection pin, see section 8.5.8, SRAM Interface with Byte Selection. Figure 8.2 shows the basic timings of normal space access. A no-wait normal access is completed in two cycles. The BS# signal is asserted for one cycle to indicate the start of a bus cycle.

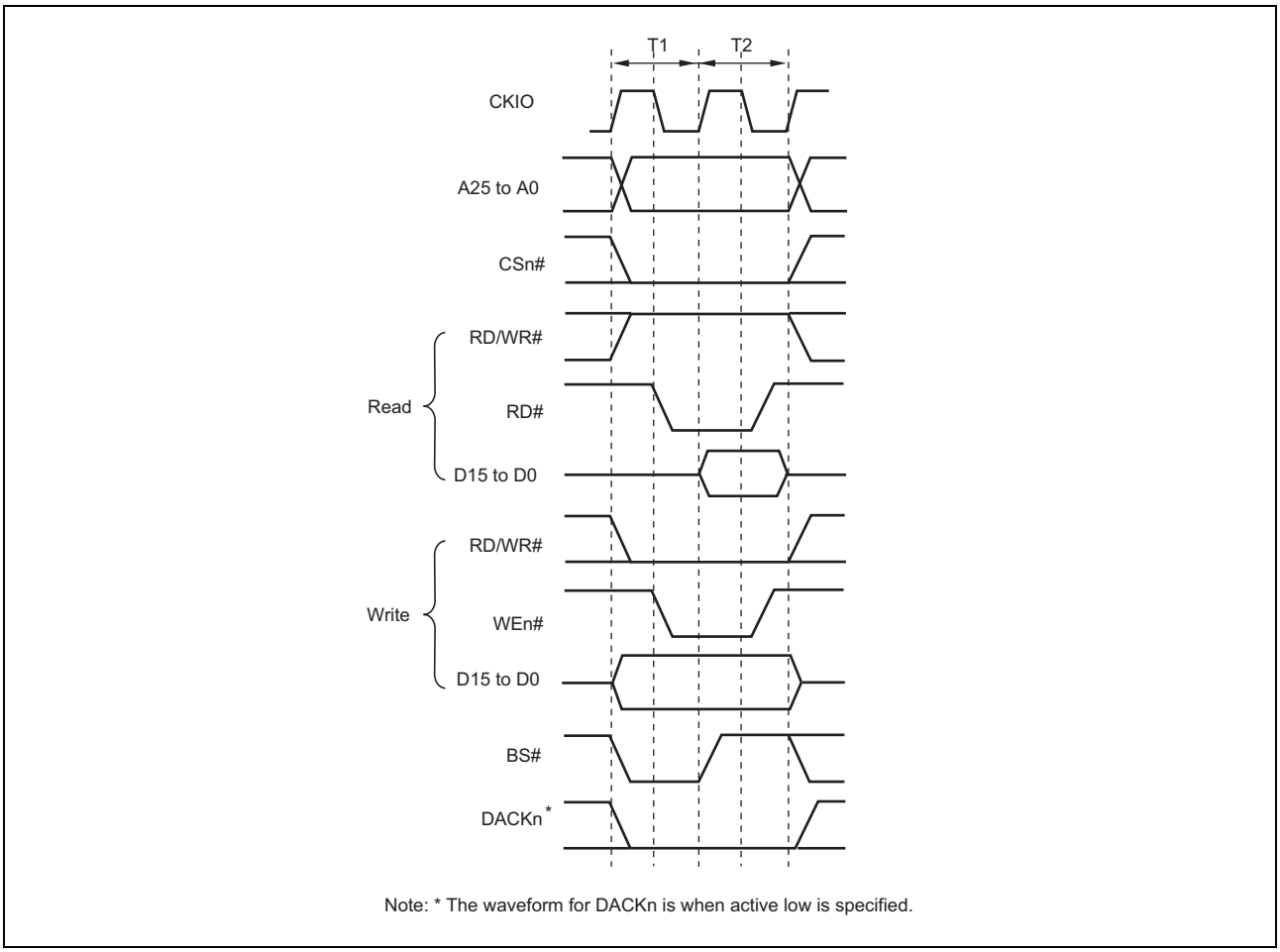


Figure 8.2 Normal Space Basic Access Timing (Access Wait 0)

There is no access size specification when reading. The correct access start address is output in the least significant bit of the address, but since there is no access size specification, 16 bits are always read in case of a 16-bit device. When writing, only the WEn# signal for the byte to be written is asserted.

It is necessary to output the data that has been read using RD# when a buffer is established in the data bus. The RD#/WR# signal is in a read state (high output) when no access has been carried out. Therefore, care must be taken when controlling the external data buffer with this signal, to avoid output collision.

Figure 8.3 and Figure 8.4 show the basic timings in continuous access to normal space. If the WM bit in CSnWCR is cleared to 0, a Tnop cycle is inserted after the CSn space access to evaluate the external wait (Figure 8.3). If the WM bit in CSnWCR is set to 1, external waits are ignored and no Tnop cycle is inserted (Figure 8.4).

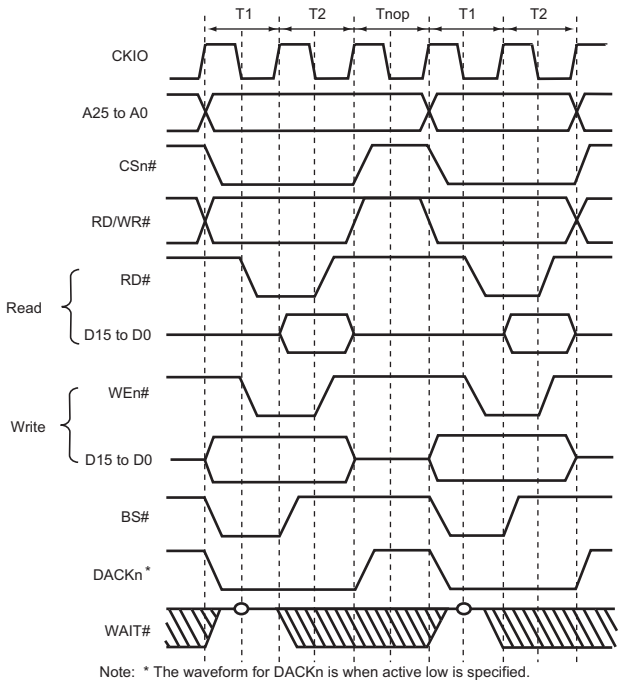


Figure 8.3 Continuous Access to Normal Space (1) Bus Width = 16 Bits, 32-Bit Access, CSnWCR.WM Bit = 0 (Access Wait = 0, Cycle Wait = 0)

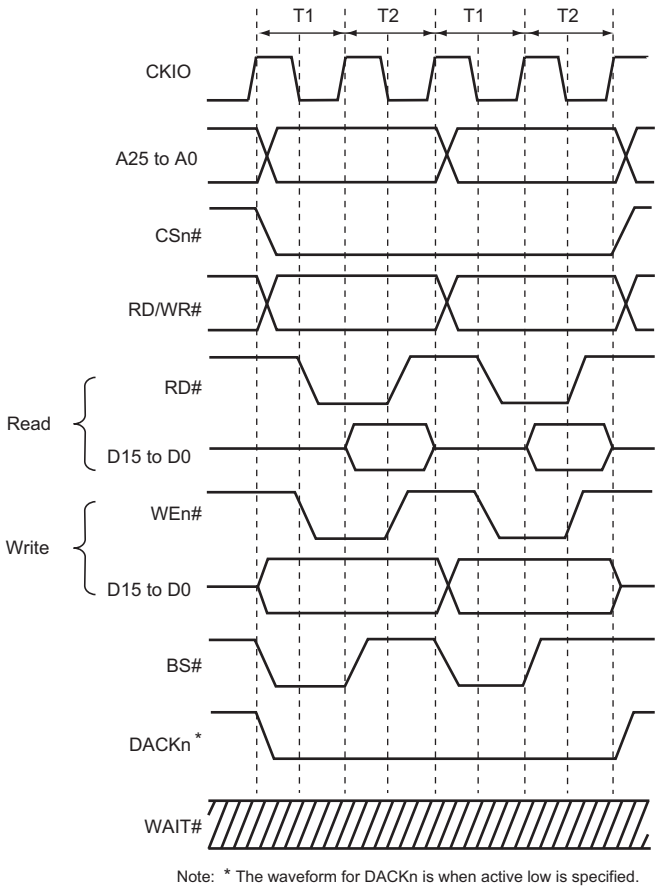


Figure 8.4 Continuous Access to Normal Space (2) Bus Width = 16 Bits, 32-Bit Access, CSnWCR.WM Bit = 1 (Access Wait = 0, Cycle Wait = 0)

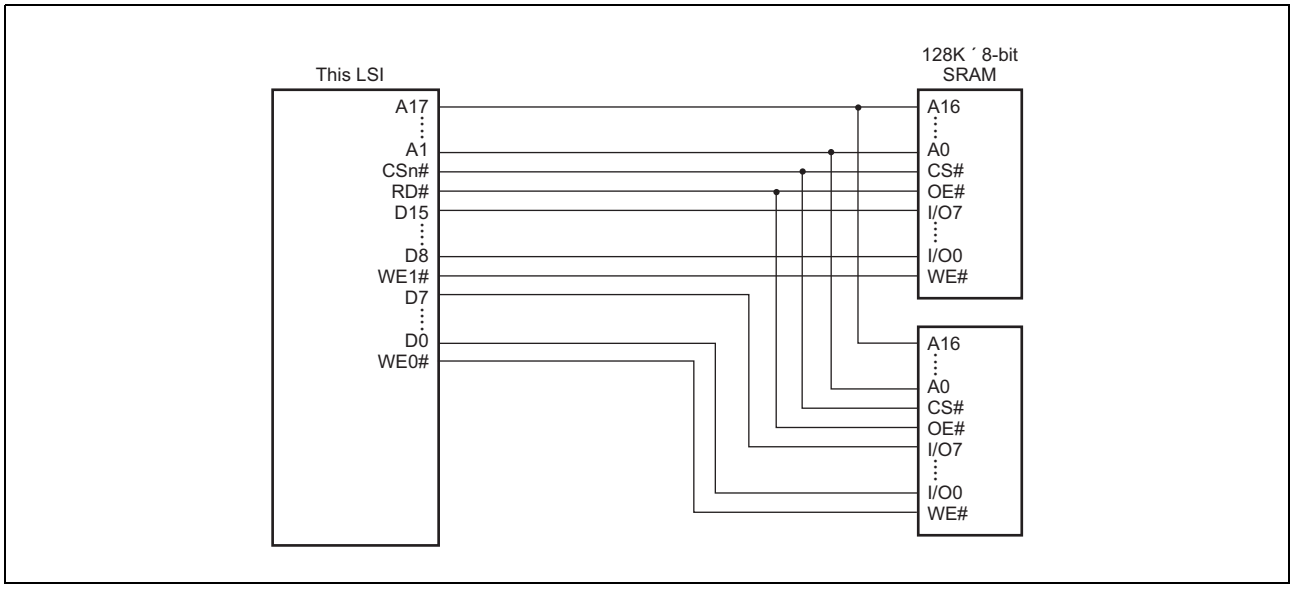


Figure 8.5 Example of 16-Bit Data-Width SRAM Connection

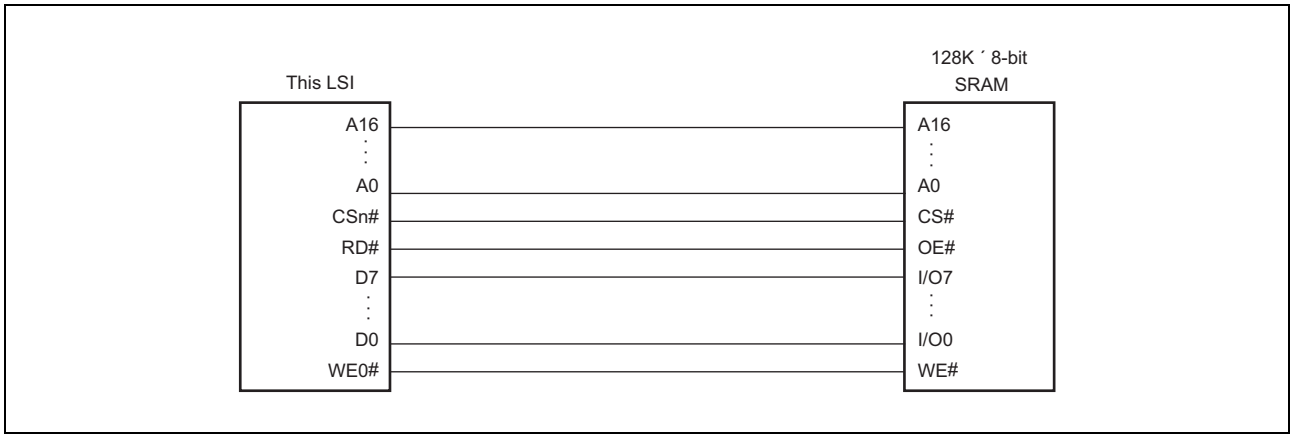


Figure 8.6 Example of 8-Bit Data-Width SRAM Connection

8.5.3 Access Wait Control

Wait cycle insertion on a normal space access can be controlled by the settings of bits WR3 to WR0 in CSnWCR. It is possible for areas 1, 4, and 5 to insert wait cycles independently in read access and in write access. Areas 0, 2, and 3 have common access wait for read cycle and write cycle. The specified number of Tw cycles are inserted as wait cycles in a normal space access shown in Figure 8.7.

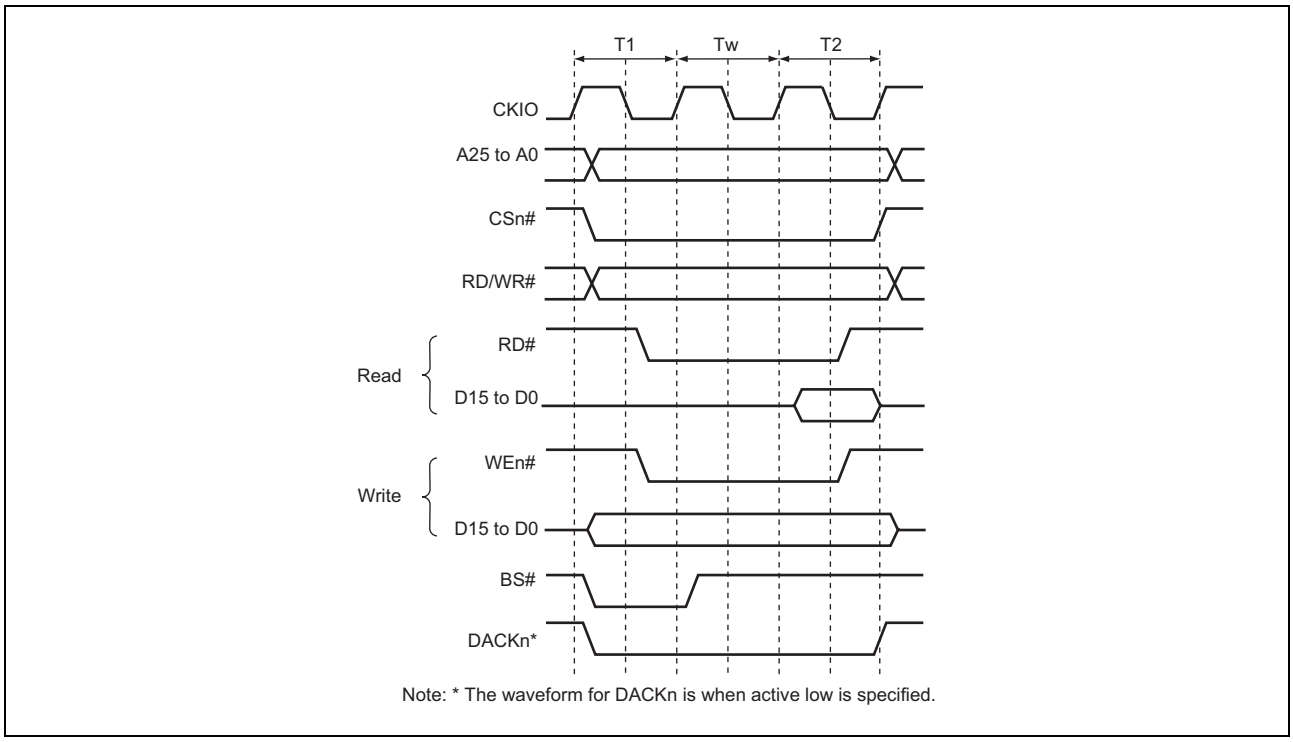


Figure 8.7 Wait Timing for Normal Space Access (Software Wait Only)

When the WM bit in CSnWCR is cleared to 0, the external wait input WAIT# signal is also sampled. WAIT# pin sampling is shown in Figure 8.8. A 2-cycle wait is specified as a software wait. The WAIT# signal is sampled on the falling edge of CKIO at the transition from the T1 or Tw cycle to the T2 cycle.

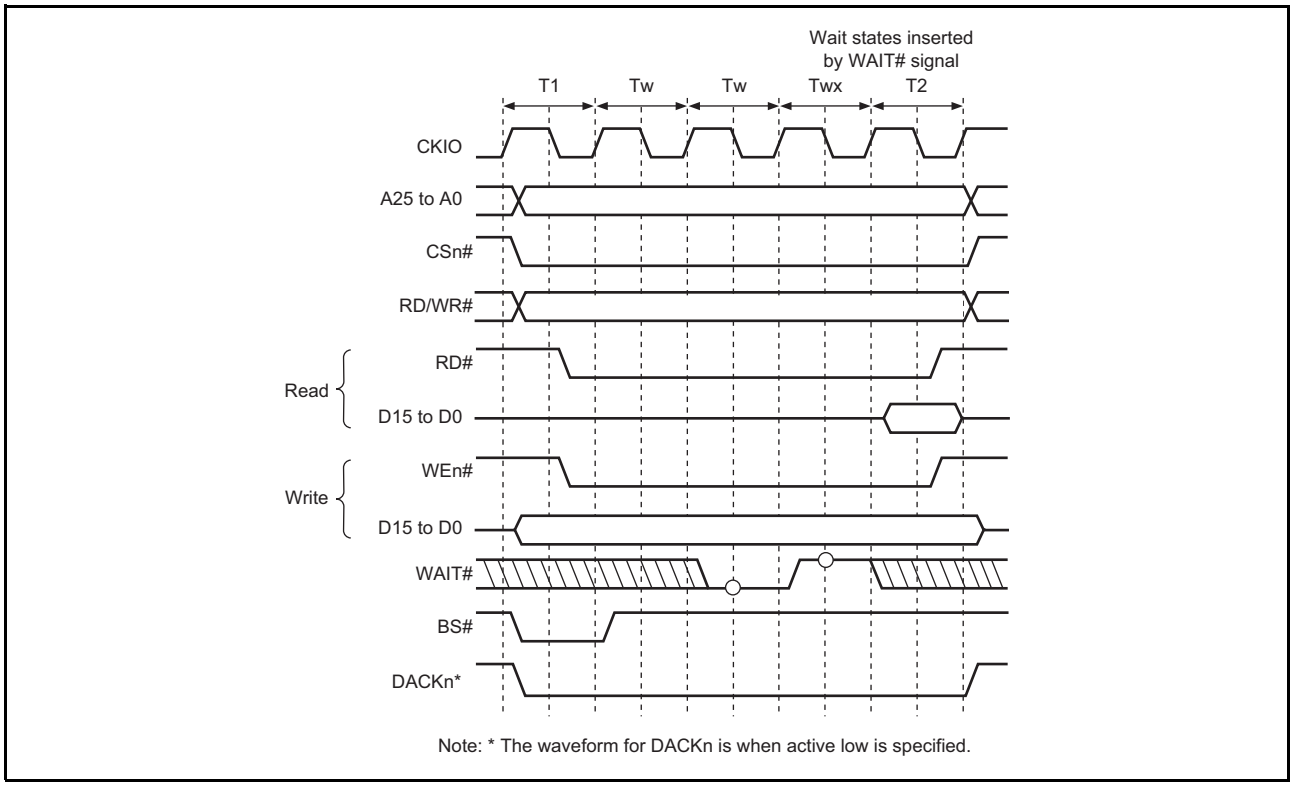


Figure 8.8 Wait Cycle Timing for Normal Space Access (Wait Cycle Insertion Using WAIT# Signal)

8.5.4 CSn# Assert Period Expansion

The number of cycles from CSn# assertion to RD#, WEn# assertion can be specified by setting bits SW1 and SW0 in CSnWCR. The number of cycles from RD#, WEn# negation to CSn# negation can be specified by setting bits HW1 and HW0. Therefore, a flexible interface to an external device can be obtained. Figure 8.9 shows an example. A Th cycle and a Tf cycle are added before and after an ordinary cycle, respectively. In these cycles, RD# and WEn# are not asserted, while other signals are asserted. The data output is prolonged to the Tf cycle, and this prolongation is useful for devices with slow writing operations.

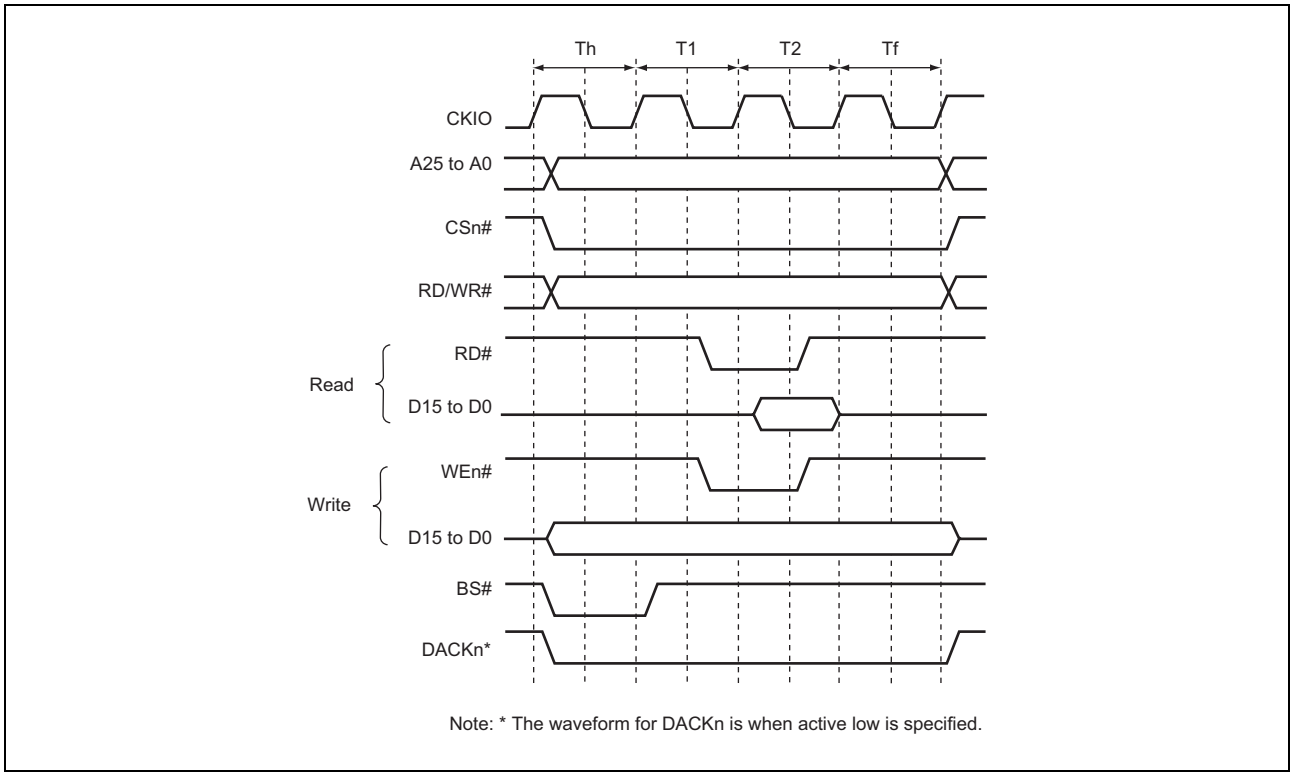


Figure 8.9 CSn# Assert Period Expansion

8.5.5 MPX-I/O Interface

Access timing for the MPX space is shown below. In the MPX space, CS5#, AH#, RD#, and WEn# signals control the accessing. The basic access for the MPX space consists of 2 cycles of address output followed by an access to a normal space. The bus width for the address output cycle or the data input/output cycle is fixed to 8 bits or 16 bits. Alternatively, it can be 8 bits or 16 bits depending on the address to be accessed.

Output of the addresses D15 to D0 or D7 to D0 is performed from cycle Ta2 to cycle Ta3. Because cycle Ta1 has a high-impedance state, collisions of addresses and data can be avoided without inserting idle cycles, even in continuous access cycles. Address output is increased to 3 cycles by setting the MPXW bit in CS5WCR to 1.

The RD/WR# signal is output at the same time as the CS5# signal; it is high in the read cycle and low in the write cycle. The data cycle is the same as that in a normal space access.

The delay cycles the number of which is specified by SW[1:0] are inserted between cycle Ta3 and cycle T1. The delay cycles the number of which is specified by HW[1:0] are added after cycle T2.

Timing charts are shown in Figure 8.10 to Figure 8.13.

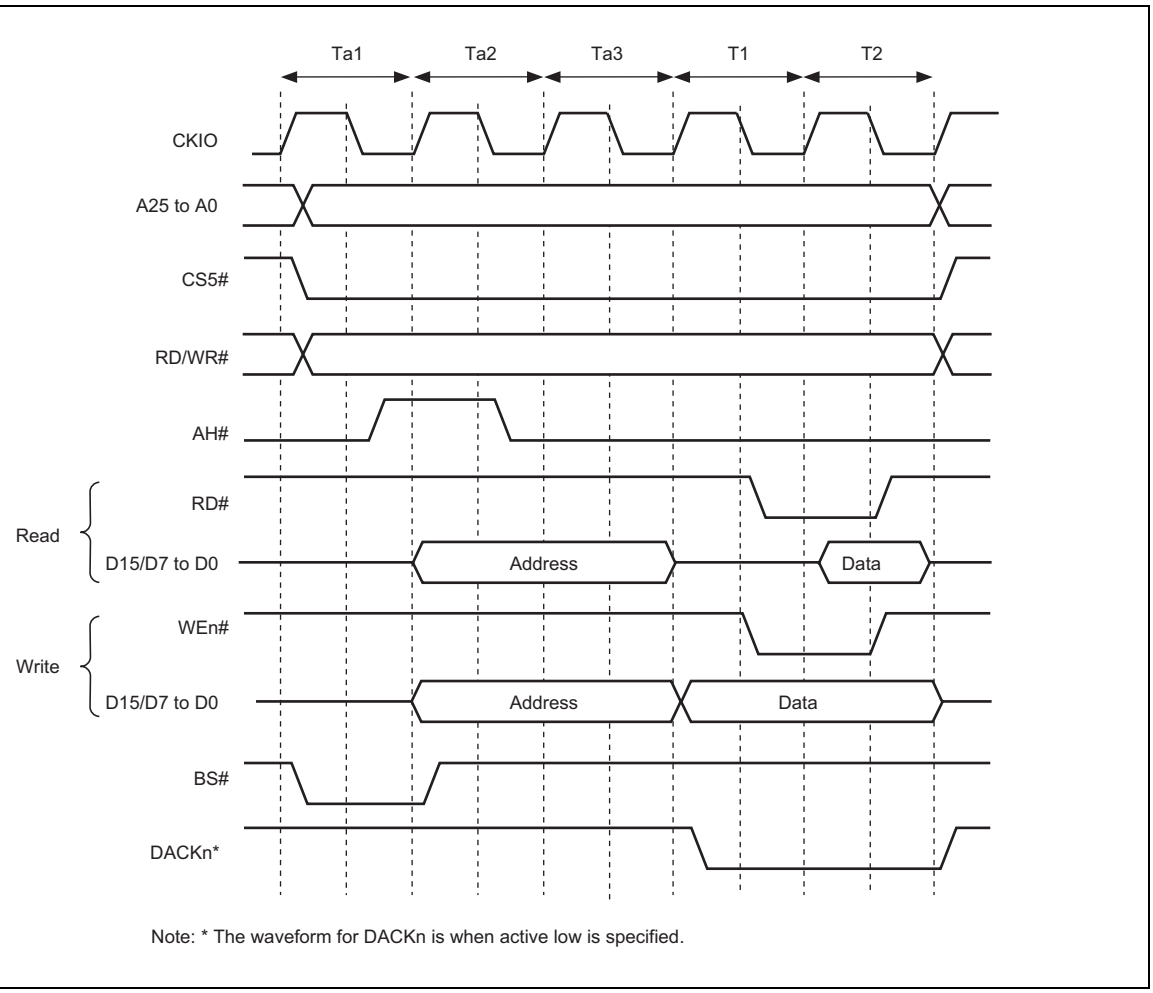


Figure 8.10 (1) Access Timing for MPX Space (Address Cycle No Wait, Data Cycle No Wait)

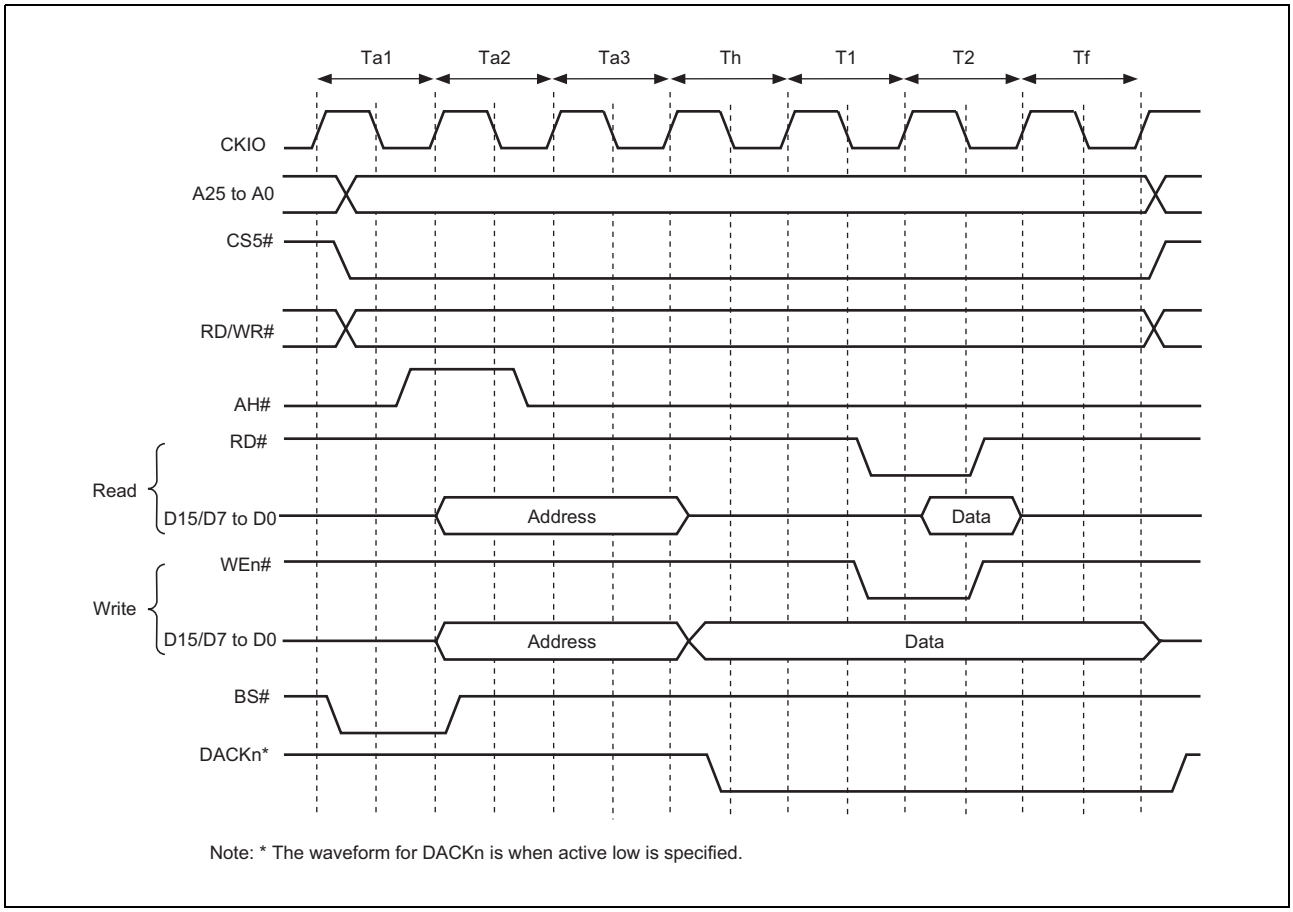


Figure 8.11 (2) Access Timing for MPX Space (Address Cycle No Wait, Extended Assertion Cycle 1.5, Data Cycle No Wait, Extended Negation Cycle 1.5)

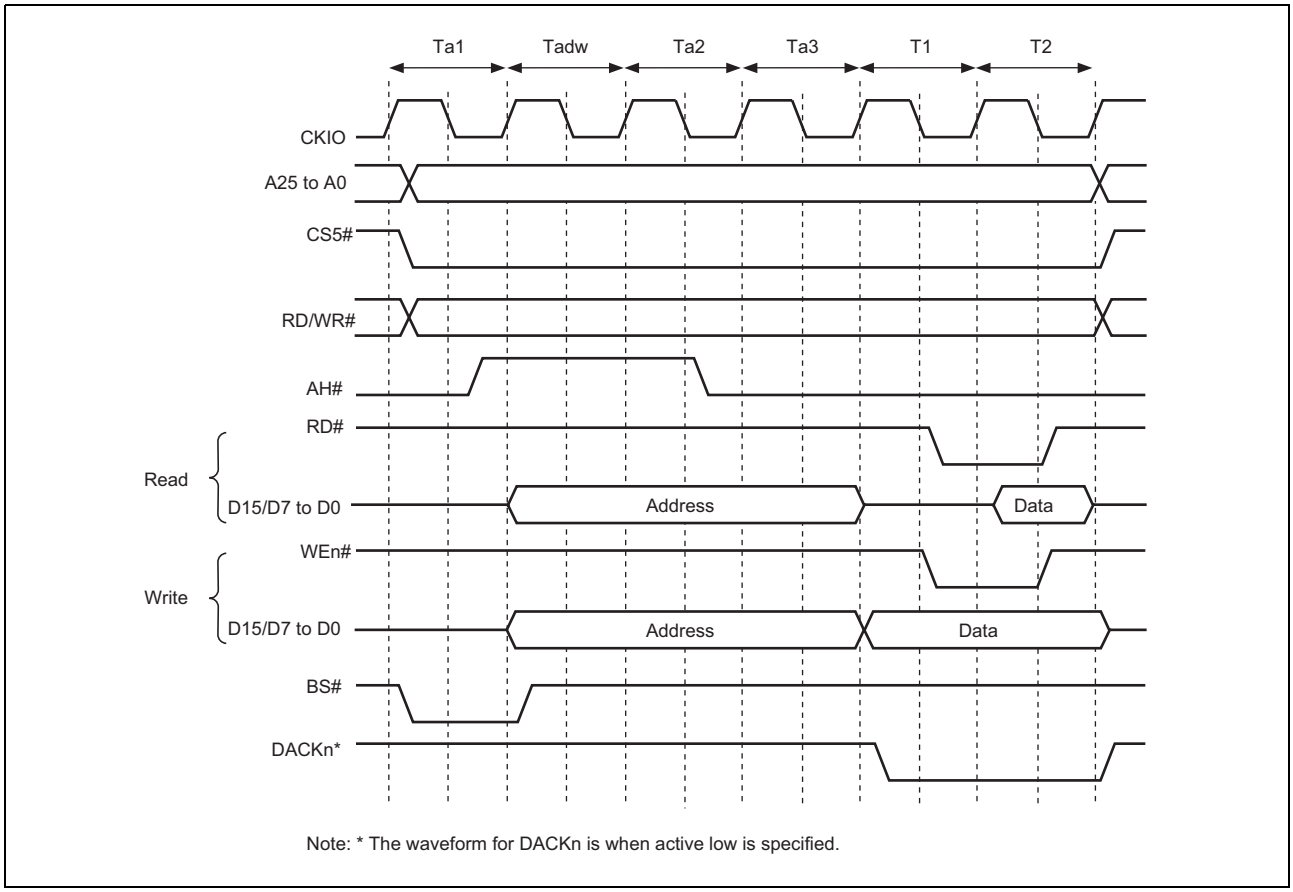


Figure 8.12 Access Timing for MPX Space (Address Cycle Wait 1, Data Cycle No Wait)

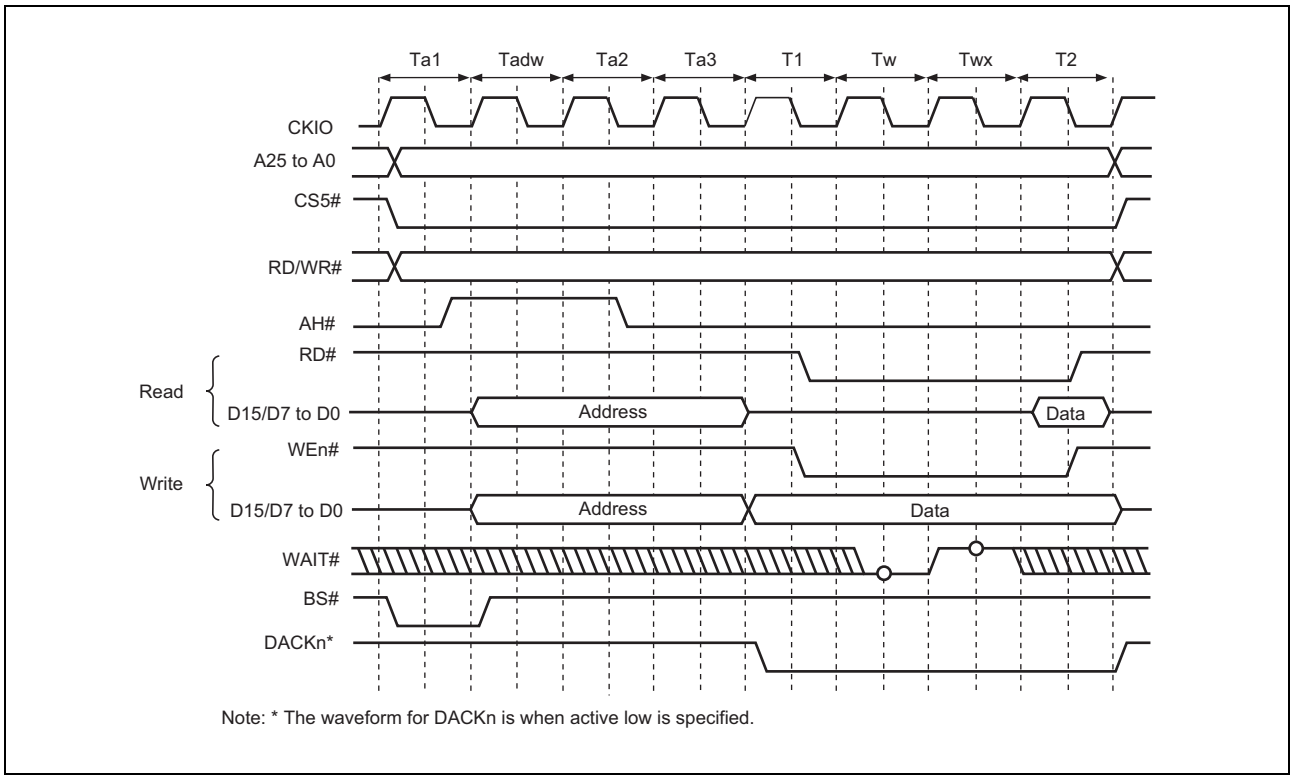


Figure 8.13 Access Timing for MPX Space (Address Cycle Access Wait 1, Data Cycle Wait 1, External Wait 1)

8.5.6 SDRAM Interface

(1) SDRAM Direct Connection

The SDRAM that can be connected to this LSI is a product that has 11/12/13 bits of row address, 8/9/10 bits of column address, 4 or less banks, and uses the A10 pin for setting precharge mode in read and write command cycles.

The control signals for direct connection of SDRAM are RAS#, CAS#, RD/WR#, DQMU, DQML, CKE, CS2#, and CS3#. All the signals other than CS2# and CS3# are common to all areas, and signals other than CKE are valid only when CS2# or CS3# is asserted. SDRAM can be connected to up to 2 spaces. The data bus width of the area that is connected to SDRAM is 16 bits only.
Burst read/single write (burst length 1) and burst read/burst write (burst length 1) are supported as the SDRAM operating mode.
Commands for SDRAM can be specified by RAS#, CAS#, RD/WR#, and specific address signals. These commands supports:

- NOP
- Auto-refresh (REF)
- Self-refresh (SELF)
- All banks pre-charge (PALL)
- Specified bank pre-charge (PRE)
- Bank active (ACTV)
- Read (READ)
- Read with pre-charge (READA)
- Write (WRIT)
- Write with pre-charge (WRITA)
- Write mode register (MRS, EMRS)

The byte to be accessed is specified by DQMU and DQML. Reading or writing is performed for a byte whose corresponding DQMx is low. For details on the relationship between DQMx and the byte to be accessed, see section 8.5.1, Access Size and Data Alignment.

Figure 8.14 shows examples of the connection of the SDRAM with the LSI.

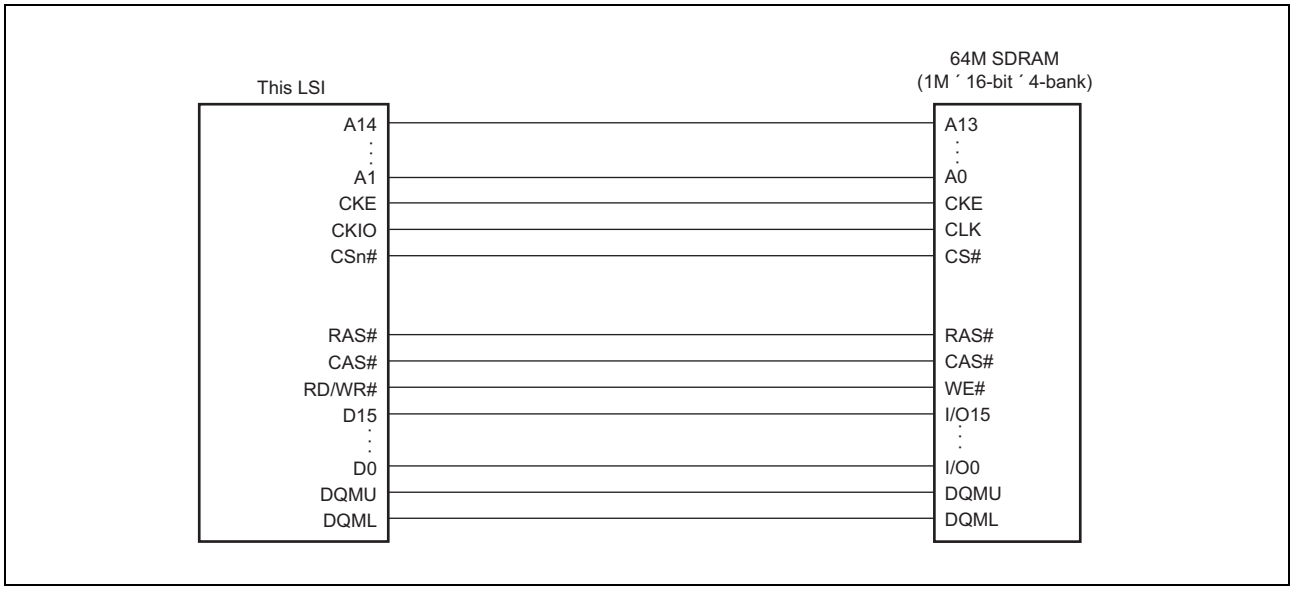


Figure 8.14 Example of 16-Bit Data Width SDRAM Connection

(2) Address Multiplexing

An address multiplexing is specified so that SDRAM can be connected without external multiplexing circuitry according to the setting of bits BSZ[1:0] in CSnBCR and bits A2ROW[1:0], A2COL[1:0], A3ROW[1:0], and A3COL[1:0] in SDCR. Table 8.7 to Table 8.9 show the relationship between the settings of bits BSZ[1:0], A2ROW[1:0], A2COL[1:0], A3ROW[1:0], and A3COL[1:0] and the bits output at the address pins. Do not specify those bits in the manner other than this table, otherwise the operation of this LSI is not guaranteed. A25 to A18 are not multiplexed and the original values of address are always output at these pins.
When the data bus width is 16 bits (BSZ1 and BSZ0 = B'10), A0 of SDRAM specifies a 16-bit address. Therefore, connect this A0 pin of SDRAM to the A1 pin of the LSI; the A1 pin of SDRAM to the A2 pin of the LSI, and so on.

Table 8.7 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (1)

Setting					Setting				
BSZ [1:0]	A2/3ROW [1:0]	A2/3COL [1:0]			BSZ [1:0]	A2/3ROW [1:0]	A2/3COL [1:0]		
10 (16 bits)	00 (11 bits)	00 (8 bits)			10 (16 bits)	01 (12 bits)	00 (8 bits)		
Output Pin of This LSI	Row Address Output Cycle	Column Address Output Cycle	SDRAM Pin	Function	Output Pin of This LSI	Row Address Output Cycle	Column Address Output Cycle	SDRAM Pin	Function
A17	A25	A17		Unused	A17	A25	A17		Unused
A16	A24	A16			A16	A24	A16		
A15	A23	A15			A15	A23	A15		
A14	A22	A14			A14	A22*2	A22*2	A13 (BA1)	Specifies bank
A13	A21	A21			A13	A21*2	A21*2	A12 (BA0)	
A12	A20*2	A20*2	A11 (BA0)	Specifies bank	A12	A20	A12	A11	Address
A11	A19	L/H*1	A10/AP	Specifies address/ precharge	A11	A19	L/H*1	A10/AP	Specifies address/ precharge
A10	A18	A10	A9	Address	A10	A18	A10	A9	Address
A9	A17	A9	A8		A9	A17	A9	A8	
A8	A16	A8	A7		A8	A16	A8	A7	
A7	A15	A7	A6		A7	A15	A7	A6	
A6	A14	A6	A5		A6	A14	A6	A5	
A5	A13	A5	A4		A5	A13	A5	A4	
A4	A12	A4	A3		A4	A12	A4	A3	
A3	A11	A3	A2		A3	A11	A3	A2	
A2	A10	A2	A1		A2	A10	A2	A1	
A1	A9	A1	A0		A1	A9	A1	A0	
A0	A8	A0		Unused	A0	A8	A0		Unused
Example of connected memory					Example of connected memory				
16-Mbit product (512 Kwords × 16 bits × 2 banks, column 8 bits product): 1					64-Mbit product (1 Mwords × 16 bits × 4 banks, column 8 bits product): 1				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.
Note 2. Bank address specification

Table 8.8 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (2)

Setting					Setting				
BSZ [1:0]	A2/3ROW [1:0]	A2/3COL [1:0]			BSZ [1:0]	A2/3ROW [1:0]	A2/3COL [1:0]		
10 (16 bits)	01 (12 bits)	01 (9 bits)			10 (16 bits)	01 (12 bits)	10 (10 bits)		
Output Pin of This LSI	Row Address Output Cycle	Column Address Output Cycle	SDRAM Pin	Function	Output Pin of This LSI	Row Address Output Cycle	Column Address Output Cycle	SDRAM Pin	Function
A17	A26	A17		Unused	A17	A27	A17		Unused
A16	A25	A16			A16	A26	A16		
A15	A24	A15			A15	A25	A15		
A14	A23*2	A23*2	A13 (BA1)	Specifies bank	A14	A24*2	A24*2	A13 (BA1)	Specifies bank
A13	A22*2	A22*2	A12 (BA0)		A13	A23*2	A23*2	A12 (BA0)	
A12	A21	A12	A11	Address	A12	A22	A12	A11	Address
A11	A20	L/H*1	A10/AP	Specifies address/ precharge	A11	A21	L/H*1	A10/AP	Specifies address/ precharge
A10	A19	A10	A9	Address	A10	A20	A10	A9	Address
A9	A18	A9	A8		A9	A19	A9	A8	
A8	A17	A8	A7		A8	A18	A8	A7	
A7	A16	A7	A6		A7	A17	A7	A6	
A6	A15	A6	A5		A6	A16	A6	A5	
A5	A14	A5	A4		A5	A15	A5	A4	
A4	A13	A4	A3		A4	A14	A4	A3	
A3	A12	A3	A2		A3	A13	A3	A2	
A2	A11	A2	A1		A2	A12	A2	A1	
A1	A10	A1	A0		A1	A11	A1	A0	
A0	A9	A0		Unused	A0	A10	A0		Unused
Example of connected memory					Example of connected memory				
128-Mbit product (2 Mwords × 16 bits × 4 banks, column 9 bits product): 1					256-Mbit product (4 Mwords × 16 bits × 4 banks, column 10 bits product): 1				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.
Note 2. Bank address specification

Table 8.9 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (3)

Setting					Setting				
BSZ [1:0]	A2/3ROW [1:0]	A2/3COL [1:0]			BSZ [1:0]	A2/3ROW [1:0]	A2/3COL [1:0]		
10 (16 bits)	10 (13 bits)	01 (9 bits)			10 (16 bits)	10 (13 bits)	10 (10 bits)		
Output Pin of This LSI	Row Address Output Cycle	Column Address Output Cycle	SDRAM Pin	Function	Output Pin of This LSI	Row Address Output Cycle	Column Address Output Cycle	SDRAM Pin	Function
A17	A26	A17		Unused	A17	A27	A17		Unused
A16	A25	A16			A16	A26	A16		
A15	A24*2	A24*2	A14 (BA1)	Specifies bank	A15	A25*2	A25*2	A14 (BA1)	Specifies bank
A14	A23*2	A23*2	A13 (BA0)		A14	A24*2	A24*2	A13 (BA0)	
A13	A22	A13	A12	Address	A13	A23	A13	A12	Address
A12	A21	A12	A11		A12	A22	A12	A11	
A11	A20	L/H*1	A10/AP	Specifies address/ precharge	A11	A21	L/H*1	A10/AP	Specifies address/ precharge
A10	A19	A10	A9	Address	A10	A20	A10	A9	Address
A9	A18	A9	A8		A9	A19	A9	A8	
A8	A17	A8	A7		A8	A18	A8	A7	
A7	A16	A7	A6		A7	A17	A7	A6	
A6	A15	A6	A5		A6	A16	A6	A5	
A5	A14	A5	A4		A5	A15	A5	A4	
A4	A13	A4	A3		A4	A14	A4	A3	
A3	A12	A3	A2		A3	A13	A3	A2	
A2	A11	A2	A1		A2	A12	A2	A1	
A1	A10	A1	A0		A1	A11	A1	A0	
A0	A9	A0		Unused	A0	A10	A0		Unused
Example of connected memory					Example of connected memory				
256-Mbit product (4 Mwords × 16 bits × 4 banks, column 9 bits product): 1					512-Mbit product (8 Mwords × 16 bits × 4 banks, column 10 bits product): 1				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.
Note 2. Bank address specification

(3) Burst Read

A burst read occurs in the following cases with this LSI.

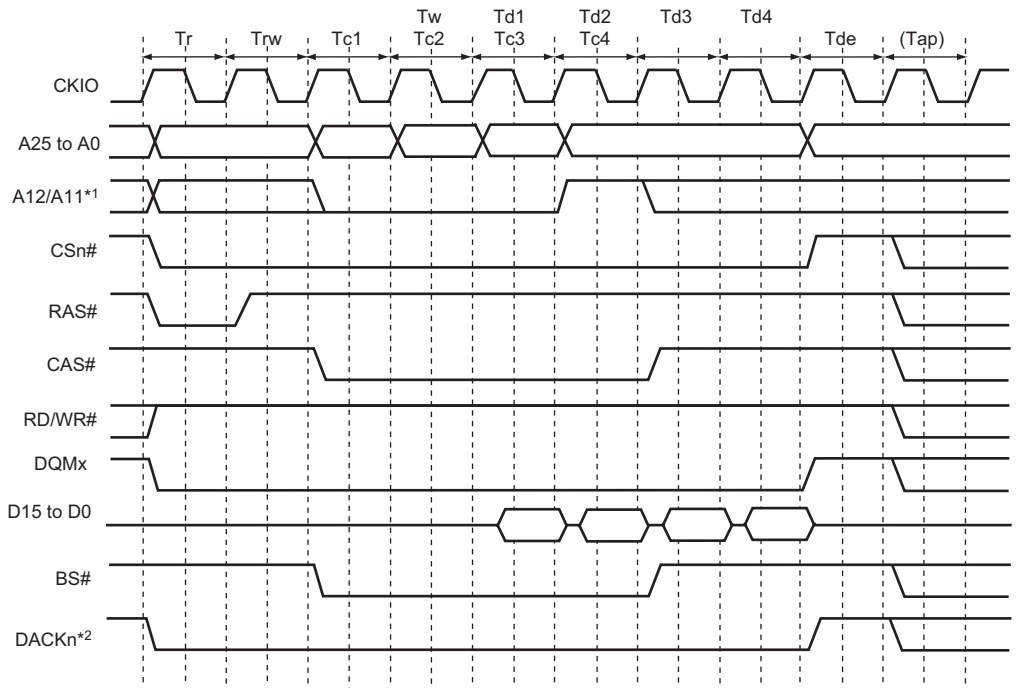
1. Access size in reading is larger than data bus width.
2. 16-, 32- or 64-byte transfer

This LSI always accesses the SDRAM with burst length 1. For example, read access of burst length 1 is performed consecutively 8 times to read 16-byte continuous data from the SDRAM that is connected to a 16-bit data bus. This access is called the burst read with the burst number 8. Table 8.10 shows the relationship between the access size and the number of bursts.

Table 8.10 Relationship between Access Size and Number of Bursts

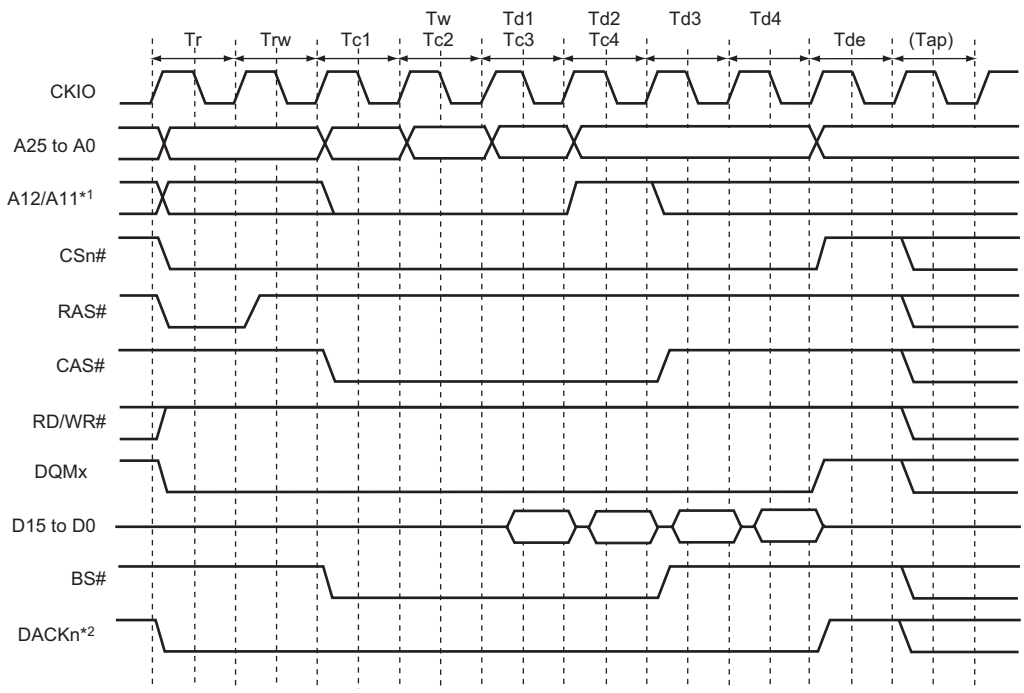
Bus Width	Access Size	Number of Bursts
16 bits	8 bits	1
	16 bits	1
	32 bits	2
	16 bytes	8
	32 bytes	16
	64 bytes	32

Figure 8.15 and Figure 8.16 show timing charts in burst read. In burst read, an ACTV command is output in the Tr cycle, the READ command is issued in the Tc1, Tc2, and Tc3 cycles, the READA command is issued in the Tc4 cycle, and the read data is received at the rising edge of the external clock (CKIO) in the Td1 to Td4 cycles. The Tap cycle is used to wait for the completion of an auto-precharge induced by the READA command in the SDRAM. In the Tap cycle, a new command will not be issued to the same bank. However, access to another CS space or another bank in the same SDRAM space is enabled. The number of Tap cycles is specified by the WTRP1 and WTRP0 bits in CS3WCR.
In this LSI, wait cycles can be inserted by specifying each bit in CS3WCR to connect the SDRAM in variable frequencies. Figure 8.15 shows an example in which wait cycles are inserted. The number of cycles from the Tr cycle where the ACTV command is output to the Tc1 cycle where the READ command is output can be specified using the WTRCD1 and WTRCD0 bits in CS3WCR. If the WTRCD1 and WTRCD0 bits specify one cycles or more, a Trw cycle where the NOP command is issued is inserted between the Tr cycle and Tc1 cycle. The number of cycles from the Tc1 cycle where the READ command is output to the Td1 cycle where the read data is latched can be specified for the CS2 and CS3 spaces independently, using the A2CL1 and A2CL0 bits in CS2WCR or the A3CL1 and A3CL0 bits in CS3WCR. The number of cycles from Tc1 to Td1 corresponds to the SDRAM CAS latency. The CAS latency for the SDRAM is normally defined as up to three cycles. However, the CAS latency in this LSI can be specified as 1 to 4 cycles. This CAS latency can be achieved by connecting a latch circuit between this LSI and the SDRAM.
A Tde cycle is an idle cycle required to transfer the read data into this LSI and occurs once for every burst read or every single read.



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 8.15 Burst Read Basic Timing (CAS Latency 1, Auto Pre-Charge)



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 8.16 Burst Read Wait Specification Timing (CAS Latency 2, WTRCD[1:0] = 1 Cycle, Auto Pre-Charge)

(4) Single Read

A read access ends in one cycle when the data bus width is larger than or equal to the access size. As the SDRAM is set to the burst read with the burst length 1, only the required data is output. A read access that ends in one cycle is called single read.

Figure 8.17 shows the single read basic timing.

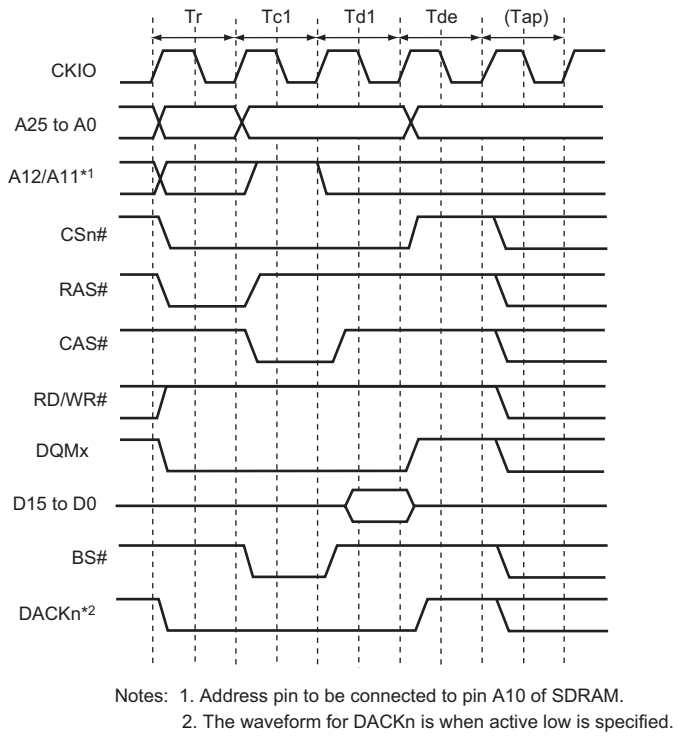


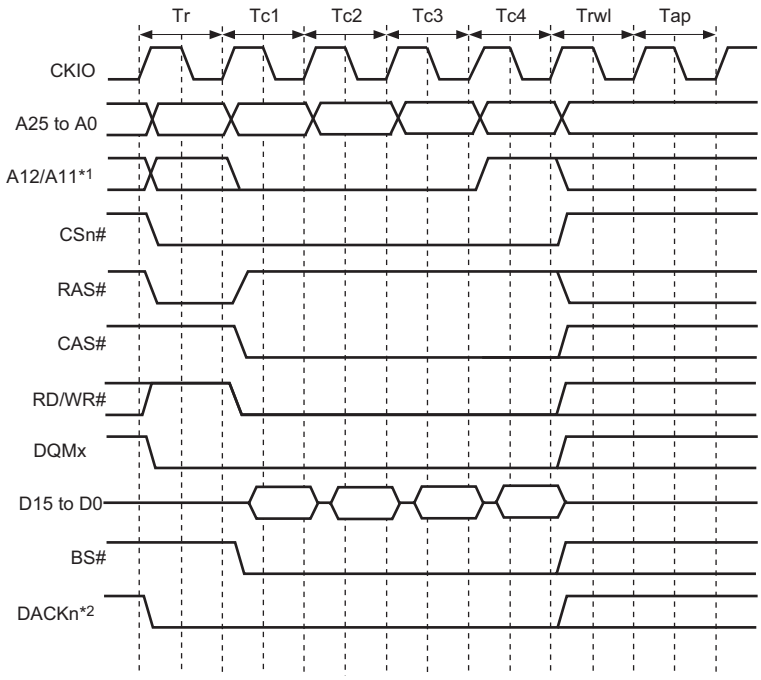
Figure 8.17 Basic Timing for Single Read (CAS Latency 1, Auto Pre-Charge)

(5) Burst Write

A burst write occurs in the following cases in this LSI.

1. Access size in writing is larger than data bus width.
2. 16-, 32- or 64-byte transfer

This LSI always accesses SDRAM with burst length 1. For example, write access of burst length 1 is performed continuously 8 times to write 16-byte continuous data to the SDRAM that is connected to a 16-bit data bus. This access is called burst write with the burst number 8. The relationship between the access size and the number of bursts is shown in Table 8.10. Figure 8.18 shows a timing chart for burst writes. In burst write, an ACTV command is output in the Tr cycle, the WRIT command is issued in the Tc1, Tc2, and Tc3 cycles, and the WRITA command is issued to execute an auto-precharge in the Tc4 cycle. In the write cycle, the write data is output simultaneously with the write command. After the write command with the auto-precharge is output, the Trw1 cycle that waits for the auto-precharge initiation is followed by the Tap cycle that waits for completion of the auto-precharge induced by the WRITA command in the SDRAM. Between the Trwl and the Tap cycle, a new command will not be issued to the same bank. However, access to another CS space or another bank in the same SDRAM space is enabled. The number of Trw1 cycles is specified by the TRWL1 and TRWL0 bits in CS3WCR. The number of Tap cycles is specified by the WTRP1 and WTRP0 bits in CS3WCR.

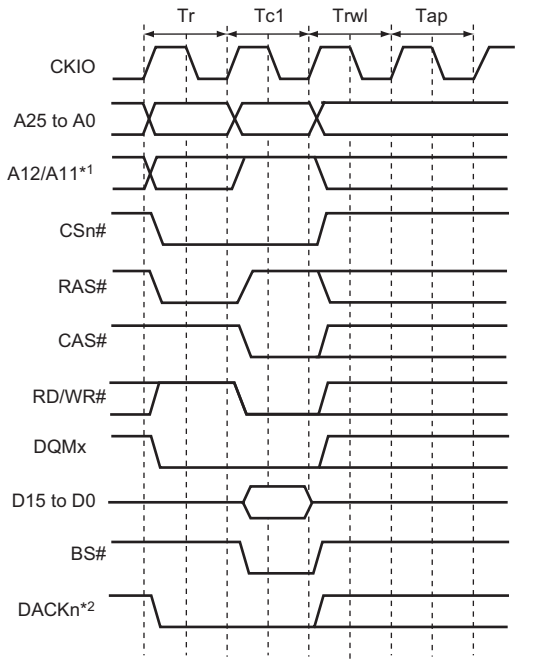


Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 8.18 Basic Timing for Burst Write (Auto Pre-Charge)

(6) Single Write

A write access ends in one cycle when the data bus width is larger than or equal to access size. As a single write or burst write with burst length 1 is set in SDRAM, only the required data is output. The write access that ends in one cycle is called single write. Figure 8.19 shows the single write basic timing.



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 8.19 Single Write Basic Timing (Auto-Precharge)

(7) Bank Active

The SDRAM bank function can be used to support high-speed access to the same row address. When the BACTV bit in SDCR is 1, access is performed using commands without auto-precharge (READ or WRIT). This function is called bank-active function. This function is valid only for area 3. When area 3 is set to bank-active mode, area 2 should be set to normal space or SRAM with byte selection. When areas 2 and 3 are both set to SDRAM, auto precharge mode must be set.

When the bank-active function is used, precharging is not performed when the access ends. When accessing the same row address in the same bank, it is possible to issue the READ or WRIT command immediately, without issuing an ACTV command. As SDRAM is internally divided into several banks, it is possible to activate one row address in each bank. If the next access is to a different row address, a PRE command is first issued to precharge the relevant bank, then when precharging is completed, the access is performed by issuing an ACTV command followed by a READ or WRIT command. If this is followed by an access to a different row address, the access time will be longer because of the precharging performed after the access request is issued. The number of cycles between issuance of the PRE command and the ACTV command is determined by the WTRP1 and WTPR0 bits in CS3WCR.

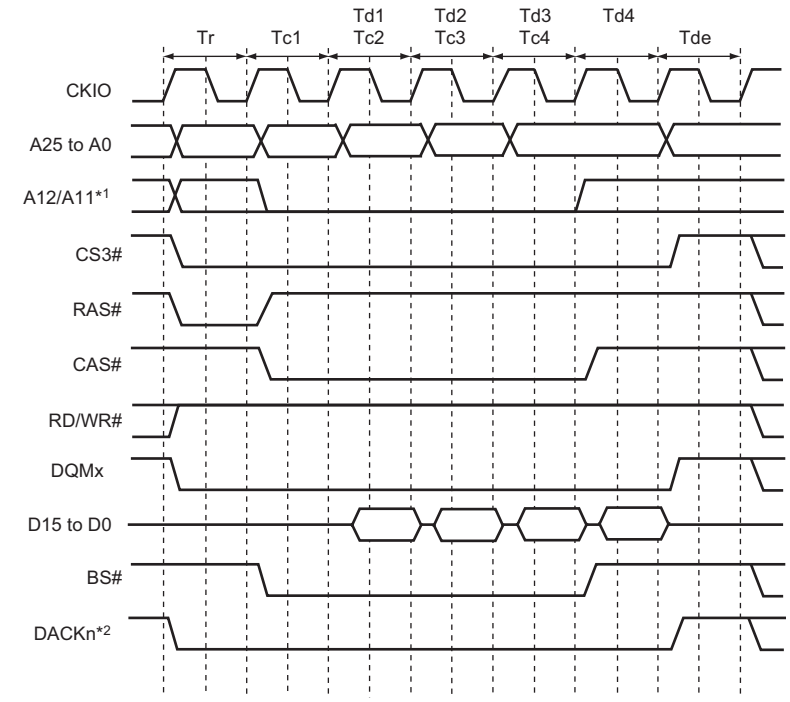
In a write, when an auto-precharge is performed, a command cannot be issued to the same bank for a period of Trwl + Tap cycles after issuance of the WRITA command. When bank active mode is used, READ or WRIT commands can be issued successively if the row address is the same. The number of cycles can thus be reduced by Trwl + Tap cycles for each write.

There is a limit on tRAS, the time for placing each bank in the active state. If there is no guarantee that there will not be a cache hit and another row address will be accessed within the period in which this value is maintained by program execution, it is necessary to set auto-refresh and set the refresh cycle to no more than the maximum value of tRAS.

A burst read cycle without auto-precharge is shown in Figure 8.20, a burst read cycle for the same row address in Figure 8.21, and a burst read cycle for different row addresses in Figure 8.22. Similarly, a single write cycle without auto-precharge is shown in Figure 8.23, a single write cycle for the same row address in Figure 8.24, and a single write cycle for different row addresses in Figure 8.25.

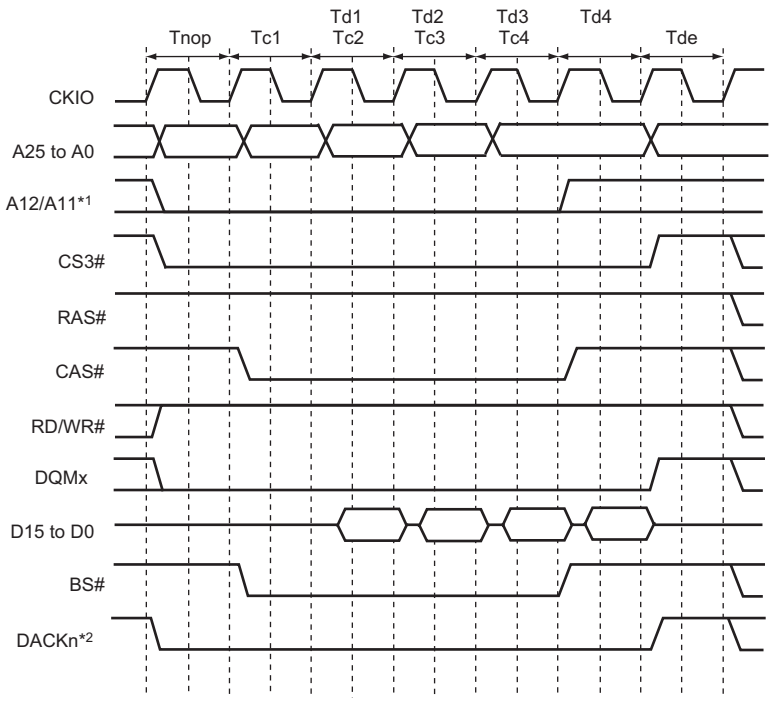
In Figure 8.21, a Tnop cycle in which no operation is performed is inserted before the Tc cycle that issues the READ command. The Tnop cycle is inserted to acquire two cycles of CAS latency for the DQMx signal that specifies the read byte in the data read from the SDRAM. If the CAS latency is specified as two cycles or more, the Tnop cycle is not inserted because the two cycles of latency can be acquired even if the DQMx signal is asserted after the Tc cycle.

When bank active mode is set, if only access cycles to the respective banks in the area 3 space are considered, as long as access cycles to the same row address continue, the operation starts with the cycle in Figure 8.20 or Figure 8.23, followed by repetition of the cycle in Figure 8.21 or Figure 8.24. An access to a different area during this time has no effect. If there is an access to a different row address in the bank active state, the bus cycle in Figure 8.22 or Figure 8.25 is executed instead of that in Figure 8.21 or Figure 8.24. In bank active mode, too, all banks become inactive after a refresh cycle.



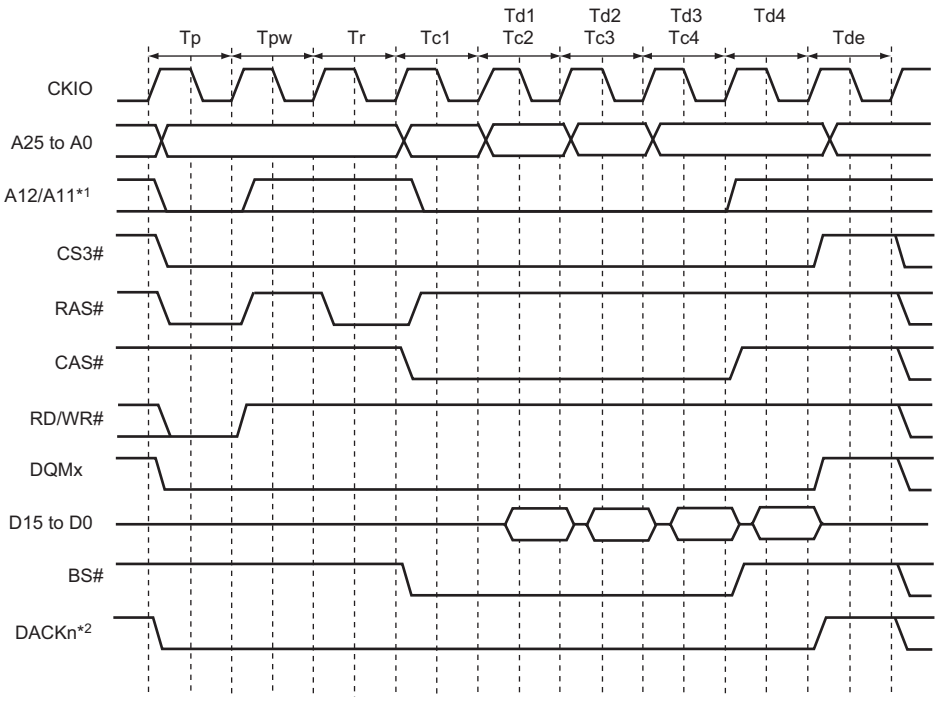
Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 8.20 Burst Read Timing (Bank Active, Different Bank, CAS Latency 1)



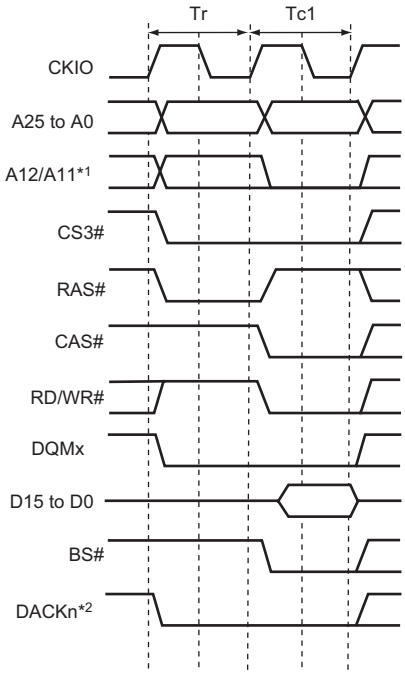
Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 8.21 Burst Read Timing (Bank Active, Same Row Addresses in the Same Bank, CAS Latency 1)



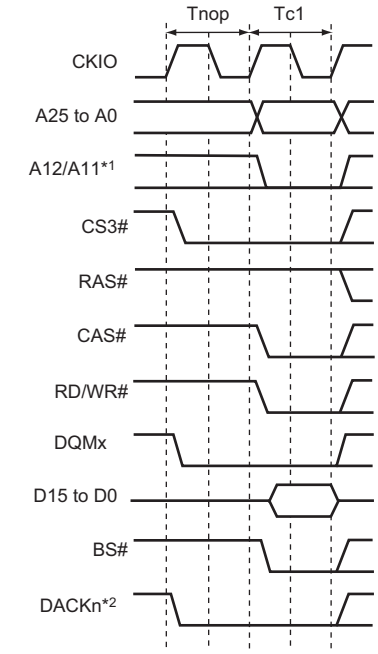
Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 8.22 Burst Read Timing (Bank Active, Different Row Addresses in the Same Bank, CAS Latency 1)



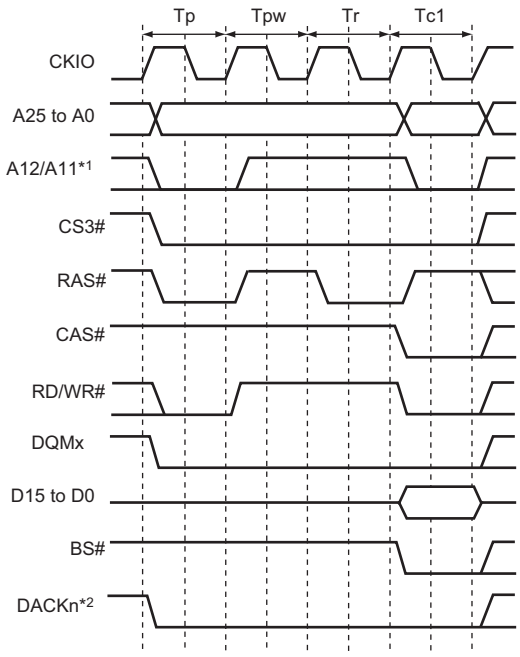
Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 8.23 Single Write Timing (Bank Active, Different Bank)



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 8.24 Single Write Timing (Bank Active, Same Row Addresses in the Same Bank)



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 8.25 Single Write Timing (Bank Active, Different Row Addresses in the Same Bank)

(8) Refreshing

This module has a function for controlling SDRAM refreshing. Auto-refreshing can be performed by clearing the RMODE bit to 0 and setting the RFSH bit to 1 in SDCR. A continuous refreshing can be performed by setting the RRC2 to RRC0 bits in RTCSR. If SDRAM is not accessed for a long period, self-refresh mode, in which the power consumption for data retention is low, can be activated by setting both the RMODE bit and the RFSH bit to 1.

(a) Auto-refreshing

Refreshing is performed at intervals determined by the input clock selected by bits CKS2 to CKS0 in RTCSR, and the value set by in RTCOR. The value of bits CKS2 to CKS0 in RTCOR should be set so as to satisfy the refresh interval stipulation for the SDRAM used. First make the settings for RTCOR, RTCNT, and the RMODE and RFSH bits in SDCR, and then make the CKS2 to CKS0 and RRC2 to RRC0 settings. When the clock is selected by bits CKS2 to CKS0, RTCNT starts counting up from the value at that time. The RTCNT value is constantly compared with the RTCOR value, and if the two values are the same, a refresh request is generated and an auto-refresh is performed for the number of times specified by the RRC2 to RRC0. At the same time, RTCNT is cleared to zero and the count-up is restarted.
Figure 8.26 shows the auto-refresh cycle timing. After starting the auto refreshing, PALL command is issued in the Tp cycle to make all the banks to pre-charged state from active state when some bank is being pre-charged. Then REF command is issued in the Trr cycle after inserting idle cycles of which number is specified by the WTRP1 and WTRP0 bits in CS3WCR. A new command is not issued for the duration of the number of cycles specified by the WTRC1 and WTRC0 bits in CS3WCR after the Trr cycle. The WTRC1 and WTRC0 bits must be set so as to satisfy the SDRAM refreshing cycle time stipulation (tRC). An idle cycle is inserted between the Tp cycle and Trr cycle when the setting value of the WTRP1 and WTRP0 bits in CS3WCR is longer than or equal to 1 cycle.

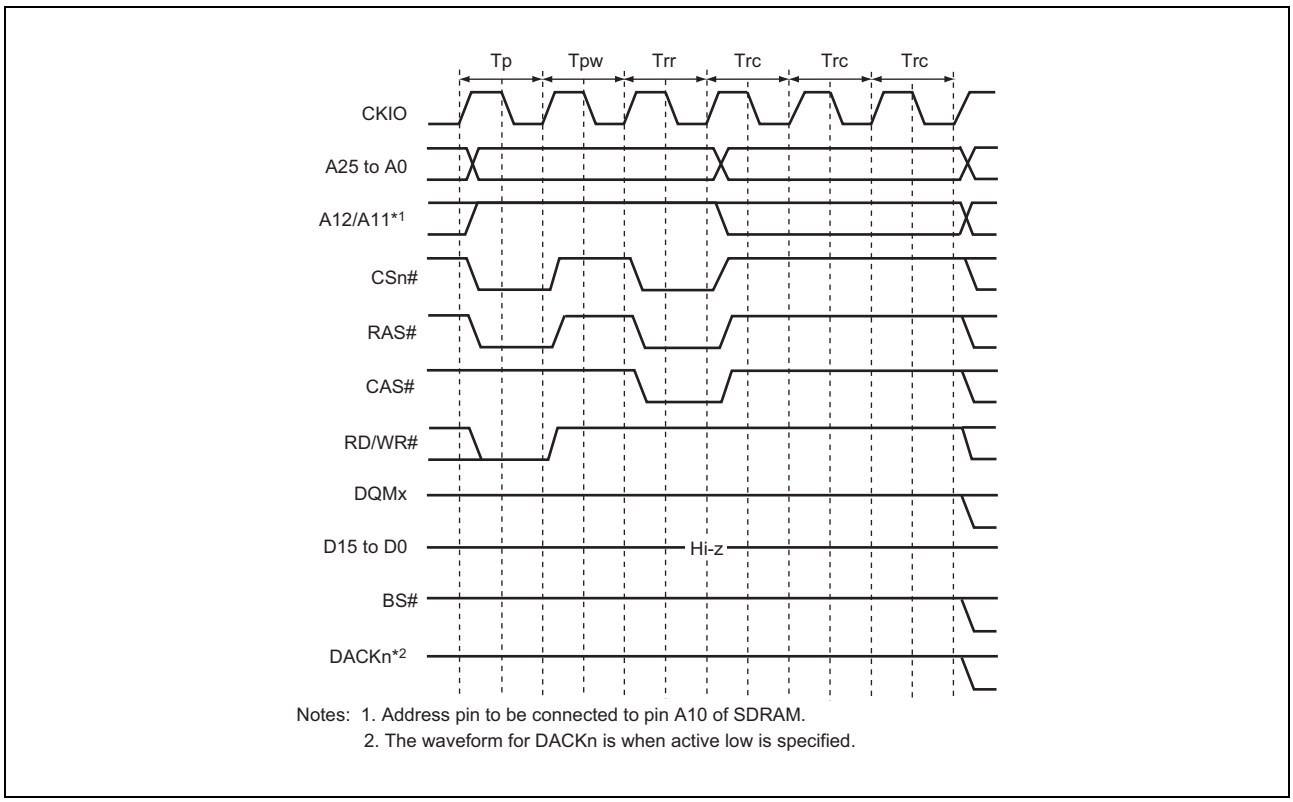
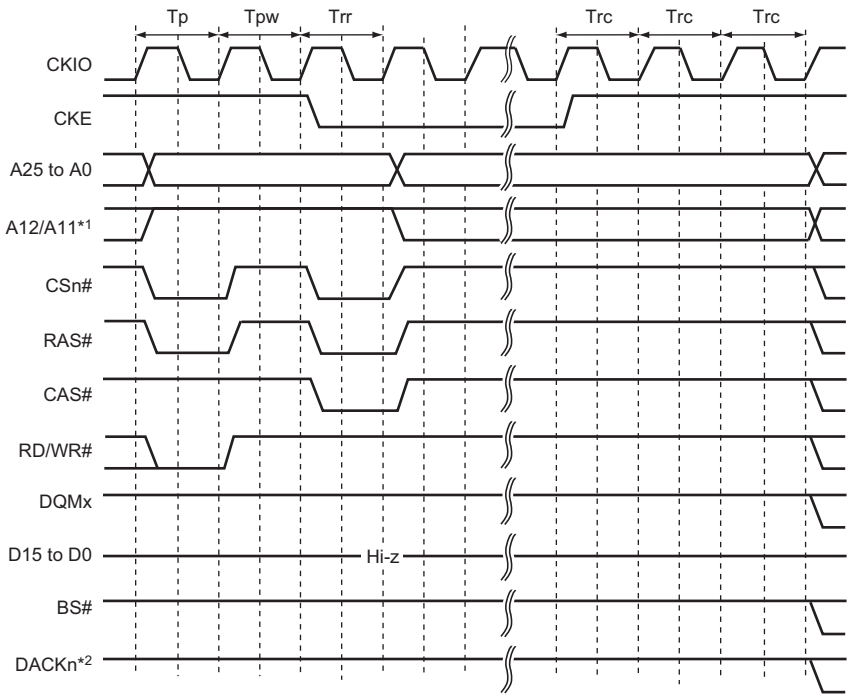


Figure 8.26 Auto-Refresh Timing

(b) Self-refreshing

Self-refresh mode is a kind of standby mode, in which the refresh timing and refresh addresses are generated within the SDRAM. Self-refreshing is activated by setting both the RMODE bit and the RFSH bit in SDCR to 1. After starting the self-refreshing, PALL command is issued in Tp cycle after the completion of the pre-charging bank. A SELF command is then issued after inserting idle cycles of which number is specified by the WTRP1 and WTRP0 bits in CS3WSR.
SDRAM cannot be accessed while in the self-refresh state. Self-refresh mode is cleared by clearing the RMODE bit to 0. After self-refresh mode has been cleared, command issuance is disabled for the number of cycles specified by the WTRC1 and WTRC0 bits in CS3WCR.
Self-refresh timing is shown in Figure 8.27. Settings must be made so that self-refresh clearing and data retention are performed correctly, and auto-refreshing is performed at the correct intervals. When self-refreshing is activated from the state in which auto-refreshing is set, auto-refreshing is restarted if the RFSH bit is set to 1 and the RMODE bit is cleared to 0 when self-refresh mode is cleared. If the transition from clearing of self-refresh mode to the start of auto-refreshing takes time, this time should be taken into consideration when setting the initial value of RTCNT. Making the RTCNT value 1 less than the RTCOR value will enable refreshing to be started immediately.
After self-refreshing has been set, the self-refresh state continues even if the chip standby state is entered using the LSI standby function, and is maintained even after recovery from standby mode due to an interrupt. Note that the necessary signals such as CKE must be driven even in standby state by setting the HIZCNT bit in CMNCR to 1.
In case of a power-on reset, the bus state controller's registers are initialized, and therefore the self-refresh state is cleared.



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 8.27 Self-Refresh Timing

(9) Relationship between Refresh Requests and Bus Cycles

If a refresh request occurs during bus cycle execution, the refresh cycle must wait for the bus cycle to be completed. If a new refresh request occurs while waiting for the previous refresh request, the previous refresh request is deleted. To refresh correctly, a bus cycle longer than the refresh interval must be prevented from occurring.

(10) Power-Down Mode

If the PDOWN bit in SDCR is set to 1, the SDRAM is placed in power-down mode by bringing the CKE signal to the low level in the non-access cycle. This power-down mode can effectively lower the power consumption in the non-access cycle. However, please note that if an access occurs in power-down mode, a cycle of overhead occurs because a cycle is needed to assert the CKE in order to cancel the power-down mode.
Figure 8.28 shows the access timing in power-down mode.

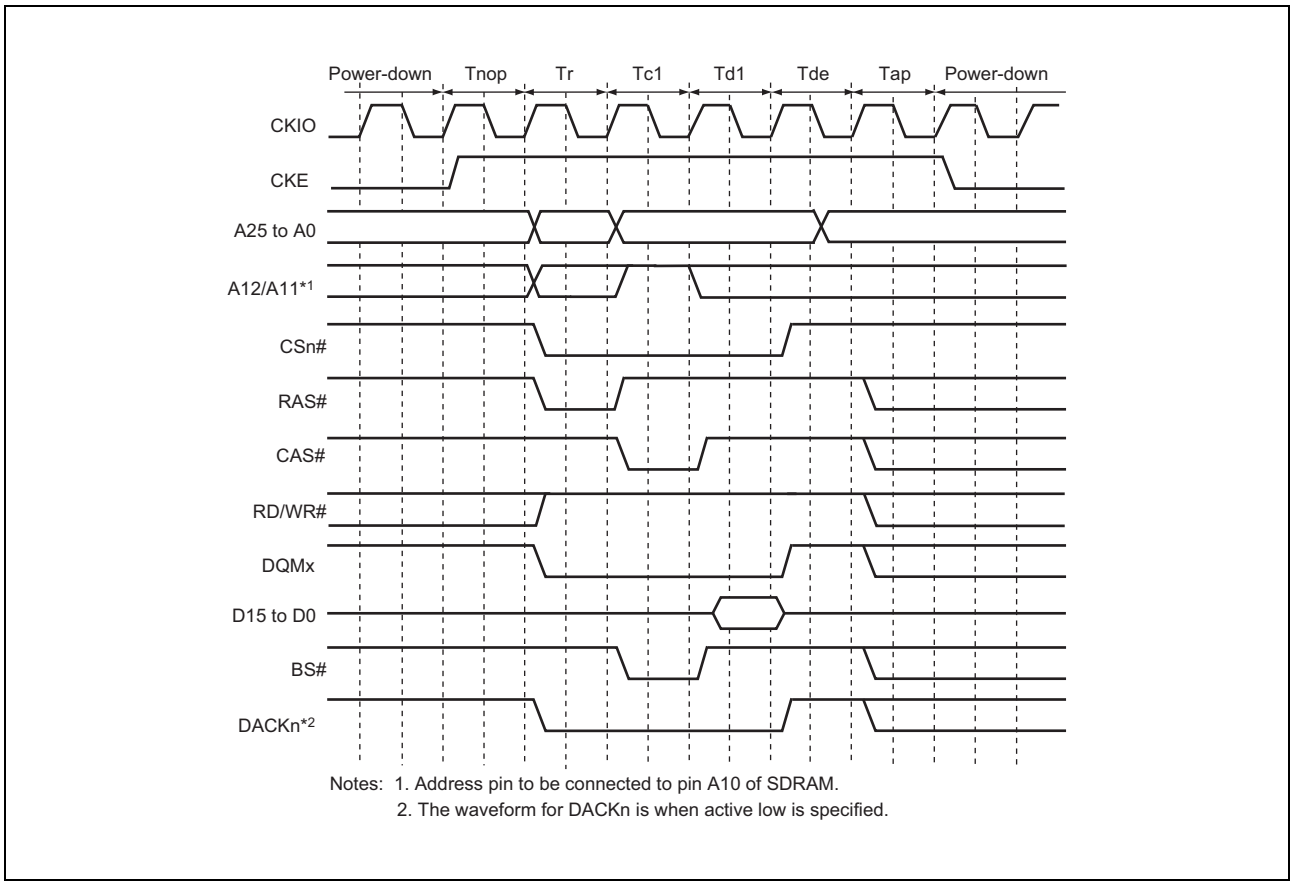


Figure 8.28 Power-Down Mode Access Timing

(11) Power-On Sequence

In order to use SDRAM, mode setting must first be made for SDRAM after the pose interval specified for the SDRAM to be used after powering on. The pose interval should be obtained by a power-on reset generating circuit or software.
To perform SDRAM initialization correctly, the registers of this module must first be set, followed by a write to the SDRAM mode register. In SDRAM mode register setting, the address signal value at that time is latched by a combination of the CSn#, RAS#, CAS#, and RD/WR# signals. If the value to be set is X, the bus state controller provides for value X to be written to the SDRAM mode register by performing a 16-bit write to address 1F001000 + X for area 2 SDRAM, and to address 1F002000 + X for area 3 SDRAM. In this operation the data is ignored, but the mode write is performed as a byte-size access. To set burst read/single write or burst read/burst write (CAS latency 2 to 3, wrap type = sequential, and burst length 1) supported by the LSI, arbitrary data is written in 16 bits to the access addresses shown in Table 8.11. In this time 0 is output at the external address pins of A12 or later.

Table 8.11 Access Address in SDRAM Mode Register Write

- Setting for Area 2

Burst read/single write (burst length 1):

Data Bus Width	CAS Latency	Access Address	External Address Pin
16 bits	2	H'1F001440	H'0000440
	3	H'1F001460	H'0000460

Burst read/burst write (burst length 1):

Data Bus Width	CAS Latency	Access Address	External Address Pin
16 bits	2	H'1F001040	H'0000040
	3	H'1F001060	H'0000060

- Setting for Area 3

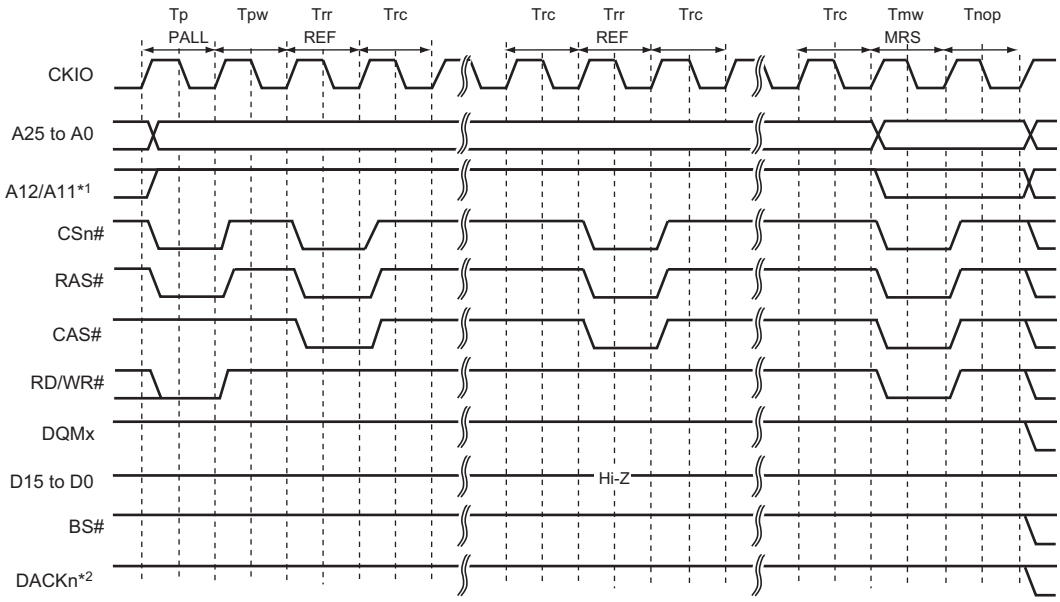
Burst read/single write (burst length 1):

Data Bus Width	CAS Latency	Access Address	External Address Pin
16 bits	2	H'1F002440	H'0000440
	3	H'1F002460	H'0000460

Burst read/burst write (burst length 1):

Data Bus Width	CAS Latency	Access Address	External Address Pin
16 bits	2	H'1F002040	H'0000040
	3	H'1F002060	H'0000060

Mode register setting timing is shown in Figure 8.29. A PALL command (all bank pre-charge command) is firstly issued. A REF command (auto refresh command) is then issued 8 times. An MRS command (mode register write command) is finally issued. Idle cycles, of which number is specified by the WTRP1 and WTRP0 bits in CS3WCR, are inserted between the PALL and the first REF. Idle cycles, of which number is specified by the WTRC1 and WTRC0 bits in CS3WCR, are inserted between REF and REF, and between the 8th REF and MRS. One or more idle cycles are inserted between the MRS and a command to be issued next.
It is necessary to keep idle time of certain cycles for SDRAM before issuing PALL command after power-on. Refer to the manual of the SDRAM for the idle time to be needed. When the pulse width of the reset signal is longer than the idle time, mode register setting can be started immediately after the reset, but care should be taken when the pulse width of the reset signal is shorter than the idle time.



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

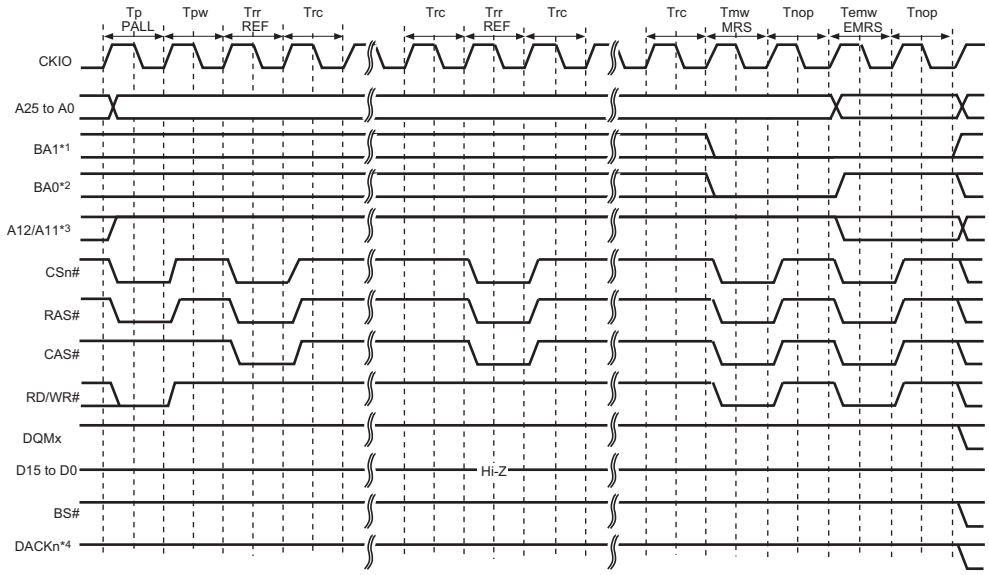
Figure 8.29 SDRAM Mode Write Timing (Based on JEDEC)

(12) Low-Power SDRAM

The low-power SDRAM can be accessed using the same protocol as the normal SDRAM.
The differences between the low-power SDRAM and normal SDRAM are that partial refresh takes place that puts only a part of the SDRAM in the self-refresh state during the self-refresh function, and that power consumption is low during refresh under user conditions such as the operating temperature. The partial refresh is effective in systems in which the data in a work area other than the specific area can be lost without severe repercussions. For details, please refer to the Data Sheet for the low-power SDRAM to be used.
The low-power SDRAM supports the extension mode register in addition to the mode registers as the normal SDRAM. This LSI supports issuing of the extension mode register write command (EMRS).
The EMRS command is issued according to the conditions specified in table below. For example, if data H'0YYYYYYY is written to address H'1F002XX0 in 32 bits, the commands are issued to the CS3 space in the following sequence: PALL -> REF × 8 -> MRS -> EMRS. In this case, the MRS and EMRS issue addresses are H'0000XX0 and H'YYYYYYY, respectively. If data H'1YYYYYYY is written to address H'1F002XX0 in 32 bits, the commands are issued to the CS3 space in the following sequence: PALL -> MRS -> EMRS.

Table 8.12 Output Addresses when EMRS Command Is Issued

Command to be Issued	Access Address	Access Data	Write Access Size	MRS Command Issue Address	EMRS Command Issue Address
CS2 MRS	H'1F001XX0	H'********	16 bits	H'0000XX0	—
CS3 MRS	H'1F002XX0	H'********	16 bits	H'0000XX0	—
CS2 MRS + EMRS (with refresh)	H'1F001XX0	H'0YYYYYYY	32 bits	H'0000XX0	H'YYYYYYY
CS3 MRS + EMRS (with refresh)	H'1F002XX0	H'0YYYYYYY	32 bits	H'0000XX0	H'YYYYYYY
CS2 MRS + EMRS (without refresh)	H'1F001XX0	H'1YYYYYYY	32 bits	H'0000XX0	H'YYYYYYY
CS3 MRS + EMRS (without refresh)	H'1F002XX0	H'1YYYYYYY	32 bits	H'0000XX0	H'YYYYYYY

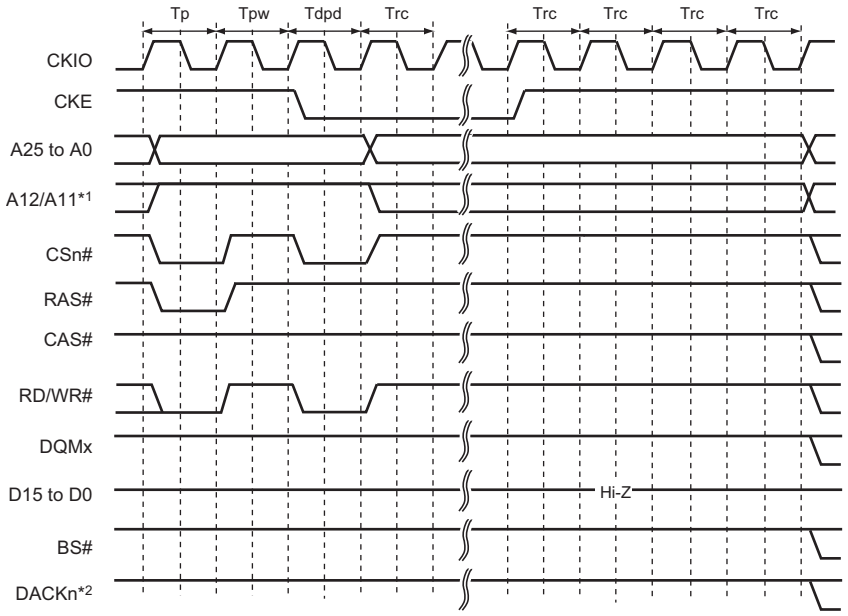


Notes: 1. Address pin to be connected to pin BA1 of SDRAM.
2. Address pin to be connected to pin BA0 of SDRAM.
3. Address pin to be connected to pin A10 of SDRAM.
4. The waveform for DACKn is when active low is specified.

Figure 8.30 EMRS Command Issue Timing

- Deep power-down mode
  The low-power SDRAM supports the deep power-down mode as a low-power consumption mode. In the partial self-refresh function, self-refresh is performed on a specific area. In the deep power-down mode, self-refresh will not be performed on any memory area. This mode is effective in systems where all of the system memory areas are used as work areas.

If the RMODE bit in the SDCR is set to 1 while the DEEP and RFSH bits in the SDCR are set to 1, the low-power SDRAM enters the deep power-down mode. If the RMODE bit is cleared to 0, the CKE signal is pulled high to cancel the deep power-down mode. Before executing an access after returning from the deep power-down mode, the power-up sequence must be re-executed.



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 8.31 Deep Power-Down Mode Transition Timing

8.5.7 Burst ROM (Clocked Asynchronous) Interface

The burst ROM (clocked asynchronous) interface is used to access a memory with a high-speed read function using a method of address switching called the burst mode or page mode. In a burst ROM (clocked asynchronous) interface, basically the same access as the normal space is performed, but the 2nd and subsequent access cycles are performed only by changing the address, without negating the RD# signal at the end of the 1st cycle. In the 2nd and subsequent access cycles, addresses are changed at the falling edge of the CKIO.

For the 1st access cycle, the number of wait cycles specified by the W3 to W0 bits in CSnWCR is inserted. For the 2nd and subsequent access cycles, the number of wait cycles specified by the BW1 and BW0 bits in CSnWCR is inserted.

In the access to the burst ROM (clocked asynchronous), the BS# signal is asserted only to the first access cycle. An external wait input is valid only to the first access cycle.

In the single access or write access that does not perform the burst operation in the burst ROM (clocked asynchronous) interface, access timing is same as a normal space.

Table 8.13 lists a relationship between bus width, access size, and the number of bursts. Figure 8.32 shows a timing chart.

Table 8.13 Relationship between Bus Width, Access Size, and Number of Bursts

Bus Width	Access Size	CSnWCR. BST[1:0] Bits	Number of Bursts	Access Count
8 bits	8 bits	Not affected	1	1
	16 bits	Not affected	2	1
	32 bits	Not affected	4	1
	16 bytes	00	16	1
		01	4	4
	32 bytes	00	16	2
		01	4	8
	64 bytes	00	16	4
		01	4	16
16 bits	8 bits	Not affected	1	1
	16 bits	Not affected	1	1
	32 bits	Not affected	2	1
	16 bytes	00	8	1
		01	2	4
		10*1	4	2
			2, 4, 2	3
	32 bytes	00	8	2
		01	2	8
		10*1	4	4
			2, 4, 2	6
	64 bytes	00	8	4
		01	2	16
		10*1	4	8
			2, 4, 2	12

Note 1. When the bus width is 16 bits, the access size is 16 bytes or more, and the BST[1:0] bits in CSnWCR are 10, the number of bursts and access count depend on the access start address. At address H'xxx0 or H'xxx8, 4-4 burst access is performed. At address H'xxx4 or H'xxxC, 2-4-2 burst access is performed.

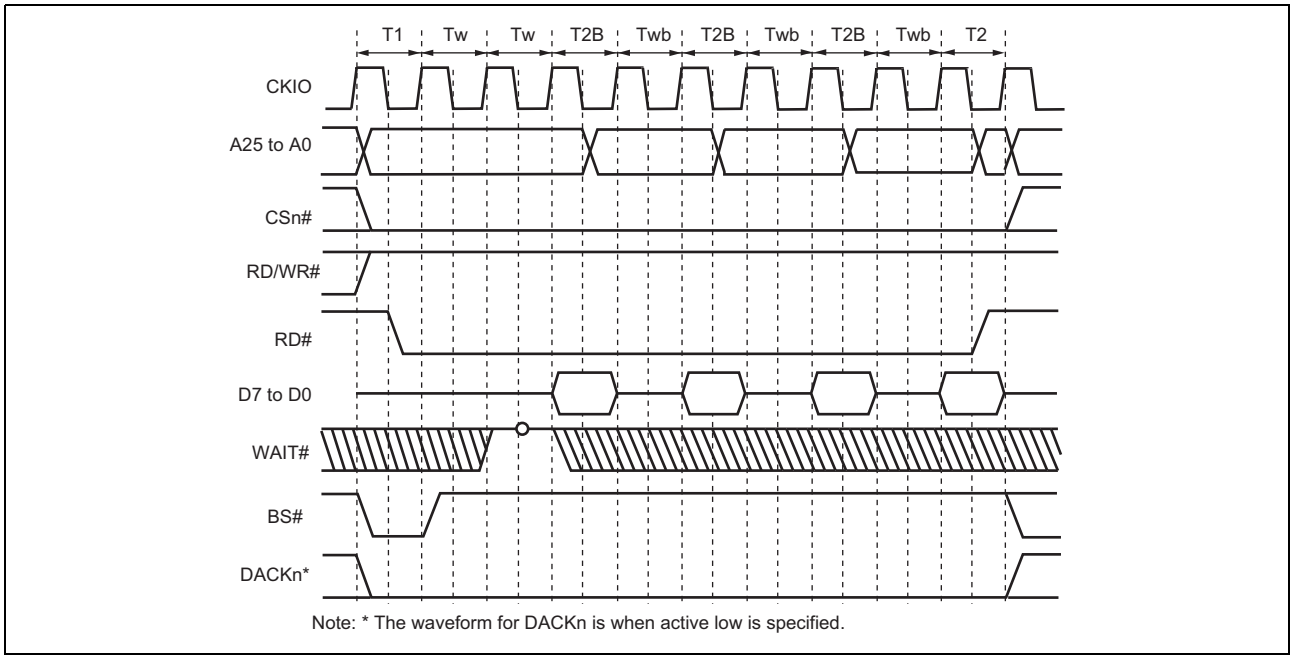


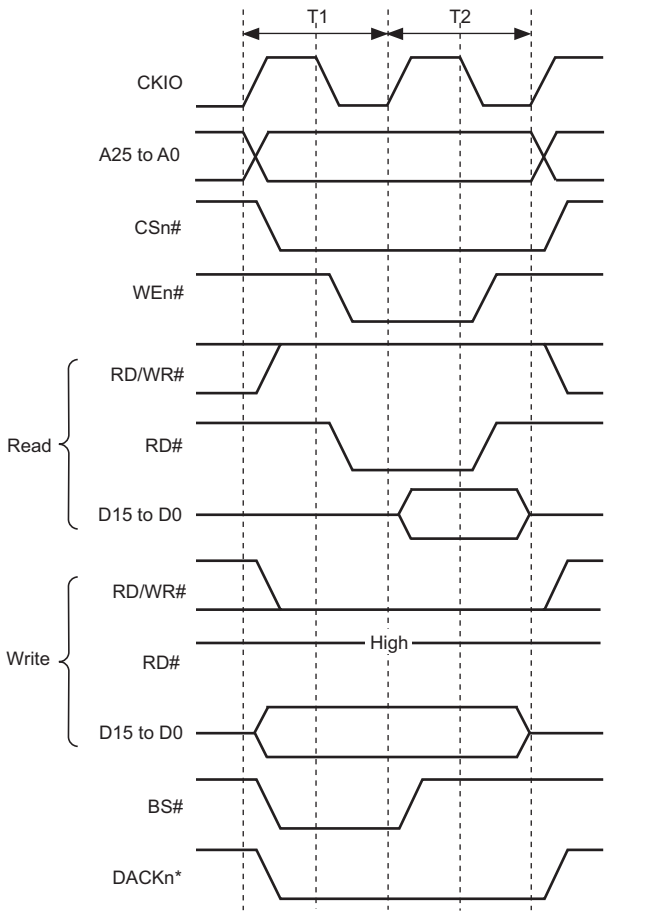
Figure 8.32 Burst ROM Access Timing (Clocked Asynchronous) (Bus Width =8 Bits, 32 Byte Transfer (Number of Burst 4), Wait Cycles Inserted in First Access = 2, Wait Cycles Inserted in Second and Subsequent Access Cycles = 1)

8.5.8 SRAM Interface with Byte Selection

The SRAM interface with byte selection is a memory interface that outputs the byte selection signal (WEn#) in both read and write bus cycles. This interface has 16-bit data pins and accesses SRAMs having upper and lower byte selection pins, such as UB# and LB#.

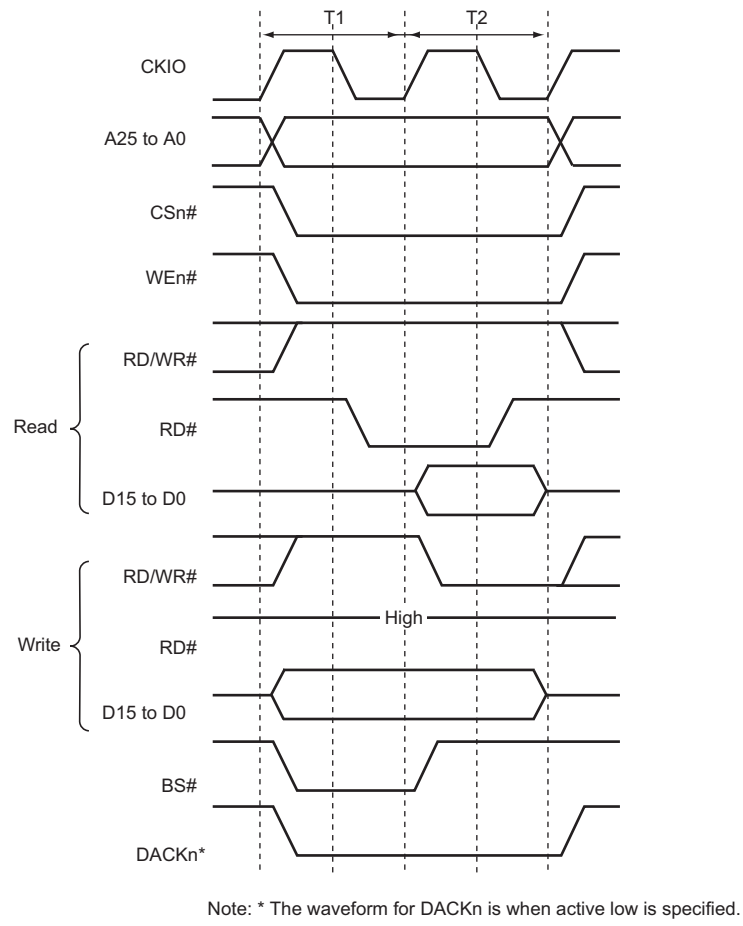
When the BAS bit in CSnWCR is cleared to 0 (initial value), the write access timing of the SRAM interface with byte selection is the same as that for the normal space interface. While in read access of a byte-selection SRAM interface, the byte-selection signal is output from the WEn# pin, which is different from that for the normal space interface. The basic access timing is shown in Figure 8.33. In write access, data is written to the memory according to the timing of the byte-selection pin (WEn#). For details, please refer to the Data Sheet for the corresponding memory.

If the BAS bit in CSnWCR is set to 1, the WEn# pin and RD/WR# pin timings change. Figure 8.34 shows the basic access timing. In write access, data is written to the memory according to the timing of the write enable pin (RD/WR#). The data hold timing from RD/WR# negation to data write must be acquired by setting the HW1 and HW0 bits in CSnWCR. Figure 8.35 shows the access timing when a software wait is specified.



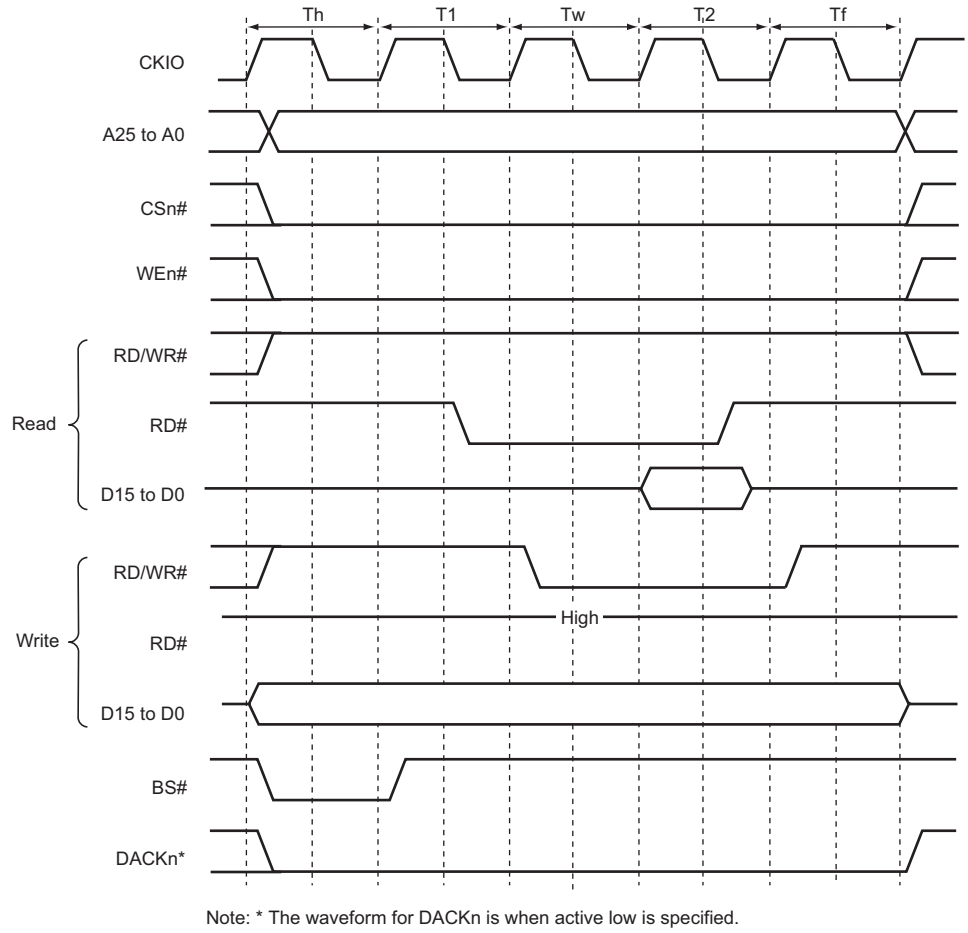
Note: * The waveform for DACKn is when active low is specified.

Figure 8.33 Basic Access Timing for SRAM with Byte Selection (BAS = 0)



Note: * The waveform for DACKn is when active low is specified.

Figure 8.34 Basic Access Timing for SRAM with Byte Selection (BAS = 1)



Note: * The waveform for DACKn is when active low is specified.

Figure 8.35 Wait Timing for SRAM with Byte Selection (BAS = 1) (SW[1:0] = 01, WR[3:0] = 0001, HW[1:0] = 01)

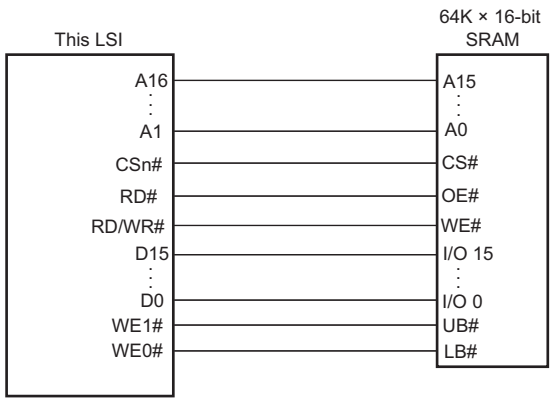


Figure 8.36 Example of Connection with 16-Bit Data-Width SRAM with Byte Selection

8.5.9 Burst ROM (Clocked Synchronous) Interface

The burst ROM (clocked synchronous) interface is supported to access a ROM with a synchronous burst function at high speed. The burst ROM interface accesses the burst ROM in the same way as a normal space. This interface is valid only for area 0.

In the first access cycle, wait cycles are inserted. In this case, the number of wait cycles to be inserted is specified by the W3 to W0 bits in CS0WCR. In the second and subsequent cycles, the number of wait cycles to be inserted is specified by the BW1 and BW0 bits in CS0WCR.

While the burst ROM (clocked synchronous) is accessed, the BS# signal is asserted only for the first access cycle and an external wait input is also valid for the first access cycle. When the bus width is 16 bits, the burst length must be specified as 8. The burst ROM interface does not support the 8-bit bus width for the burst ROM.

The burst ROM interface performs burst operations for all read access. For example, in a 32-bit access over a 16-bit bus, valid 16-bit data is read two times and invalid 16-bit data is read six times. These invalid data read cycles increase the memory access time and degrade the program execution speed and DMA transfer speed. To prevent this problem, it is recommended using a read in a 16-byte or more access size. The burst ROM interface performs write access in the same way as normal space access.

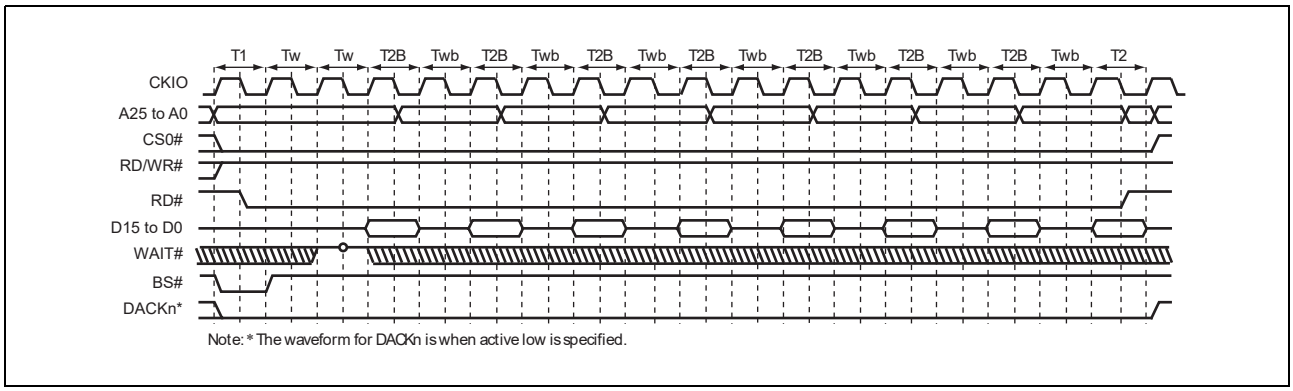


Figure 8.37 Burst ROM Access Timing (Clocked Synchronous) (Burst Length = 8, Wait Cycles Inserted in First Access = 2, Wait Cycles Inserted in Second and Subsequent Access Cycles = 1)

8.5.10 Wait between Access Cycles

As the operating frequency of LSIs becomes higher, the off-operation of the data buffer often collides with the next data access when the read operation from devices with slow access speed is completed. As a result of these collisions, the reliability of the device is low and malfunctions may occur. A function that avoids data collisions by inserting idle (wait) cycles between continuous access cycles has been newly added.
The number of wait cycles between access cycles can be set by the WM bit in CSnWCR, bits IWW2 to IWW0, IWRWD2 to IWRWD0, IWRWS2 to IWRWS0, IWRRD2 to IWRRD0, and IWRRS2 to IWRRS 0 in CSnBCR. The conditions for setting the idle cycles between access cycles are shown below.

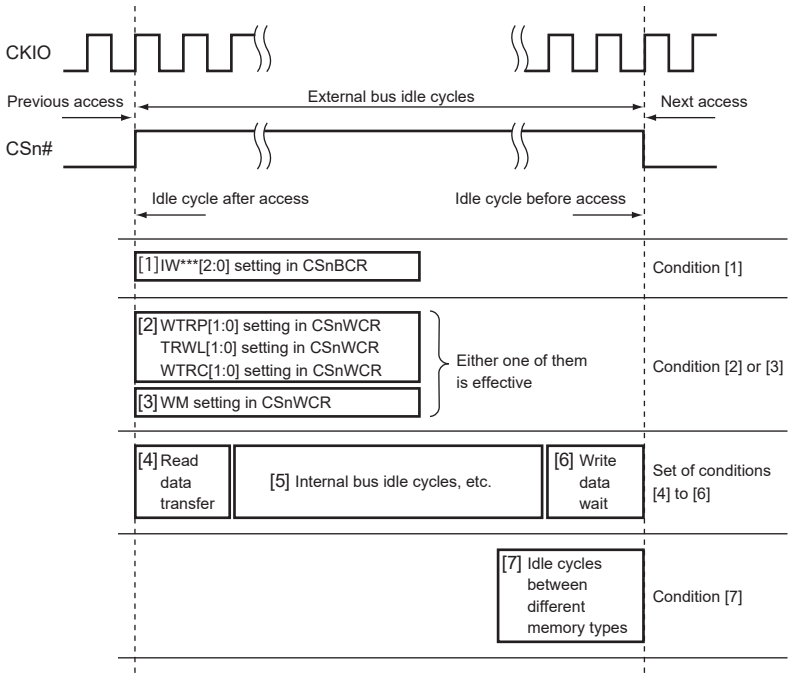
1. Continuous access cycles are write-read or write-write
2. Continuous access cycles are read-write for different spaces
3. Continuous access cycles are read-write for the same space
4. Continuous access cycles are read-read for different spaces
5. Continuous access cycles are read-read for the same space

For the specification of the number of idle cycles between access cycles described above, refer to the description of each register.
Besides the idle cycles between access cycles specified by the registers, idle cycles must be inserted to interface with the internal bus or to obtain the minimum pulse width for a multiplexed pin (WEn#). The following gives detailed information about the idle cycles and describes how to estimate the number of idle cycles.
The number of idle cycles on the external bus from CSn# negation to CSn# or CSm# assertion is described below.
There are seven conditions that determine the number of idle cycles on the external bus as shown in Table 8.14. The effects of these conditions are shown in Figure 8.38.

Table 8.14 Conditions for Determining Number of Idle Cycles

No.	Condition	Description	Range	Note
[1]	IW***[2:0] in CSnBCR	These bits specify the number of idle cycles for access. The number of idle cycles can be specified independently for each combination of the previous and next cycles. For example, in the case where reading CS1 space followed by reading other CS space, the bits IWRRD[2:0] in CS1BCR should be set to B'100 to specify six or more idle cycles. This condition is effective only for access cycles other than single address transfer and generates idle cycles after the access is completed.	0 to 12	Do not set 0 for the number of idle cycles between memory types which are not allowed to be accessed successively.
[2]	SDRAM-related bits in CSnWCR	These bits specify precharge completion and startup wait cycles and idle cycles between commands for SDRAM access. This condition is effective only for SDRAM access and generates idle cycles after the access is completed	0 to 3	Specify these bits in accordance with the specification of the target SDRAM.
[3]	WM in CSnWCR	This bit enables or disables external WAIT# pin input for the memory types other than SDRAM. When this bit is cleared to 0 (external WAIT# enabled), one idle cycle is inserted to check the external WAIT# pin input after the access is completed. When this bit is set to 1 (disabled), no idle cycle is generated.	0 or 1	
[4]	Read data transfer cycle	One idle cycle is inserted after a read access is completed. This idle cycle is not generated for the first or middle cycles in divided access cycles. This is neither generated when the HW[1:0] bits in CSnWCR are not B'00.	0 or 1	One idle cycle is always generated after a read cycle with SDRAM.
[5]	Internal bus idle cycles, etc.	External bus access requests from the CPU or the direct memory access controller and their results are passed through the internal bus. The external bus enters idle state during internal bus idle cycles or while a bus other than the external bus is being accessed. This condition is not effective for divided access cycles, which are generated by the bus state controller when the access size is larger than the external data bus width.	0 or larger	The number of internal bus idle cycles may not become 0 depending on the CPU: internal bus: CKIO
[6]	Write data wait cycles	During write access, a write cycle is executed on the external bus only after the write data becomes ready. This write data wait period generates idle cycles before the write cycle. Note that when the previous cycle is a write cycle and the internal bus idle cycles are shorter than the previous write cycle, write data can be prepared in parallel with the previous write cycle and therefore, no idle cycle is generated (write buffer effect).	0 or 1	For write → write or write → read access cycles, successive access cycles without idle cycles may be available due to the write buffer effect described in the left column. If successive access cycles without idle cycles are not allowed, specify the minimum number of idle cycles between access cycles through CSnBCR.
[7]	Idle cycles between different memory types	To ensure the minimum pulse width on the signal-multiplexed pins, idle cycles may be inserted before access after memory types are switched. For some memory types, idle cycles are inserted even when memory types are not switched.	0 to 2	The number of idle cycles depends on the target memory types. See Table 8.15.

In the above conditions, a total of four conditions, that is, condition [1], condition [2] or [3] (either one is effective), a set of conditions [4] to [6] (these are generated successively, and therefore the sum of them should be taken as one set of idle cycles), and condition [7] are generated at the same time. The maximum number of idle cycles among these four conditions become the number of idle cycles on the external bus. To ensure the minimum idle cycles, be sure to make register settings for condition [1].



Note: A total of four conditions (condition [1], condition [2] or [3], a set of conditions [4] to [6], and condition [7]) generate idle cycle at the same time. Accordingly, the maximum number of cycles among these four conditions become the number of idle cycles.

Figure 8.38 Idle Cycle Conditions

Table 8.15 Number of Idle Cycles Inserted between Access Cycles to Different Memory Types

	Next Cycle						
Previous Cycle	SRAM	Burst ROM (Asynchronous)	MPX-I/O	Byte SRAM (BAS = 0)	Byte SRAM (BAS = 1)	SDRAM	Burst ROM (Synchronous)
SRAM	0	0	1	0	0/1*1	0/1*1	0
Burst ROM (asynchronous)	0	0	1	0	0/1*1	0/1*1	0
MPX-I/O	1	1	0	1	1	1	1
Byte SRAM (BAS = 0)	0	0	1	0	0/1*1	0/1*1	0
Byte SRAM (BAS = 1)	0/1*1	0/1*1	1/2*1	0/1*1	0	0	0/1*1
SDRAM	1	1	2	1	0	0	1
Burst ROM (synchronous)	0	0	1	0	1	1	0

Note 1. The number of idle cycles is determined by the setting of bits HW[1:0] in the CSnWCR register for the previous cycle. The values on the left and right sides of the virgules show the numbers of idle cycles when HW[1:0] ≠ B'00 and HW[1:0] = B'00, respectively. If the memory connected to the CSn space in the previous cycle is of a type for which bits HW[1:0] in the CSnWCR register are ineffective, the number of idle cycles will be the value on the right side.

8.5.11 Others

(1) Reset

This module can be initialized completely only at power-on reset. At power-on reset, all signals are negated and data output buffers are turned off regardless of the bus cycle state after the internal reset is synchronized with the internal clock. All control registers are initialized. In software standby and sleep, control registers of the bus state controller are not initialized.

(2) Caution on Write Buffer

Since the bus state controller incorporates a one-stage write buffer, it can execute an access via the internal bus before the previous external bus cycle is completed in a write cycle. If the on-chip module is read or written after the external low-speed memory is written, the on-chip module can be accessed before the completion of the external low-speed memory write cycle.

In read cycles, the CPU is placed in the wait state until read operation has been completed. To continue the process after the data write to the device has been completed, perform a dummy read to the same address to check for completion of the write before the next process to be executed.

The write buffer of the bus state controller functions in the same way for an access by a bus master other than the CPU such as the direct memory access controller. Accordingly, to perform DMA transfers, the next read cycle is initiated before the previous write cycle is completed. Note, however, that if both the DMA source and destination addresses exist in external memory space, the next read cycle will not be initiated until the previous write cycle is completed.

Changing the registers in this module while the write buffer is operating may disrupt correct write access. Therefore, do not change the registers in this module immediately after a write access. If this change becomes necessary, do it after executing a dummy read of the write data.

(3) On-Chip Peripheral Module Access

To access an on-chip module register, two or more peripheral module clock (P0φ or P1φ) cycles are required. Care must be taken in system design.

When the CPU writes data to the internal peripheral registers, the CPU performs the succeeding instructions without waiting for the completion of writing to registers.

For example, a case is described here in which the system is transferring to the software standby mode for power savings. To make this transition, the WFI instruction must be performed after setting the STBY bit in the STBCR1 register to 1. However a dummy read of the STBCR1 register is required before executing the WFI instruction. If a dummy read is omitted, the CPU executes the WFI instruction before the STBY bit is set to 1, thus the system enters sleep mode not software standby mode. A dummy read of the STBCR1 register is indispensable to complete writing to the STBY bit.

To reflect the change by internal peripheral registers while performing the succeeding instructions, execute a dummy read of registers to which write instruction is given and then perform the succeeding instructions.

(4) Usage Note on Changing CKIO Clock Selections

The CKIO clock can be selected and switched between 27.5 MHz and 132 MHz, but it can not be switched while CS0 to CS5 external address space is being accessed. (Setting of CKIO select register (CKIOSEL) is prohibited while external address space is being accessed.)

For example, a procedure for switching the CKIO clock from 66 MHz to 132 MHz and using SDRAM in the CS2 or CS3 space, after starting up in the normal space interface of CS0 in boot mode 0 is as follows:

1. Transfer the program to a memory area other than CS0 to CS5 (internal RAM, etc.).
2. Jump to the area where the program was transferred.
3. Configure the bus state controller that the transferred program in the CS0 to CS5 areas can operate even if the CKIO clock is 132 MHz.
4. Set the CKIO clock to 132 MHz with the CKIO select register (CKIOSEL).
5. Continue instruction execution with internal RAM and so on, or return to the CS0 program area.

It is also prohibited to access the CS0 to CS5 external address space from a bus master other than the CPU such as the direct memory access controller while switching the CKIO clock.

9. Direct Memory Access Controller

The direct memory access controller can be used in place of the CPU to perform high-speed transfers between external devices that have DACK (transfer request acknowledge signal), external memory, on-chip memory, memory-mapped external devices, and on-chip peripheral modules.
This module has a controller that handles secure and non-secure access. For details on secure and non-secure access, see section 5.8.1 (3), 5.8.2, 5.8.3, and 5.8.4. Do not assign the same DMA transfer request to secure access and non-secure access, respectively.

9.1 Features

- Number of channels selectable: 16 channels (CH0 to CH15). Only the CH0 channel can receive external requests.
- 4-Gbyte address space (according to the architecture)
- Transfer data size: Byte, two bytes, four bytes, eight bytes, 16 bytes, 32 bytes, 64 bytes, and 128 bytes
- Maximum transfer count: 2^{32} - 1 bytes
- Address mode: Dual address mode
- Transfer requests: Can be selected from the three types of external request, on-chip peripheral module request, and auto request (software trigger)
- Transfer mode: Single transfer mode and block transfer mode are selectable.
- Priority: The channel priority levels within channels 0 to 7 and within channels 8 to 15 are selectable between fixed mode and round-robin mode (the channel priority level between the group of channels 0 to 7 and the group of channels 8 to 15 is round-robin mode).
- Interrupt request: An interrupt request can be sent to the CPU on completion of data transfer (DMA transfer end interrupt per channel) or on occurrence of a transfer error (DMA error interrupt).
- External request detection: Low level detection, high level detection, rising edge detection, and falling edge detection are selectable for DREQ input detection.
- The DMA registers have a continuous execution function that allows the next DMA transfer to be executed continuously by making settings for the next DMA transfer during execution of the current DMA transfer. This continuous execution function can be enabled or disabled independently in each channel.
- Link mode: In this mode, the setting data (descriptor data) located in the memory by the CPU is automatically retrieved by the DMAC, and DMA transfer is performed according to those values.
- Buffer sweep: If an ongoing DMA transfer is forced to end, the data already retrieved into the buffer can be output before DMA transfer ends.
- Interval: A specific DMA transfer interval can be specified to adjust the bus occupancy.

9.2 Input/Output Pins

Table 9.1 lists the pin configuration. This module has pins for a single channel (CH0) as the external bus use.

Table 9.1 Pin Configuration

Channel	Name	Pin Name	I/O	Function
0	DMA transfer request	DREQ0	Input	DMA transfer request input from an external device to channel 0
	DMA transfer request acknowledge	DACK0	Output	DMA transfer request acknowledge output from channel 0 of this module
	DMA transfer end	TEND0	Output	DMA transfer end output for channel 0 of this module

Note 1. For the active level of DACK0 and TEND0, refer to section 8, Bus State Controller.

9.3 Register Configuration

The register configuration is shown in the figure below.

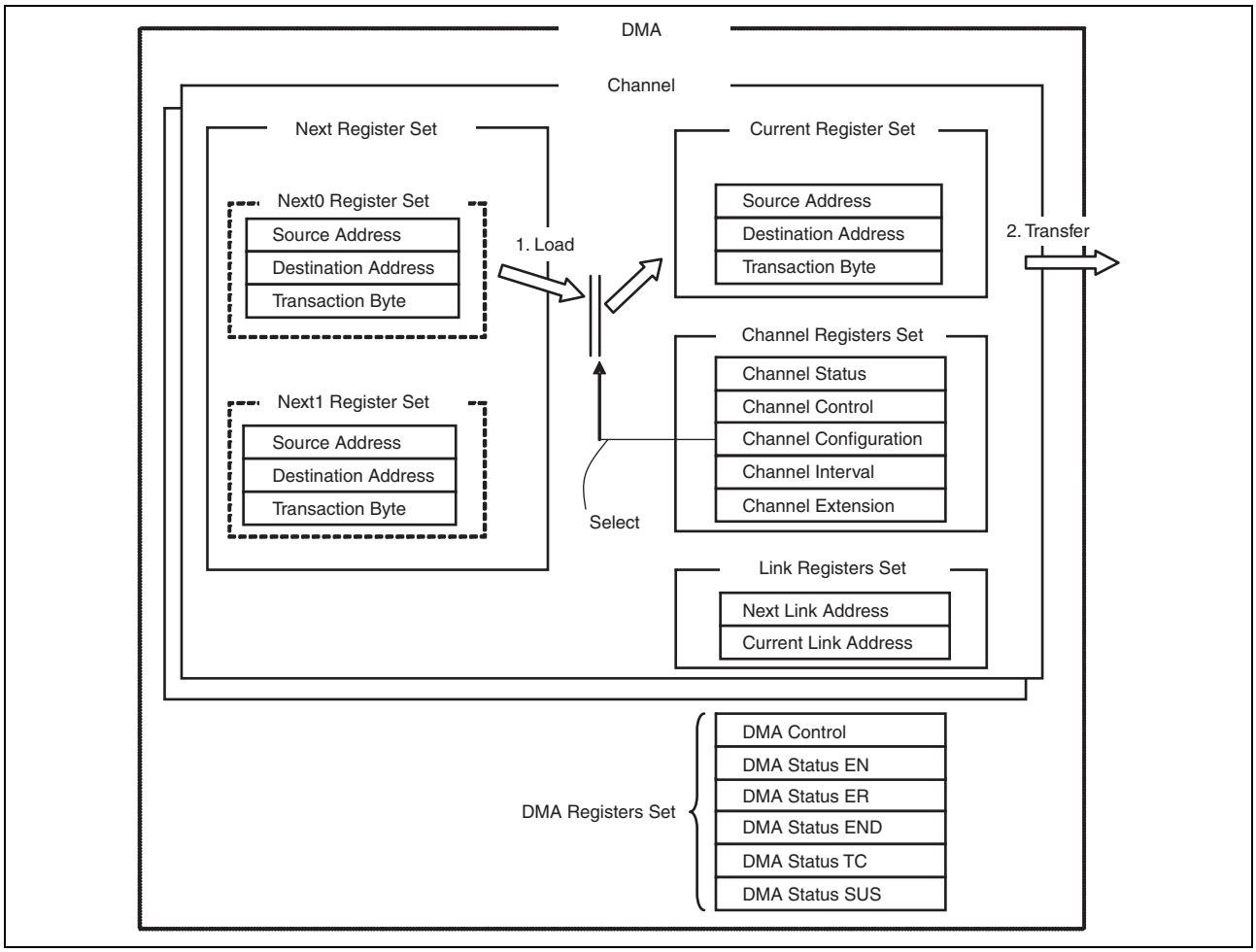


Figure 9.1 Register Configuration

(a) Next Register Set

This register set is used to set the source address, destination address, and transfer byte count of the DMA transaction to be executed next.

It consists of the Next0 register set and the Next1 register set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These register set values are loaded to the Current Register Set and used for DMA transfer.

(b) Current Register Set

This register set indicates the source address, destination address, and transfer byte count of the currently executed DMA transaction.

The values are loaded from the Next0/1 register set (register mode) or from the descriptor read data (link mode). The user cannot write directly to this register set.

The register set is automatically updated each time a DMA transaction is executed.

(c) Channel Register Set

This register set is used to make the DMA transfer settings.

The settings to be made with this register set include channel status indication, channel control, DMA transaction setting, and DMA transaction interval.

(d) Link Register Set

This register set consists of a register that sets the address of the descriptor to be loaded next in link mode (Next Link Address Register) and a register that indicates the address of the currently executed descriptor (Current Link Address Register).
The Current Link Address Register is automatically updated when a descriptor is read. The user cannot write directly to this register set.

(e) DMA Register Set

This register set consists of a register that controls DMA as a whole and registers that indicate the status of the corresponding channels. It enables channel priority control as well as the monitoring of the channel status (EN, ER, END, TC, and SUS).

(f) Extended Resource Selector Register Set

This register set is used to select the on-chip peripheral module to perform DMA transfer and the external request.

9.4 Register Descriptions

Table 9.2 lists the register configuration. There are eleven control registers and five status registers for each channel, and twelve common control registers are used by all channels. In addition, there is one extension resource selector per two channels.

The notation for the registers of each channel is as follows.

Next 0 Source Address Register of channel 0 for secure access: Next 0 Source Address Register 0S (abbreviation: N0SA_0S)

Next 0 Source Address Register of channel 0 for non-secure access: Next 0 Source Address Register 0 (abbreviation: N0SA_0)

Table 9.2 Register Configuration

		Abbreviation				Address		
Channel	Register Name	Non-secure	Secure	R/W	Initial Value	Non-secure	Secure	Access Size
0	Next0 Source Address Register 0/0S	N0SA_0	N0SA_0S	RW	H'00000000	H'E8226000	H'E8220000	32
	Next0 Destination Address Register 0/0S	N0DA_0	N0DA_0S	RW	H'00000000	H'E8226004	H'E8220004	32
	Next0 Transaction Byte Register 0/0S	N0TB_0	N0TB_0S	RW	H'00000000	H'E8226008	H'E8220008	32
	Next1 Source Address Register 0/0S	N1SA_0	N1SA_0S	RW	H'00000000	H'E822600C	H'E822000C	32
	Next1 Destination Address Register 0/0S	N1DA_0	N1DA_0S	RW	H'00000000	H'E8226010	H'E8220010	32
	Next1 Transaction Byte Register 0/0S	N1TB_0	N1TB_0S	RW	H'00000000	H'E8226014	H'E8220014	32
	Current Source Address Register 0/0S	CRSA_0	CRSA_0S	R	H'00000000	H'E8226018	H'E8220018	32
	Current Destination Address Register 0/0S	CRDA_0	CRDA_0S	R	H'00000000	H'E822601C	H'E822001C	32
	Current Transaction Byte Register 0/0S	CRTB_0	CRTB_0S	R	H'00000000	H'E8226020	H'E8220020	32
	Channel Status Register 0/0S	CHSTAT_0	CHSTAT_0S	R	H'00000000	H'E8226024	H'E8220024	32
	Channel Control Register 0/0S	CHCTRL_0	CHCTRL_0S	RW	H'00000000	H'E8226028	H'E8220028	32
	Channel Configuration Register 0/0S	CHCFG_0	CHCFG_0S	RW	H'00000000	H'E822602C	H'E822002C	32
	Channel Interval Register 0/0S	CHITVL_0	CHITVL_0S	RW	H'00000000	H'E8226030	H'E8220030	32
	Channel Extension Register 0/0S	CHEXT_0	CHEXT_0S	RW	H'00000000	H'E8226034	H'E8220034	32
	Next Link Address Register 0/0S	NXLA_0	NXLA_0S	RW	H'00000000	H'E8226038	H'E8220038	32
	Current Link Address Register 0/0S	CRLA_0	CRLA_0S	R	H'00000000	H'E822603C	H'E822003C	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
1	Next0 Source Address Register 1/1S	N0SA_1	N0SA_1S	RW	H'00000000	H'E8226040	H'E8220040	32
	Next0 Destination Address Register 1/1S	N0DA_1	N0DA_1S	RW	H'00000000	H'E8226044	H'E8220044	32
	Next0 Transaction Byte Register 1/1S	N0TB_1	N0TB_1S	RW	H'00000000	H'E8226048	H'E8220048	32
	Next1 Source Address Register 1/1S	N1SA_1	N1SA_1S	RW	H'00000000	H'E822604C	H'E822004C	32
	Next1 Destination Address Register 1/1S	N1DA_1	N1DA_1S	RW	H'00000000	H'E8226050	H'E8220050	32
	Next1 Transaction Byte Register 1/1S	N1TB_1	N1TB_1S	RW	H'00000000	H'E8226054	H'E8220054	32
	Current Source Address Register 1/1S	CRSA_1	CRSA_1S	R	H'00000000	H'E8226058	H'E8220058	32
	Current Destination Address Register 1/1S	CRDA_1	CRDA_1S	R	H'00000000	H'E822605C	H'E822005C	32
	Current Transaction Byte Register 1/1S	CRTB_1	CRTB_1S	R	H'00000000	H'E8226060	H'E8220060	32
	Channel Status Register 1/1S	CHSTAT_1	CHSTAT_1S	R	H'00000000	H'E8226064	H'E8220064	32
	Channel Control Register 1/1S	CHCTRL_1	CHCTRL_1S	RW	H'00000000	H'E8226068	H'E8220068	32
	Channel Configuration Register 1/1S	CHCFG_1	CHCFG_1S	RW	H'00000000	H'E822606C	H'E822006C	32
	Channel Interval Register 1/1S	CHITVL_1	CHITVL_1S	RW	H'00000000	H'E8226070	H'E8220070	32
	Channel Extension Register 1/1S	CHEXT_1	CHEXT_1S	RW	H'00000000	H'E8226074	H'E8220074	32
	Next Link Address Register 1/1S	NXLA_1	NXLA_1S	RW	H'00000000	H'E8226078	H'E8220078	32
	Current Link Address Register 1/1S	CRLA_1	CRLA_1S	R	H'00000000	H'E822607C	H'E822007C	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
2	Next0 Source Address Register 2/2S	N0SA_2	N0SA_2S	RW	H'00000000	H'E8226080	H'E8220080	32
	Next0 Destination Address Register 2/2S	N0DA_2	N0DA_2S	RW	H'00000000	H'E8226084	H'E8220084	32
	Next0 Transaction Byte Register 2/2S	N0TB_2	N0TB_2S	RW	H'00000000	H'E8226088	H'E8220088	32
	Next1 Source Address Register 2/2S	N1SA_2	N1SA_2S	RW	H'00000000	H'E822608C	H'E822008C	32
	Next1 Destination Address Register 2/2S	N1DA_2	N1DA_2S	RW	H'00000000	H'E8226090	H'E8220090	32
	Next1 Transaction Byte Register 2/2S	N1TB_2	N1TB_2S	RW	H'00000000	H'E8226094	H'E8220094	32
	Current Source Address Register 2/2S	CRSA_2	CRSA_2S	R	H'00000000	H'E8226098	H'E8220098	32
	Current Destination Address Register 2/2S	CRDA_2	CRDA_2S	R	H'00000000	H'E822609C	H'E822009C	32
	Current Transaction Byte Register 2/2S	CRTB_2	CRTB_2S	R	H'00000000	H'E82260A0	H'E82200A0	32
	Channel Status Register 2/2S	CHSTAT_2	CHSTAT_2S	R	H'00000000	H'E82260A4	H'E82200A4	32
	Channel Control Register 2/2S	CHCTRL_2	CHCTRL_2S	RW	H'00000000	H'E82260A8	H'E82200A8	32
	Channel Configuration Register 2/2S	CHCFG_2	CHCFG_2S	RW	H'00000000	H'E82260AC	H'E82200AC	32
	Channel Interval Register 2/2S	CHITVL_2	CHITVL_2S	RW	H'00000000	H'E82260B0	H'E82200B0	32
	Channel Extension Register 2/2S	CHEXT_2	CHEXT_2S	RW	H'00000000	H'E82260B4	H'E82200B4	32
	Next Link Address Register 2/2S	NXLA_2	NXLA_2S	RW	H'00000000	H'E82260B8	H'E82200B8	32
	Current Link Address Register 2/2S	CRLA_2	CRLA_2S	R	H'00000000	H'E82260BC	H'E82200BC	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
3	Next0 Source Address Register 3/3S	N0SA_3	N0SA_3S	RW	H'00000000	H'E82260C0	H'E82200C0	32
	Next0 Destination Address Register 3/3S	N0DA_3	N0DA_3S	RW	H'00000000	H'E82260C4	H'E82200C4	32
	Next0 Transaction Byte Register 3/3S	N0TB_3	N0TB_3S	RW	H'00000000	H'E82260C8	H'E82200C8	32
	Next1 Source Address Register 3/3S	N1SA_3	N1SA_3S	RW	H'00000000	H'E82260CC	H'E82200CC	32
	Next1 Destination Address Register 3/3S	N1DA_3	N1DA_3S	RW	H'00000000	H'E82260D0	H'E82200D0	32
	Next1 Transaction Byte Register 3/3S	N1TB_3	N1TB_3S	RW	H'00000000	H'E82260D4	H'E82200D4	32
	Current Source Address Register 3/3S	CRSA_3	CRSA_3S	R	H'00000000	H'E82260D8	H'E82200D8	32
	Current Destination Address Register 3/3S	CRDA_3	CRDA_3S	R	H'00000000	H'E82260DC	H'E82200DC	32
	Current Transaction Byte Register 3/3S	CRTB_3	CRTB_3S	R	H'00000000	H'E82260E0	H'E82200E0	32
	Channel Status Register 3/3S	CHSTAT_3	CHSTAT_3S	R	H'00000000	H'E82260E4	H'E82200E4	32
	Channel Control Register 3/3S	CHCTRL_3	CHCTRL_3S	RW	H'00000000	H'E82260E8	H'E82200E8	32
	Channel Configuration Register 3/3S	CHCFG_3	CHCFG_3S	RW	H'00000000	H'E82260EC	H'E82200EC	32
	Channel Interval Register 3/3S	CHITVL_3	CHITVL_3S	RW	H'00000000	H'E82260F0	H'E82200F0	32
	Channel Extension Register 3/3S	CHEXT_3	CHEXT_3S	RW	H'00000000	H'E82260F4	H'E82200F4	32
	Next Link Address Register 3/3S	NXLA_3	NXLA_3S	RW	H'00000000	H'E82260F8	H'E82200F8	32
	Current Link Address Register 3/3S	CRLA_3	CRLA_3S	R	H'00000000	H'E82260FC	H'E82200FC	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
4	Next0 Source Address Register 4/4S	N0SA_4	N0SA_4S	RW	H'00000000	H'E8226100	H'E8220100	32
	Next0 Destination Address Register 4/4S	N0DA_4	N0DA_4S	RW	H'00000000	H'E8226104	H'E8220104	32
	Next0 Transaction Byte Register 4/4S	N0TB_4	N0TB_4S	RW	H'00000000	H'E8226108	H'E8220108	32
	Next1 Source Address Register 4/4S	N1SA_4	N1SA_4S	RW	H'00000000	H'E822610C	H'E822010C	32
	Next1 Destination Address Register 4/4S	N1DA_4	N1DA_4S	RW	H'00000000	H'E8226110	H'E8220110	32
	Next1 Transaction Byte Register 4/4S	N1TB_4	N1TB_4S	RW	H'00000000	H'E8226114	H'E8220114	32
	Current Source Address Register 4/4S	CRSA_4	CRSA_4S	R	H'00000000	H'E8226118	H'E8220118	32
	Current Destination Address Register 4/4S	CRDA_4	CRDA_4S	R	H'00000000	H'E822611C	H'E822011C	32
	Current Transaction Byte Register 4/4S	CRTB_4	CRTB_4S	R	H'00000000	H'E8226120	H'E8220120	32
	Channel Status Register 4/4S	CHSTAT_4	CHSTAT_4S	R	H'00000000	H'E8226124	H'E8220124	32
	Channel Control Register 4/4S	CHCTRL_4	CHCTRL_4S	RW	H'00000000	H'E8226128	H'E8220128	32
	Channel Configuration Register 4/4S	CHCFG_4	CHCFG_4S	RW	H'00000000	H'E822612C	H'E822012C	32
	Channel Interval Register 4/4S	CHITVL_4	CHITVL_4S	RW	H'00000000	H'E8226130	H'E8220130	32
	Channel Extension Register 4/4S	CHEXT_4	CHEXT_4S	RW	H'00000000	H'E8226134	H'E8220134	32
	Next Link Address Register 4/4S	NXLA_4	NXLA_4S	RW	H'00000000	H'E8226138	H'E8220138	32
	Current Link Address Register 4/4S	CRLA_4	CRLA_4S	R	H'00000000	H'E822613C	H'E822013C	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
5	Next0 Source Address Register 5/5S	N0SA_5	N0SA_5S	RW	H'00000000	H'E8226140	H'E8220140	32
	Next0 Destination Address Register 5/5S	N0DA_5	N0DA_5S	RW	H'00000000	H'E8226144	H'E8220144	32
	Next0 Transaction Byte Register 5/5S	N0TB_5	N0TB_5S	RW	H'00000000	H'E8226148	H'E8220148	32
	Next1 Source Address Register 5/5S	N1SA_5	N1SA_5S	RW	H'00000000	H'E822614C	H'E822014C	32
	Next1 Destination Address Register 5/5S	N1DA_5	N1DA_5S	RW	H'00000000	H'E8226150	H'E8220150	32
	Next1 Transaction Byte Register 5/5S	N1TB_5	N1TB_5S	RW	H'00000000	H'E8226154	H'E8220154	32
	Current Source Address Register 5/5S	CRSA_5	CRSA_5S	R	H'00000000	H'E8226158	H'E8220158	32
	Current Destination Address Register 5/5S	CRDA_5	CRDA_5S	R	H'00000000	H'E822615C	H'E822015C	32
	Current Transaction Byte Register 5/5S	CRTB_5	CRTB_5S	R	H'00000000	H'E8226160	H'E8220160	32
	Channel Status Register 5/5S	CHSTAT_5	CHSTAT_5S	R	H'00000000	H'E8226164	H'E8220164	32
	Channel Control Register 5/5S	CHCTRL_5	CHCTRL_5S	RW	H'00000000	H'E8226168	H'E8220168	32
	Channel Configuration Register 5/5S	CHCFG_5	CHCFG_5S	RW	H'00000000	H'E822616C	H'E822016C	32
	Channel Interval Register 5/5S	CHITVL_5	CHITVL_5S	RW	H'00000000	H'E8226170	H'E8220170	32
	Channel Extension Register 5/5S	CHEXT_5	CHEXT_5S	RW	H'00000000	H'E8226174	H'E8220174	32
	Next Link Address Register 5/5S	NXLA_5	NXLA_5S	RW	H'00000000	H'E8226178	H'E8220178	32
	Current Link Address Register 5/5S	CRLA_5	CRLA_5S	R	H'00000000	H'E822617C	H'E822017C	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
6	Next0 Source Address Register 6/6S	N0SA_6	N0SA_6S	RW	H'00000000	H'E8226180	H'E8220180	32
	Next0 Destination Address Register 6/6S	N0DA_6	N0DA_6S	RW	H'00000000	H'E8226184	H'E8220184	32
	Next0 Transaction Byte Register 6/6S	N0TB_6	N0TB_6S	RW	H'00000000	H'E8226188	H'E8220188	32
	Next1 Source Address Register 6/6S	N1SA_6	N1SA_6S	RW	H'00000000	H'E822618C	H'E822018C	32
	Next1 Destination Address Register 6/6S	N1DA_6	N1DA_6S	RW	H'00000000	H'E8226190	H'E8220190	32
	Next1 Transaction Byte Register 6/6S	N1TB_6	N1TB_6S	RW	H'00000000	H'E8226194	H'E8220194	32
	Current Source Address Register 6/6S	CRSA_6	CRSA_6S	R	H'00000000	H'E8226198	H'E8220198	32
	Current Destination Address Register 6/6S	CRDA_6	CRDA_6S	R	H'00000000	H'E822619C	H'E822019C	32
	Current Transaction Byte Register 6/6S	CRTB_6	CRTB_6S	R	H'00000000	H'E82261A0	H'E82201A0	32
	Channel Status Register 6/6S	CHSTAT_6	CHSTAT_6S	R	H'00000000	H'E82261A4	H'E82201A4	32
	Channel Control Register 6/6S	CHCTRL_6	CHCTRL_6S	RW	H'00000000	H'E82261A8	H'E82201A8	32
	Channel Configuration Register 6/6S	CHCFG_6	CHCFG_6S	RW	H'00000000	H'E82261AC	H'E82201AC	32
	Channel Interval Register 6/6S	CHITVL_6	CHITVL_6S	RW	H'00000000	H'E82261B0	H'E82201B0	32
	Channel Extension Register 6/6S	CHEXT_6	CHEXT_6S	RW	H'00000000	H'E82261B4	H'E82201B4	32
	Next Link Address Register 6/6S	NXLA_6	NXLA_6S	RW	H'00000000	H'E82261B8	H'E82201B8	32
	Current Link Address Register 6/6S	CRLA_6	CRLA_6S	R	H'00000000	H'E82261BC	H'E82201BC	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
7	Next0 Source Address Register 7/7S	N0SA_7	N0SA_7S	RW	H'00000000	H'E82261C0	H'E82201C0	32
	Next0 Destination Address Register 7/7S	N0DA_7	N0DA_7S	RW	H'00000000	H'E82261C4	H'E82201C4	32
	Next0 Transaction Byte Register 7/7S	N0TB_7	N0TB_7S	RW	H'00000000	H'E82261C8	H'E82201C8	32
	Next1 Source Address Register 7/7S	N1SA_7	N1SA_7S	RW	H'00000000	H'E82261CC	H'E82201CC	32
	Next1 Destination Address Register 7/7S	N1DA_7	N1DA_7S	RW	H'00000000	H'E82261D0	H'E82201D0	32
	Next1 Transaction Byte Register 7/7S	N1TB_7	N1TB_7S	RW	H'00000000	H'E82261D4	H'E82201D4	32
	Current Source Address Register 7/7S	CRSA_7	CRSA_7S	R	H'00000000	H'E82261D8	H'E82201D8	32
	Current Destination Address Register 7/7S	CRDA_7	CRDA_7S	R	H'00000000	H'E82261DC	H'E82201DC	32
	Current Transaction Byte Register 7/7S	CRTB_7	CRTB_7S	R	H'00000000	H'E82261E0	H'E82201E0	32
	Channel Status Register 7/7S	CHSTAT_7	CHSTAT_7S	R	H'00000000	H'E82261E4	H'E82201E4	32
	Channel Control Register 7/7S	CHCTRL_7	CHCTRL_7S	RW	H'00000000	H'E82261E8	H'E82201E8	32
	Channel Configuration Register 7/7S	CHCFG_7	CHCFG_7S	RW	H'00000000	H'E82261EC	H'E82201EC	32
	Channel Interval Register 7/7S	CHITVL_7	CHITVL_7S	RW	H'00000000	H'E82261F0	H'E82201F0	32
	Channel Extension Register 7/7S	CHEXT_7	CHEXT_7S	RW	H'00000000	H'E82261F4	H'E82201F4	32
	Next Link Address Register 7/7S	NXLA_7	NXLA_7S	RW	H'00000000	H'E82261F8	H'E82201F8	32
	Current Link Address Register 7/7S	CRLA_7	CRLA_7S	R	H'00000000	H'E82261FC	H'E82201FC	32
Common for 0 to 7	DMA Control Registers 0-7/0-7S	DCTRL_0_7	DCTRL_0_7S	R	H'00000000	H'E8226300	H'E8220300	32
	DMA Status EN Registers 0-7/0-7S	DSTAT_EN_0_7	DSTAT_EN_0_7S	R	H'00000000	H'E8226310	H'E8220310	32
	DMA Status ER Registers 0-7/0-7S	DSTAT_ER_0_7	DSTAT_ER_0_7S	R	H'00000000	H'E8226314	H'E8220314	32
	DMA Status END Registers 0-7/0-7S	DSTAT_END_0_7	DSTAT_END_0_7S	R	H'00000000	H'E8226318	H'E8220318	32
	DMA Status TC Registers 0-7/0-7S	DSTAT_TC_0_7	DSTAT_TC_0_7S	R	H'00000000	H'E822631C	H'E822031C	32
	DMA Status SUS Registers 0-7/0-7S	DSTAT_SUS_0_7	DSTAT_SUS_0_7S	R	H'00000000	H'E8226320	H'E8220320	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
8	Next0 Source Address Register 8/8S	N0SA_8	N0SA_8S	RW	H'00000000	H'E8226400	H'E8220400	32
	Next0 Destination Address Register 8/8S	N0DA_8	N0DA_8S	RW	H'00000000	H'E8226404	H'E8220404	32
	Next0 Transaction Byte Register 8/8S	N0TB_8	N0TB_8S	RW	H'00000000	H'E8226408	H'E8220408	32
	Next1 Source Address Register 8/8S	N1SA_8	N1SA_8S	RW	H'00000000	H'E822640C	H'E822040C	32
	Next1 Destination Address Register 8/8S	N1DA_8	N1DA_8S	RW	H'00000000	H'E8226410	H'E8220410	32
	Next1 Transaction Byte Register 8/8S	N1TB_8	N1TB_8S	RW	H'00000000	H'E8226414	H'E8220414	32
	Current Source Address Register 8/8S	CRSA_8	CRSA_8S	R	H'00000000	H'E8226418	H'E8220418	32
	Current Destination Address Register 8/8S	CRDA_8	CRDA_8S	R	H'00000000	H'E822641C	H'E822041C	32
	Current Transaction Byte Register 8/8S	CRTB_8	CRTB_8S	R	H'00000000	H'E8226420	H'E8220420	32
	Channel Status Register 8/8S	CHSTAT_8	HSTAT_8S	R	H'00000000	H'E8226424	H'E8220424	32
	Channel Control Register 8/8S	CHCTRL_8	CHCTRL_8S	RW	H'00000000	H'E8226428	H'E8220428	32
	Channel Configuration Register 8/8S	CHCFG_8	CHCFG_8S	RW	H'00000000	H'E822642C	H'E822042C	32
	Channel Interval Register 8/8S	CHITVL_8	CHITVL_8S	RW	H'00000000	H'E8226430	H'E8220430	32
	Channel Extension Register 8/8S	CHEXT_8	CHEXT_8S	RW	H'00000000	H'E8226434	H'E8220434	32
	Next Link Address Register 8/8S	NXLA_8	NXLA_8S	RW	H'00000000	H'E8226438	H'E8220438	32
	Current Link Address Register 8/8S	CRLA_8	CRLA_8S	R	H'00000000	H'E822643C	H'E822043C	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
9	Next0 Source Address Register 9/9S	N0SA_9	N0SA_9S	RW	H'00000000	H'E8226440	H'E8220440	32
	Next0 Destination Address Register 9/9S	N0DA_9	N0DA_9S	RW	H'00000000	H'E8226444	H'E8220444	32
	Next0 Transaction Byte Register 9/9S	N0TB_9	N0TB_9S	RW	H'00000000	H'E8226448	H'E8220448	32
	Next1 Source Address Register 9/9S	N1SA_9	N1SA_9S	RW	H'00000000	H'E822644C	H'E822044C	32
	Next1 Destination Address Register 9/9S	N1DA_9	N1DA_9S	RW	H'00000000	H'E8226450	H'E8220450	32
	Next1 Transaction Byte Register 9/9S	N1TB_9	N1TB_9S	RW	H'00000000	H'E8226454	H'E8220454	32
	Current Source Address Register 9/9S	CRSA_9	CRSA_9S	R	H'00000000	H'E8226458	H'E8220458	32
	Current Destination Address Register 9/9S	CRDA_9	CRDA_9S	R	H'00000000	H'E822645C	H'E822045C	32
	Current Transaction Byte Register 9/9S	CRTB_9	CRTB_9S	R	H'00000000	H'E8226460	H'E8220460	32
	Channel Status Register 9/9S	CHSTAT_9	CHSTAT_9S	R	H'00000000	H'E8226464	H'E8220464	32
	Channel Control Register 9/9S	CHCTRL_9	CHCTRL_9S	RW	H'00000000	H'E8226468	H'E8220468	32
	Channel Configuration Register 9/9S	CHCFG_9	CHCFG_9S	RW	H'00000000	H'E822646C	H'E822046C	32
	Channel Interval Register 9/9S	CHITVL_9	CHITVL_9S	RW	H'00000000	H'E8226470	H'E8220470	32
	Channel Extension Register 9/9S	CHEXT_9	CHEXT_9S	RW	H'00000000	H'E8226474	H'E8220474	32
	Next Link Address Register 9/9S	NXLA_9	NXLA_9S	RW	H'00000000	H'E8226478	H'E8220478	32
	Current Link Address Register 9/9S	CRLA_9	CRLA_9S	R	H'00000000	H'E822647C	H'E822047C	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
10	Next0 Source Address Register 10/10S	N0SA_10	N0SA_10S	RW	H'00000000	H'E8226480	H'E8220480	32
	Next0 Destination Address Register 10/10S	N0DA_10	N0DA_10S	RW	H'00000000	H'E8226484	H'E8220484	32
	Next0 Transaction Byte Register 10/10S	N0TB_10	N0TB_10S	RW	H'00000000	H'E8226488	H'E8220488	32
	Next1 Source Address Register 10/10S	N1SA_10	N1SA_10S	RW	H'00000000	H'E822648C	H'E822048C	32
	Next1 Destination Address Register 10/10S	N1DA_10	N1DA_10S	RW	H'00000000	H'E8226490	H'E8220490	32
	Next1 Transaction Byte Register 10/10S	N1TB_10	N1TB_10S	RW	H'00000000	H'E8226494	H'E8220494	32
	Current Source Address Register 10/10S	CRSA_10	CRSA_10S	R	H'00000000	H'E8226498	H'E8220498	32
	Current Destination Address Register 10/10S	CRDA_10	CRDA_10S	R	H'00000000	H'E822649C	H'E822049C	32
	Current Transaction Byte Register 10/10S	CRTB_10	CRTB_10S	R	H'00000000	H'E82264A0	H'E82204A0	32
	Channel Status Register 10/10S	CHSTAT_10	CHSTAT_10S	R	H'00000000	H'E82264A4	H'E82204A4	32
	Channel Control Register 10/10S	CHCTRL_10	CHCTRL_10S	RW	H'00000000	H'E82264A8	H'E82204A8	32
	Channel Configuration Register 10/10S	CHCFG_10	CHCFG_10S	RW	H'00000000	H'E82264AC	H'E82204AC	32
	Channel Interval Register 10/10S	CHITVL_10	CHITVL_10S	RW	H'00000000	H'E82264B0	H'E82204B0	32
	Channel Extension Register 10/10S	CHEXT_10	CHEXT_10S	RW	H'00000000	H'E82264B4	H'E82204B4	32
	Next Link Address Register 10/10S	NXLA_10	NXLA_10S	RW	H'00000000	H'E82264B8	H'E82204B8	32
	Current Link Address Register 10/10S	CRLA_10	CRLA_10S	R	H'00000000	H'E82264BC	H'E82204BC	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
11	Next0 Source Address Register 11/11S	N0SA_11	N0SA_11S	RW	H'00000000	H'E82264C0	H'E82204C0	32
	Next0 Destination Address Register 11/11S	N0DA_11	N0DA_11S	RW	H'00000000	H'E82264C4	H'E82204C4	32
	Next0 Transaction Byte Register 11/11S	N0TB_11	N0TB_11S	RW	H'00000000	H'E82264C8	H'E82204C8	32
	Next1 Source Address Register 11/11S	N1SA_11	N1SA_11S	RW	H'00000000	H'E82264CC	H'E82204CC	32
	Next1 Destination Address Register 11/11S	N1DA_11	N1DA_11S	RW	H'00000000	H'E82264D0	H'E82204D0	32
	Next1 Transaction Byte Register 11/11S	N1TB_11	N1TB_11S	RW	H'00000000	H'E82264D4	H'E82204D4	32
	Current Source Address Register 11/11S	CRSA_11	CRSA_11S	R	H'00000000	H'E82264D8	H'E82204D8	32
	Current Destination Address Register 11/11S	CRDA_11	CRDA_11S	R	H'00000000	H'E82264DC	H'E82204DC	32
	Current Transaction Byte Register 11/11S	CRTB_11	CRTB_11S	R	H'00000000	H'E82264E0	H'E82204E0	32
	Channel Status Register 11/11S	CHSTAT_11	CHSTAT_11S	R	H'00000000	H'E82264E4	H'E82204E4	32
	Channel Control Register 11/11S	CHCTRL_11	CHCTRL_11S	RW	H'00000000	H'E82264E8	H'E82204E8	32
	Channel Configuration Register 11/11S	CHCFG_11	CHCFG_11S	RW	H'00000000	H'E82264EC	H'E82204EC	32
	Channel Interval Register 11/11S	CHITVL_11	CHITVL_11S	RW	H'00000000	H'E82264F0	H'E82204F0	32
	Channel Extension Register 11/11S	CHEXT_11	CHEXT_11S	RW	H'00000000	H'E82264F4	H'E82204F4	32
	Next Link Address Register 11/11S	NXLA_11	NXLA_11S	RW	H'00000000	H'E82264F8	H'E82204F8	32
	Current Link Address Register 11/11S	CRLA_11	CRLA_11S	R	H'00000000	H'E82264FC	H'E82204FC	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
12	Next0 Source Address Register 12/12S	N0SA_12	N0SA_12S	RW	H'00000000	H'E8226500	H'E8220500	32
	Next0 Destination Address Register 12/12S	N0DA_12	N0DA_12S	RW	H'00000000	H'E8226504	H'E8220504	32
	Next0 Transaction Byte Register 12/12S	N0TB_12	N0TB_12S	RW	H'00000000	H'E8226508	H'E8220508	32
	Next1 Source Address Register 12/12S	N1SA_12	N1SA_12S	RW	H'00000000	H'E822650C	H'E822050C	32
	Next1 Destination Address Register 12/12S	N1DA_12	N1DA_12S	RW	H'00000000	H'E8226510	H'E8220510	32
	Next1 Transaction Byte Register 12/12S	N1TB_12	N1TB_12S	RW	H'00000000	H'E8226514	H'E8220514	32
	Current Source Address Register 12/12S	CRSA_12	CRSA_12S	R	H'00000000	H'E8226518	H'E8220518	32
	Current Destination Address Register 12/12S	CRDA_12	CRDA_12S	R	H'00000000	H'E822651C	H'E822051C	32
	Current Transaction Byte Register 12/12S	CRTB_12	CRTB_12S	R	H'00000000	H'E8226520	H'E8220520	32
	Channel Status Register 12/12S	CHSTAT_12	CHSTAT_12S	R	H'00000000	H'E8226524	H'E8220524	32
	Channel Control Register 12/12S	CHCTRL_12	CHCTRL_12S	RW	H'00000000	H'E8226528	H'E8220528	32
	Channel Configuration Register 12/12S	CHCFG_12	CHCFG_12S	RW	H'00000000	H'E822652C	H'E822052C	32
	Channel Interval Register 12/12S	CHITVL_12	CHITVL_12S	RW	H'00000000	H'E8226530	H'E8220530	32
	Channel Extension Register 12/12S	CHEXT_12	CHEXT_12S	RW	H'00000000	H'E8226534	H'E8220534	32
	Next Link Address Register 12/12S	NXLA_12	NXLA_12S	RW	H'00000000	H'E8226538	H'E8220538	32
	Current Link Address Register 12/12S	CRLA_12	CRLA_12S	R	H'00000000	H'E822653C	H'E822053C	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
13	Next0 Source Address Register 13/13S	N0SA_13	N0SA_13S	RW	H'00000000	H'E8226540	H'E8220540	32
	Next0 Destination Address Register 13/13S	N0DA_13	N0DA_13S	RW	H'00000000	H'E8226544	H'E8220544	32
	Next0 Transaction Byte Register 13/13S	N0TB_13	N0TB_13S	RW	H'00000000	H'E8226548	H'E8220548	32
	Next1 Source Address Register 13/13S	N1SA_13	N1SA_13S	RW	H'00000000	H'E822654C	H'E822054C	32
	Next1 Destination Address Register 13/13S	N1DA_13	N1DA_13S	RW	H'00000000	H'E8226550	H'E8220550	32
	Next1 Transaction Byte Register 13/13S	N1TB_13	N1TB_13S	RW	H'00000000	H'E8226554	H'E8220554	32
	Current Source Address Register 13/13S	CRSA_13	CRSA_13S	R	H'00000000	H'E8226558	H'E8220558	32
	Current Destination Address Register 13/13S	CRDA_13	CRDA_13S	R	H'00000000	H'E822655C	H'E822055C	32
	Current Transaction Byte Register 13/13S	CRTB_13	CRTB_13S	R	H'00000000	H'E8226560	H'E8220560	32
	Channel Status Register 13/13S	CHSTAT_13	CHSTAT_13S	R	H'00000000	H'E8226564	H'E8220564	32
	Channel Control Register 13/13S	CHCTRL_13	CHCTRL_13S	RW	H'00000000	H'E8226568	H'E8220568	32
	Channel Configuration Register 13/13S	CHCFG_13	CHCFG_13S	RW	H'00000000	H'E822656C	H'E822056C	32
	Channel Interval Register 13/13S	CHITVL_13	CHITVL_13S	RW	H'00000000	H'E8226570	H'E8220570	32
	Channel Extension Register 13/13S	CHEXT_13	CHEXT_13S	RW	H'00000000	H'E8226574	H'E8220574	32
	Next Link Address Register 13/13S	NXLA_13	NXLA_13S	RW	H'00000000	H'E8226578	H'E8220578	32
	Current Link Address Register 13/13S	CRLA_13	CRLA_13S	R	H'00000000	H'E822657C	H'E822057C	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
14	Next0 Source Address Register 14/14S	N0SA_14	N0SA_14S	RW	H'00000000	H'E8226580	H'E8220580	32
	Next0 Destination Address Register 14/14S	N0DA_14	N0DA_14S	RW	H'00000000	H'E8226584	H'E8220584	32
	Next0 Transaction Byte Register 14/14S	N0TB_14	N0TB_14S	RW	H'00000000	H'E8226588	H'E8220588	32
	Next1 Source Address Register 14/14S	N1SA_14	N1SA_14S	RW	H'00000000	H'E822658C	H'E822058C	32
	Next1 Destination Address Register 14/14S	N1DA_14	N1DA_14S	RW	H'00000000	H'E8226590	H'E8220590	32
	Next1 Transaction Byte Register 14/14S	N1TB_14	N1TB_14S	RW	H'00000000	H'E8226594	H'E8220594	32
	Current Source Address Register 14/14S	CRSA_14	CRSA_14S	R	H'00000000	H'E8226598	H'E8220598	32
	Current Destination Address Register 14/14S	CRDA_14	CRDA_14S	R	H'00000000	H'E822659C	H'E822059C	32
	Current Transaction Byte Register 14/14S	CRTB_14	CRTB_14S	R	H'00000000	H'E82265A0	H'E82205A0	32
	Channel Status Register 14/14S	CHSTAT_14	CHSTAT_14S	R	H'00000000	H'E82265A4	H'E82205A4	32
	Channel Control Register 14/14S	CHCTRL_14	CHCTRL_14S	RW	H'00000000	H'E82265A8	H'E82205A8	32
	Channel Configuration Register 14/14S	CHCFG_14	CHCFG_14S	RW	H'00000000	H'E82265AC	H'E82205AC	32
	Channel Interval Register 14/14S	CHITVL_14	CHITVL_14S	RW	H'00000000	H'E82265B0	H'E82205B0	32
	Channel Extension Register 14/14S	CHEXT_14	CHEXT_14S	RW	H'00000000	H'E82265B4	H'E82205B4	32
	Next Link Address Register 14/14S	NXLA_14	NXLA_14S	RW	H'00000000	H'E82265B8	H'E82205B8	32
	Current Link Address Register 14/14S	CRLA_14	CRLA_14S	R	H'00000000	H'E82265BC	H'E82205BC	32

Table 9.2 Register Configuration

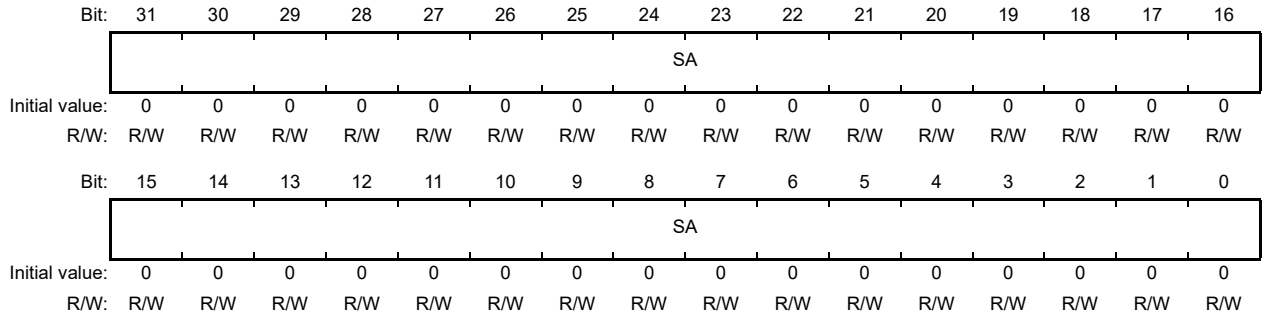
Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
15	Next0 Source Address Register 15/15S	N0SA_15	N0SA_15S	RW	H'00000000	H'E82265C0	H'E82205C0	32
	Next0 Destination Address Register 15/15S	N0DA_15	N0DA_15S	RW	H'00000000	H'E82265C4	H'E82205C4	32
	Next0 Transaction Byte Register 15/15S	N0TB_15	N0TB_15S	RW	H'00000000	H'E82265C8	H'E82205C8	32
	Next1 Source Address Register 15/15S	N1SA_15	N1SA_15S	RW	H'00000000	H'E82265CC	H'E82205CC	32
	Next1 Destination Address Register 15/15S	N1DA_15	N1DA_15S	RW	H'00000000	H'E82265D0	H'E82205D0	32
	Next1 Transaction Byte Register 15/15S	N1TB_15	N1TB_15S	RW	H'00000000	H'E82265D4	H'E82205D4	32
	Current Source Address Register 15/15S	CRSA_15	CRSA_15S	R	H'00000000	H'E82265D8	H'E82205D8	32
	Current Destination Address Register 15/15S	CRDA_15	CRDA_15S	R	H'00000000	H'E82265DC	H'E82205DC	32
	Current Transaction Byte Register 15/15S	CRTB_15	CRTB_15S	R	H'00000000	H'E82265E0	H'E82205E0	32
	Channel Status Register 15/15S	CHSTAT_15	CHSTAT_15S	R	H'00000000	H'E82265E4	H'E82205E4	32
	Channel Control Register 15/15S	CHCTRL_15	CHCTRL_15S	RW	H'00000000	H'E82265E8	H'E82205E8	32
	Channel Configuration Register 15/15S	CHCFG_15	CHCFG_15S	RW	H'00000000	H'E82265EC	H'E82205EC	32
	Channel Interval Register 15/15S	CHITVL_15	CHITVL_15S	RW	H'00000000	H'E82265F0	H'E82205F0	32
	Channel Extension Register 15/15S	CHEXT_15	CHEXT_15S	RW	H'00000000	H'E82265F4	H'E82205F4	32
	Next Link Address Register 15/15S	NXLA_15	NXLA_15S	RW	H'00000000	H'E82265F8	H'E82205F8	32
	Current Link Address Register 15/15S	CRLA_15	CRLA_15S	R	H'00000000	H'E82265FC	H'E82205FC	32
Common for 8 to 15	DMA Control Registers 8-15/8-15S	DCTRL_8_15	DCTRL_8_15S	R	H'00000000	H'E8226700	H'E8220700	32
	DMA Status EN Registers 8-15/8-15S	DSTAT_EN_8_15	DSTAT_EN_8_15S	R	H'00000000	H'E8226710	H'E8220710	32
	DMA Status ER Registers 8-15/8-15S	DSTAT_ER_8_15	DSTAT_ER_8_15S	R	H'00000000	H'E8226714	H'E8220714	32
	DMA Status END Registers 8-15/8-15S	DSTAT_END_8_15	DSTAT_END_8_15S	R	H'00000000	H'E8226718	H'E8220718	32
	DMA Status TC Registers 8-15/8-15S	DSTAT_TC_8_15	DSTAT_TC_8_15S	R	H'00000000	H'E822671C	H'E822071C	32
	DMA Status SUS Registers 8-15/8-15S	DSTAT_SUS_8_15	DSTAT_SUS_8_15S	R	H'00000000	H'E8226720	H'E8220720	32
0/1	DMA Extension Resource Selectors 0/0S	DMARS0	DMARS0S	RW	H'00000000	H'FCFE2000	H'FCFE1000	32
2/3	DMA Extension Resource Selectors 1/1S	DMARS1	DMARS1S	RW	H'00000000	H'FCFE2004	H'FCFE1004	32

Table 9.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
4/5	DMA Extension Resource Selectors 2/2S	DMARS 2	DMARS 2S	RW	H'00000000	H'FCFE2008	H'FCFE1008	32
6/7	DMA Extension Resource Selectors 3/3S	DMARS 3	DMARS 3S	RW	H'00000000	H'FCFE200C	H'FCFE100C	32
8/9	DMA Extension Resource Selectors 4/4S	DMARS 4	DMARS 4S	RW	H'00000000	H'FCFE2010	H'FCFE1010	32
10/11	DMA Extension Resource Selectors 5/5S	DMARS 5	DMARS 5S	RW	H'00000000	H'FCFE2014	H'FCFE1014	32
12/13	DMA Extension Resource Selectors 6/6S	DMARS 6	DMARS 6S	RW	H'00000000	H'FCFE2018	H'FCFE1018	32
14/15	DMA Extension Resource Selectors 7/7S	DMARS 7	DMARS 7S	RW	H'00000000	H'FCFE201C	H'FCFE101C	32

9.4.1 Next Source Address Register n/nS (N0SA_n/nS, N1SA_n/nS)

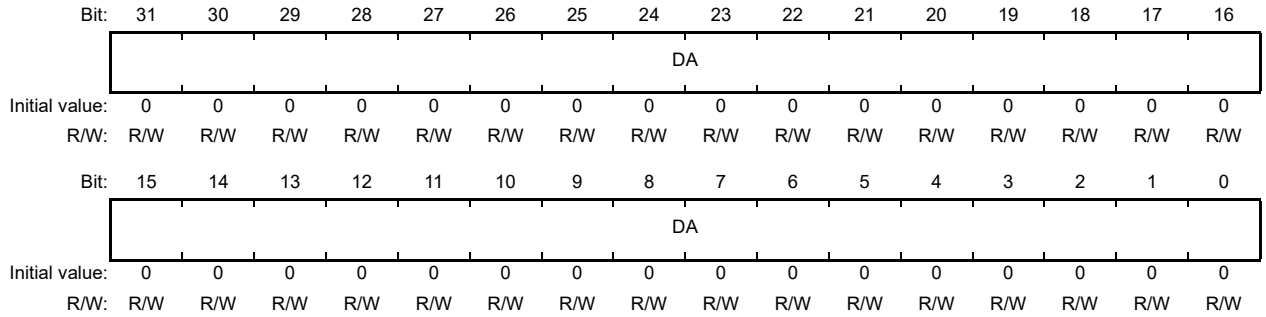
This register sets the DMA transfer source address (32 bits) of DMA channel n (n = 0 to 15) which is to be executed next.
N0SA_n/nS is for the Next0 Register Set, and N1SA_n/nS is for the Next1 Register Set.
In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.
These register set values are loaded to the Current Register Set and used for DMA transfer.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SA	All 0	R/W	Source Address Sets the start address of the DMA transfer source.

9.4.2 Next Destination Address Register n/nS (N0DA_n/nS, N1DA_n/nS)

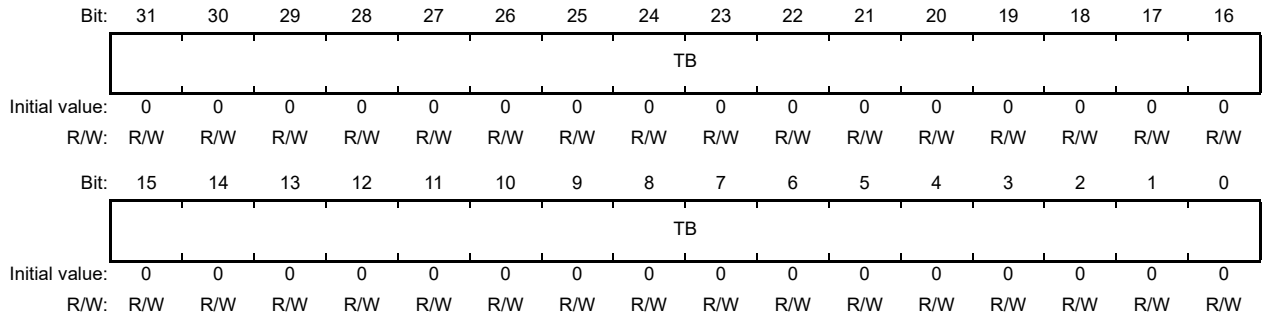
This register sets the DMA transfer destination address (32 bits) of DMA channel n (n = 0 to 15) which is to be executed next.
N0DA_n/nS is for the Next0 Register Set, and N1DA_n/nS is for the Next1 Register Set.
In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.
These register set values are loaded to the Current Register Set and used for DMA transfer.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DA	All 0	R/W	Destination Address Sets the start address of the DMA transfer destination.

9.4.3 Next Transaction Byte Register n/nS (N0TB_n/nS, N1TB_n/nS)

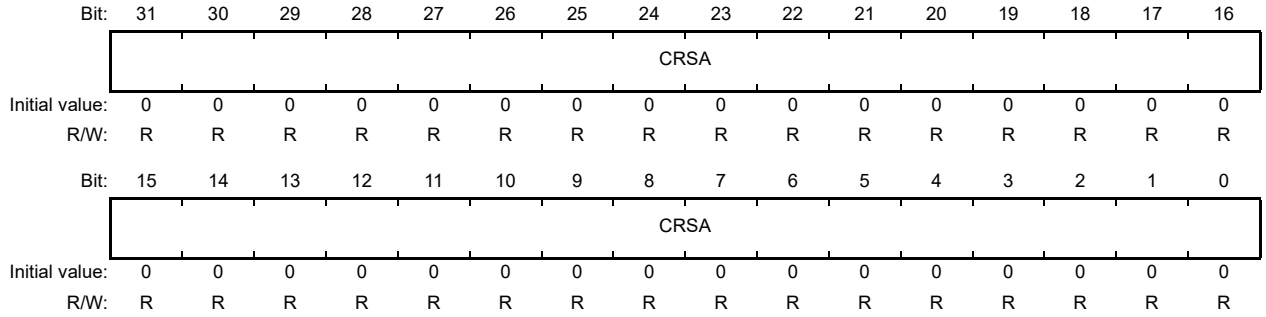
This register sets the total transfer byte count (DMA transaction) of DMA channel (n = 0 to 15) which is to be executed next.
N0TB_n/nS is for the Next0 Register Set, and N1TB_n/nS is for the Next1 Register Set.
In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.
These register set values are loaded to the Current Register Set and used for DMA transfer.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TB	All 0	R/W	Transaction Byte Sets the total transfer byte count. Caution: Do not start a DMA transaction with 0 set in this register.

9.4.4 Current Source Address Register n/nS (CRSA_n/nS)

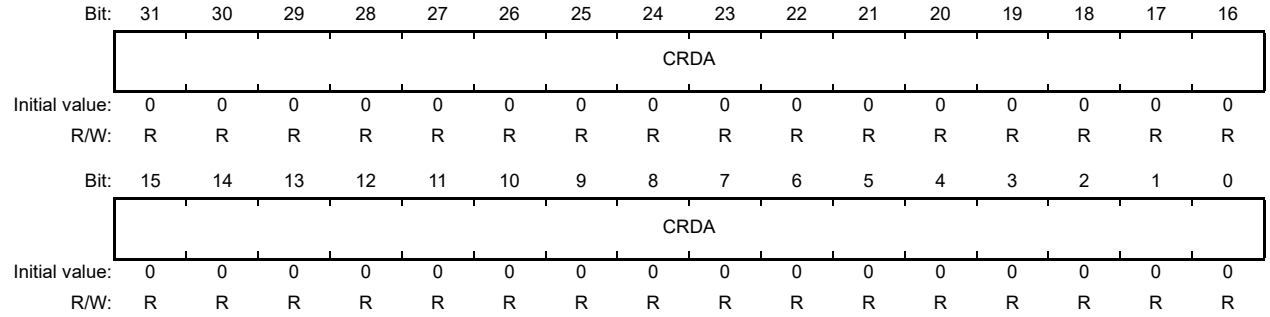
This register indicates the DMA transfer source address of DMA channel n (n = 0 to 15).
The values are loaded from the Next0/1 register set in register mode or from the descriptor read data in link mode. This register cannot be written by software.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRSA	All 0	R	Current Source Address Register Indicates the read address of the next DMA transaction. The value automatically increments during the DMA transaction. (The value is fixed when 1 is set in SAD of the CHCFG_n/nS register.) The value increments when a read transfer starts. Read this register after DMA stops (0 is set in EN of the CHSTAT_n/nS register). (Any value obtained during the DMA operation should be handled as a reference value.)

9.4.5 Current Destination Address Register n/nS (CRDA_n/nS)

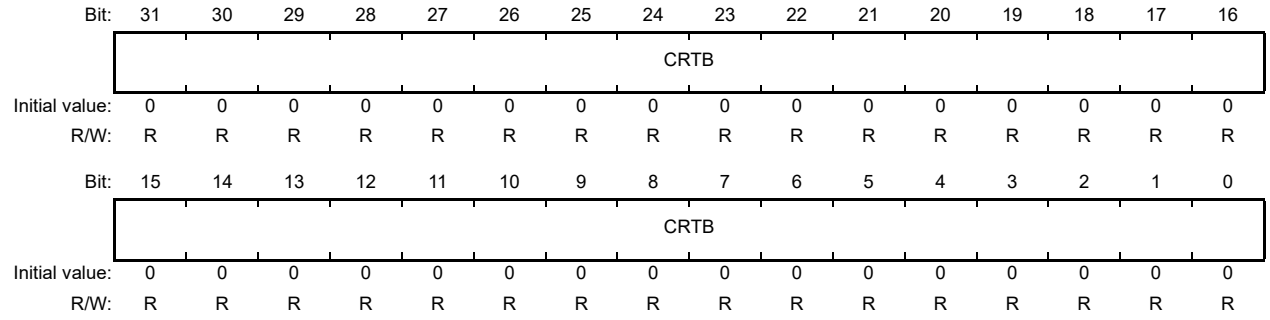
This register indicates the DMA transfer destination address of DMA channel n (n = 0 to 15).
The values are loaded from the Next0/1 register set in register mode or from the descriptor read data in link mode. This register cannot be written by software.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRDA	All 0	R	Current Destination Address Register Indicates the write address of the next DMA transaction. The value automatically increments during the DMA transaction. (The value is fixed when 1 is set in DAD of the CHCFG_n/nS register.) The value increments when a write transfer starts. Read this register after DMA stops (0 is set in EN of the CHSTAT_n/nS register). (Any value obtained during the DMA operation should be handled as a reference value.)

9.4.6 Current Transaction Byte Register n/nS (CRTB_n/nS)

This register indicates the total transfer byte count of DMA channel n (n = 0 to 15). The value of this register becomes 0 when the transaction ends.
The values are loaded from the Next0/1 register set in register mode or from the descriptor read data in link mode. This register cannot be written by software.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRTB	All 0	R	Current Transaction Byte Register Indicates the remaining transfer byte count of the currently executed DMA transaction. The value automatically decrements during the DMA transaction. The value decrements when a write transfer is completed. Read this register after DMA stops (0 is set in EN of the CHSTAT_n/nS register). (Any value obtained during the DMA operation should be handled as a reference value.)

9.4.7 Channel Status Register n/nS (CHSTAT_n/nS)

This register indicates the status of DMA channel n (n = 0 to 15).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INT MSK
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MODE	DER	DW	DL	SR	TC	END	ER	SUS	TACT	RQST	EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
16	INTMSK	0	R	Indicates the temporary mask status of the DMA transfer end interrupt. 1: Masked temporarily 0: Unmasked temporarily Set condition(s): • When SETINTMSK (CHCTRL_n/nS) is set to 1 Reset condition(s): • When CLRINTMSK (CHCTRL_n/nS) is set to 1 • When SWRST (CHCTRL_n/nS) is set to 1
15 to 12	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
11	MODE	0	R	DMA Mode Indicates the DMA mode. It corresponds to the value set in the DMS bit of the CHCFG_n/nS register. 0: Register mode 1: Link mode
10	DER	0	R	Descriptor Error Indicates whether the link valid value of the read descriptor is invalid (LV = 0) (this is not dependent on the DIM level of the descriptor). If a descriptor error has occurred, the transfer is stopped but no DMA error interrupt occurs. 0: Descriptor Error not detected 1: Descriptor Error detected Set condition(s): • When the LV value loaded with the descriptor in link mode is 0 Reset condition(s): • When SWRST (CHCTRL_n/nS) is set to 1
9	DW	0	R	Descriptor WriteBack Indicates the descriptor writeback status. The bit maintains 1 if a bus error is received during descriptor writeback. 0: Operation other than writeback is being performed for the header in link mode. 1: (ER = 0) Writeback is being performed for the header in link mode. (ER = 1) A bus error occurs during writeback for the header in link mode. Set condition(s): • When header writeback in link mode starts Reset condition(s): • When header writeback in link mode ends with an OK response • When SWRST (CHCTRL_n/nS) is set to 1

Bit	Bit Name	Initial Value	R/W	Description
8	DL	0	R	Descriptor Load Indicates whether the descriptor is being loaded. The bit maintains 1 if a bus error is received during descriptor load. 0: Operation other than descriptor load 1: (ER = 0) Descriptor load is in progress in link mode. (ER = 1) A bus error occurs during descriptor load in link mode. Set condition(s): • When descriptor load in link mode starts Reset condition(s): • When descriptor load in link mode ends with an OK response • When SWRST (CHCTRL_n/nS) is set to 1
7	SR	0	R	Selected Register Set Indicates the register set currently selected in register mode. 0: Next0 Register Set 1: Next1 Register Set Set condition(s): • When RSEL (CHCFG_n/nS) is set to 1 Reset condition(s): • When RSEL (CHCFG_n/nS) is set to 0
6	TC	0	R	Terminal Count Indicates whether the DMA transaction is completed. 0: DMA transfer not completed 1: DMA transfer completed Set condition(s): • When data equivalent to the total transfer byte count set in the CRTB register has been transferred in register mode • When data equivalent to the total transfer byte count set in the CRTB register has been transferred in link mode, with 1 set in WBD of the descriptor header • When descriptor writeback is completed in link mode, with 0 set in WBD of the descriptor header Clear condition(s): • When the CLRTC (CHCTRL_n/nS) bit is set to 1 • When the SWRST (CHCTRL_n/nS) bit is set to 1
5	END	0	R	DMAEND Interrupted Indicates whether the DMA transaction is completed and whether the DMA transfer end interrupt has occurred. 0: DMA transfer not completed 1: DMA transfer completed Set condition(s): • When one of the set conditions for the TC bit is met and 0 is set in DEM of the CHCFG_n/nS register • When the descriptor is read in link mode and both LV of the header and DIM are set to 0 Clear condition(s): • When CLREND (CHCTRL_n/nS) is set to 1 • When SWRST (CHCTRL_n/nS) is set to 1
4	ER	0	R	Error bit Indicates that a DMA error interrupt has occurred because an error response has been received from the transfer source or destination and a bus error has occurred during the DMA transfer. 0: No bus error has occurred 1: A DMA error interrupt has occurred due to a bus error Set condition(s): • When a bus error has occurred during a bus cycle Clear condition(s): • When SWRST (CHCTRL_n/nS) is set to 1

Bit	Bit Name	Initial Value	R/W	Description
3	SUS	0	R	Suspend Indicates whether the channel is suspended. 0: Channel_n not suspended 1: Channel_n suspended Set condition(s): • When SETSUS (CHCTRL_n/nS) is set to 1 during a DMA transfer on Channel_n, creating a SUSPEND status internally Clear condition(s): • When CLRSUS (CHCTRL_n/nS) is set to 1 • When CLREN (CHCTRL_n/nS) is set to 1
2	TACT	0	R	Transaction Active Indicates whether the DMAC is active. This bit is intended to check that the channel is completely inactive. 0: DMA on Channel_n inactive 1: DMA on Channel_n active Set condition(s): • When a DMA transaction starts on Channel_n Clear condition(s): • When a DMA transaction is completed
1	RQST	0	R	Request Indicates whether a transfer request is being received. 0: DMA transfer request not being received 1: DMA transfer request being received Set condition(s): • When the STG bit (CHCTRL_n/nS) is set to 1 (auto request) • When a transfer request is received from the DMA request source set in the CHCFG_n/nS register Clear condition(s): • When SWRST (CHCTRL_n/nS) is set to 1 • When CLRRQ (CHCTRL_n/nS) is set to 1 • When a transfer is executed on the side specified by REQD (CHCFG_n/nS) in single transfer mode (TM = 0). • When all DMA transactions are completed in register mode (the transaction ends with REN set to 0) • When the DMA transfer of the last descriptor (LE = 1) is completed in link mode • When descriptor read stops (LV = 0) in link mode • When a bus error is received due to an error response
0	EN	0	R	Enable Indicates whether the operation of DMA channel n is enabled or disabled. 0: Operation disabled 1: Operation enabled Set condition(s): • When SETEN (CHCTRL_n/nS) is set to 1 Clear condition(s): • When SWRST (CHCTRL_n/nS) is set to 1 • When CLREN (CHCTRL_n/nS) is set to 1 • When a bus error is received due to an error response during the transfer • When all DMA transactions are completed in register mode (the transaction ends with REN set to 0) • When the DMA transfer of the last descriptor (LE = 1) is completed in link mode (writeback when WBD is set to 0) • When descriptor read stops (LV = 0) in link mode

<Caution> If the ER bit is set to 1 for any transfer, the whole transfer should be handled as invalid.
To suspend a DMA transaction, mask or clear the transfer request or clear the Enable bit (for the procedure, see section 9.7.11 (2), Transfer Stop).
If a transfer request from an on-chip peripheral module or the external DREQ input is made concurrently with an auto request (by setting 1 in the STG bit) for the same one channel, the trigger source that takes effect cannot be identified. Make sure that only one of these transfer requests is used in the system.
When transfer is requested by an auto request, wait for the last requested DMA transfer to complete (use the Current Register or other data to check the status) before setting the STG bit for the next transfer request.

9.4.8 Channel Control Register n/nS (CHCTRL_n/nS)

This register controls the DMA transfer operation on DMA channel n (n = 0 to 15).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLRINTMSK	SETINTMSK
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLRSUS	SETSUS	—	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
17	CLRINTMSK	0	R/W	When this bit is set to 1, the mask of the DMA transfer end interrupt is cleared. Also, the INTMSK bit of the CHSTAT_n/nS register is set to 0. If the mask is cleared when 1 is set in both LVINT of the DCTRL register and END of the CHSTAT_n/nS register, the DMA transfer end interrupt becomes active. (It does not become active when 0 is set in LVINT.) A read operation results in 0 being read. 1: Clears the mask set by SETINTMSK. 0: Does not affect the operation.
16	SETINTMSK	0	R/W	When this bit is set to 1, the DMA transfer end interrupt is temporarily masked. Also, the INTMSK bit of the CHSTAT_n/nS register is set to 1. A read operation results in 0 being read. 1: Masks the DMA transfer end interrupt. 0: Does not affect the operation.
15 to 10	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
9	CLRSUS	0	R/W	Clear Suspend Clears the suspend status. Setting this bit to 1 when 1 is set in SUS of the CHSTAT_n/nS register can clear the suspend status. An attempt to read this bit results in 0 being read. 1: Clears the suspend status of the current DMA transfer. 0: Does not affect the operation.
8	SETSUS	0	R/W	Set Suspend Suspends the current DMA transfer. Setting this bit to 1 when 1 is set in EN of the CHSTAT_n/nS register can suspend the current DMA transfer. An attempt to read this bit results in 0 being read. 1: Suspends the current DMA transfer. 0: Does not affect the operation.
7	—	0	R	Reserved. Set 0. A read operation results in 0 being read.
6	CLRTC	0	R/W	Clear TC bit Setting this bit to 1 can clear the TC bit of the CHSTAT_n/nS register. An attempt to read this bit results in 0 being read. 1: Clears the TC bit. 0: Does not affect the operation.
5	CLREND	0	R/W	Clear End bit Setting this bit to 1 can clear the END bit of the CHSTAT_n/nS register. Also, the DMA transfer end interrupt is cleared. An attempt to read this bit results in 0 being read. 1: Clears the END bit. 0: Does not affect the operation.
4	CLRRQ	0	R/W	Clear Request bit Setting this bit to 1 can clear the RQST bit of the CHSTAT_n/nS register. An attempt to read this bit results in 0 being read. 1: Clears the RQST bit. 0: Does not affect the operation.

Bit	Bit Name	Initial Value	R/W	Description
3	SWRST	0	R/W	Software Reset Setting this bit to 1 can clear the channel status register (CHSTAT_n/nS). When setting this bit to 1, make sure that both the EN bit and TACT bit are set to 0. An attempt to read this bit results in 0 being read. 1: Resets the channel status register. 0: Does not affect the operation.
2	STG	0	R/W	Software Trigger Setting this bit to 1 sets an auto request. If this bit is set at the same time the SWRST bit is set, the clear operation by the SWRST bit takes precedence. An attempt to read this bit results in 0 being read. 1: Sets a transfer request triggered by an auto request (sets 1 in the RQST bit). 0: Does not affect the operation.
1	CLREN	0	R/W	Clear Enable Setting this bit to 1 can clear the EN bit (for details, see section 9.7.11 (2), Transfer Stop). An attempt to read this bit results in 0 being read. 1: Stops the DMA transfer (clears the EN bit). 0: Does not affect the operation.
0	SETEN	0	R/W	Set Enable Enables a DMA transfer on DMA channel n. If this bit is set at the same time the SWRST bit is set, the clear operation by the SWRST bit takes precedence and the transfer does not start. An attempt to read this bit results in 0 being read. 1: Enables a DMA transfer (sets 1 in the EN bit). 0: Does not affect the operation.

9.4.9 Channel Configuration Register n/nS (CHCFG_n/nS)

This register controls the DMA transfer operation on DMA channel n (n = 0 to 15).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMS	REN	RSW	RSEL	SBE	—	—	DEM	—	TM	DAD	SAD	DDS[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDS[3:0]				—	AM[2:0]			—	LVL	HIEN	LOEN	REQD	SEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	DMS	0	R/W	DMA Mode Select Sets the DMA mode. 0: Register mode (initial value) 1: Link mode
30	REN	0	R/W	Register Set Enable After a DMA transaction is completed, DMA transfers are continued using the Next register set selected by RSEL. This bit is valid only in register mode. 0: Does not continue DMA transfers. 1: Continues DMA transfers. Set condition(s): • When 1 is written to this bit Clear condition(s): • When 0 is written to this bit • When a DMA transaction is completed, with REN set to 1
29	RSW	0	R/W	Register Select Switch Inverts RSEL automatically after a DMA transaction is completed. This bit is valid only in register mode. 0: Does not invert RSEL automatically after a DMA transaction (initial value). 1: Inverts RSEL automatically after a DMA transaction.
28	RSEL	0	R/W	Register Set Select Selects the Next register set to be executed next. This bit is valid only in register mode. When RSW is set to 1, this bit is inverted automatically when a DMA transaction is completed. 0: Executes the Next0 Register Set (initial value). 1: Executes the Next1 Register Set. Transition condition(s): • When a DMA transaction is completed, with RSW set to 1
27	SBE	0	R/W	Sweep Buffer Enable Selects whether to sweep (write) the data already read into the buffer and stop the DMA transfer if the Enable bit is cleared to 0 during a DMA transaction. The sweep mode is available only when REQD is set to 0. 0: Stops the DMA transfer without sweeping the buffer (initial value). 1: Stops the DMA transfer after sweeping the buffer.
26, 25	—	0	R	Reserved. Set 0. A read operation results in 0 being read.
24	DEM	0	R/W	DMA Transfer End Interrupt Mask Masks the DMA transfer end interrupt for register mode transfer. If 1 is set in this bit when a DMA transfer end interrupt is output, the DMA transfer end interrupt signal is not asserted. In this case, DEM is cleared to 0 automatically. 0: Does not mask the DMA transfer end interrupt (initial value). 1: Masks the DMA transfer end interrupt. Clear condition(s): • When a DMA transaction is completed with DEM set to 1

Bit	Bit Name	Initial Value	R/W	Description
23	—	0	R	Reserved. Set 0. A read operation results in 0 being read.
22	TM	0	R/W	Transfer Mode Sets the DMA transfer mode. 0: Single transfer mode (initial value) 1: Block transfer mode
21	DAD	0	R/W	Sets the destination address counting direction of DMA channel n. 0: Increment (initial value) 1: Fixed
20	SAD	0	R/W	Sets the source address counting direction of DMA channel n. 0: Increment (initial value) 1: Fixed
19 to 16	DDS[3:0]	0000	R/W	Destination Data Size Sets the DMA transfer size of the transfer destination.

Value	Size	Remark
0000	8 bits	Initial value
0001	16 bits	
0010	32 bits	
0011	64 bits	
0100	128 bits	
0101	256 bits	
0110	512 bits	
0111	1024 bits	
Other than the above	—	Setting prohibited

15 to 12	SDS[3:0]	0000	R/W	Source Data Size Sets the DMA transfer size of the transfer source.

Value	Size	Remark
0000	8 bits	Initial value
0001	16 bits	
0010	32 bits	
0011	64 bits	
0100	128 bits	
0101	256 bits	
0110	512 bits	
0111	1024 bits	
Other than the above	—	Setting prohibited

11	—	0	R	Reserved. Set 0. A read operation results in 0 being read.
10 to 8	AM[2:0]	000	R/W	ACK Mode Sets the DMAACK output mode. 000: (initial value) 001: Level mode (active until the transfer request from an on-chip peripheral module or the external DREQ input becomes inactive) 01x: Bus cycle mode (active while the DMA transfer is in a bus cycle) 1xx: DMAACK not to be output (this setting should be made when an auto request is made by STG (CHCTRL_n/nS) or when the SCIFA transfer is performed)
7	—	0	R	Reserved. Set 0. A read operation results in 0 being read.
6	LVL	0	R/W	Level Selects whether to detect a DMA request based on the level or edge of the signal. 0: Detects based on the edge (initial value). 1: Detects based on the level.

Bit	Bit Name	Initial Value	R/W	Description
5	HIEN	0	R/W	High Enable Selects whether to detect a DMA request using the High level or rising edge of the signal. When LVL = 0: HIEN = 1: Detects a request in response to the rising edge of the signal. HIEN = 0: Does not detect a request in response to the rising edge of the signal (initial value). When LVL = 1: HIEN = 1: Detects a request when the signal is at the High level. HIEN = 0: Does not detect a request even when the signal is at the High level (initial value).
4	LOEN	0	R/W	Low Enable Selects whether to detect a DMA request using the Low level or falling edge of the signal. When LVL = 0: LOEN = 1: Detects a request in response to the falling edge of the signal. LOEN = 0: Does not detect a request in response to the falling edge of the signal (initial value). When LVL = 1: LOEN = 1: Detects a request when the signal is at the Low level. LOEN = 0: Does not detect a request even when the signal is at the Low level (initial value).
3	REQD	0	R/W	Request Direction Selects whether DMAREQ selected by the SEL bit is the source or destination. This bit is also used to define when DMAACK is to become active. 0: Source; DMAACK is to become active when read (initial value). 1: Destination; DMAACK is to become active when written.
2 to 0	SEL[2:0]	000	R/W	These bits are used to set a DMAC channel. Set one of the following values so that the channel set by the SEL bits matches the CHCFG_n/nS channel. 000: CH0/CH8 001: CH1/CH9 010: CH2/CH10 011: CH3/CH11 100: CH4/CH12 101: CH5/CH13 110: CH6/CH14 111: CH7/CH15

9.4.10 Channel Interval Register n/nS (CHITVL_n/nS)

This register sets the transfer interval for DMA channel n (n = 0 to 15).

For details, see section 9.7.9, Interval Count Function.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ITVL															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
15 to 0	ITVL	All 0	R/W	Sets the channel transfer interval.

9.4.11 Channel Extension Register n/nS (CHEXT_n/nS)

This is an extension register for DMA channel n (n = 0 to 15).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCA[3:0]				—	DPR[2:0]			SCA[3:0]				—	SPR[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

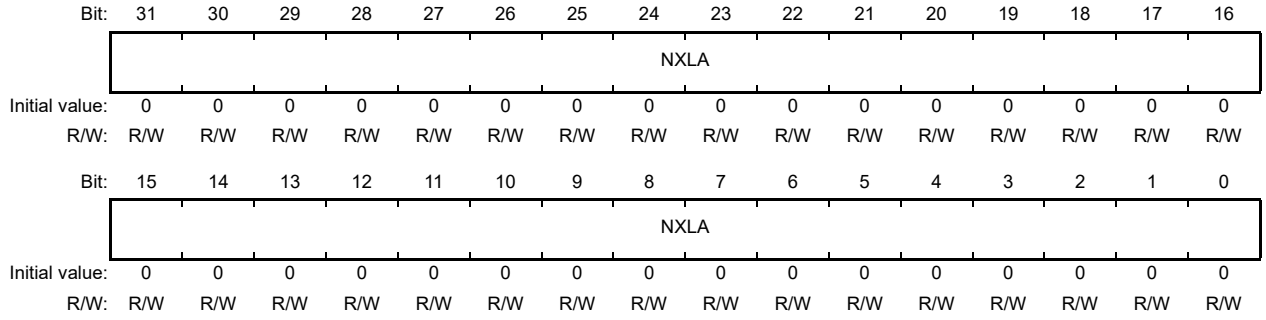
Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Set 0. A read operation results in 0 being read.
15 to 12	DCA[3:0]	0000	R/W	Destination CACHE Sets the value to be output to AWCACHE[3:0] for DMA write transfer. See Note 1 below.
11	—	0	R	Set 0. A read operation results in 0 being read.
10 to 8	DPR[2:0]	000	R/W	Destination PROT Sets the value to be output to AWPROT[2:0] for DMA write transfer. In the case of non-secure access, the value output as AWPROT[1] is fixed to 1 regardless of the setting of the DPR[1] bit. See Note 2 below.
7 to 4	SCA[3:0]	0000	R/W	Source CACHE Sets the value to be output to ARCACHE[3:0] for DMA read transfer. See Note 1 below.
3	—	0	R	Set 0. A read operation results in 0 being read.
2 to 0	SPR[2:0]	000	R/W	Source PROT Sets the value to be output to ARPROT[2:0] for DMA read transfer. In the case of non-secure access, the value output as ARPROT[1] is fixed to 1 regardless of the setting of the SPR[1] bit. See Note 2 below.

Note 1. Cache support: Bits SCA and DCA are used to change the settings.
When the transfer destination or source is not in the external bus space, set these bits to 0000.
Even when the transfer destination or source is in the external bus space but the secondary cache is not in use, set these bits to 0000. In this case, the DACK0 output and TEND0 output are issued in response to the DREQ0 transfer request.
When the secondary cache is in use in the external bus space, set CACHE[3:0].

Note 2. Protection unit support: Bits SPR and DPR are used to change the settings.
For the setting value, see AMBA AXI Protocol Specification from Arm Limited.

9.4.12 Next Link Address Register n/nS (NXLA_n/nS)

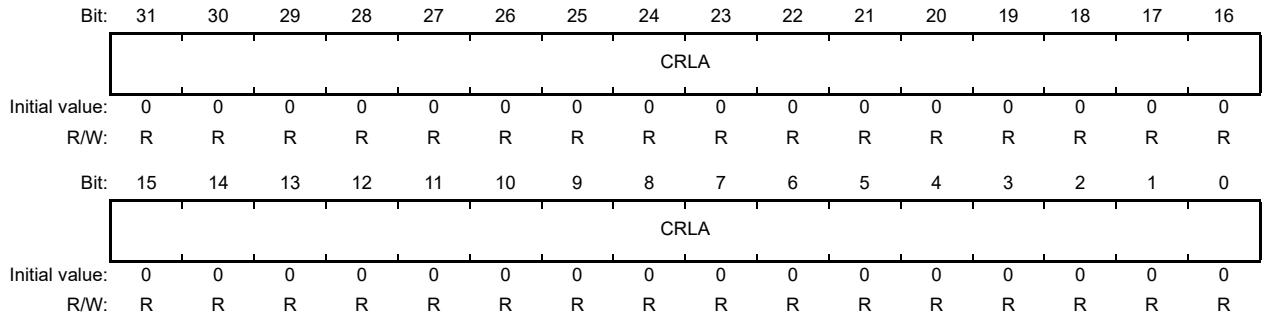
This is a 32-bit register that sets the link address of DMA channel n (n = 0 to 15).
For information about the link mode, see section 9.6.3, Link Mode.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NXLA	All 0	R/W	Sets a link address. The low-order 2 bits are masked with 0s. Only an address aligned with a 4-byte boundary can be set.

9.4.13 Current Link Address Register n/nS (CRLA_n/nS)

This is a 32-bit register that indicates the link address of DMA channel n (n = 0 to 15).
For information about the link mode, see section 9.6.3, Link Mode.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRLA	All 0	R	Indicates the address of the currently executed descriptor.

9.4.14 DMA Control Register (DCTRL_0_7/0_7S, DCTRL_8_15/8_15S)

This register sets the transfer type for descriptor access and the arbitration between channels.
(DCTRL_0_7/0_7S is common for channels 0 to 7 and DCTRL_8_15/8_15S is common for channels 8 to 15.)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LWCA				—	LWPR			LDCA				—	LDPR		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LVINT	PR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	LWCA	0000	R/W	Link WriteBack CACHE Sets the value to be output to AWCACHE[3:0] during descriptor writeback in link mode. For the setting value, see Note in section 9.4.11, Channel Extension Register n/nS (CHEXT_n/nS).
27	—	0	R	Reserved. Set 0. The initial value is 0.
26 to 24	LWPR	000	R/W	Link WriteBack PROT Sets the value to be output to AWPROT[2:0] during descriptor writeback in link mode. In the case of non-secure access, the value output as AWPROT[1] is fixed to 1 regardless of the setting of the LWPR[1] bit. For the setting value, see AMBA AXI Protocol Specification from Arm Limited.
23 to 20	LDCA	0000	R/W	Link Descriptor CACHE Sets the value to be output to ARCACHE[3:0] during descriptor load in link mode. For the setting value, see Note in section 9.4.11, Channel Extension Register n/nS (CHEXT_n/nS).
19	—	0	R	Reserved. Set 0. The initial value is 0.
18 to 16	LDPR	000	R/W	Link Descriptor PROT Sets the value to be output to ARPROT[2:0] during descriptor load in link mode. In the case of non-secure access, the value output as ARPROT[1] is fixed to 1 regardless of the setting of the LDPR[1] bit. For the setting value, see AMBA AXI Protocol Specification from Arm Limited.
15 to 2	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
1	LVINT	0	R/W	Sets whether to use pulse output or level output for the DMA transfer end interrupt and DMA error interrupt. Set pulse output for this product. 0: Pulse output (initial value) 1: Level output
0	PR	0	R/W	Sets the transfer priority control mode between channels (see section 9.7.2, Priority Control for DMA Channels). 0: Fixed priority mode (initial value) 1: Round robin mode

9.4.15 DMA Status EN Register (DSTAT_EN_0_7/0_7S)

This register indicates the EN bit status of the CHSTAT_n/nS register (n = 0 to 7).
Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	EN7	0	R	Indicates the EN bit status of DMA channel 7.
6	EN6	0	R	Indicates the EN bit status of DMA channel 6.
5	EN5	0	R	Indicates the EN bit status of DMA channel 5.
4	EN4	0	R	Indicates the EN bit status of DMA channel 4.
3	EN3	0	R	Indicates the EN bit status of DMA channel 3.
2	EN2	0	R	Indicates the EN bit status of DMA channel 2.
1	EN1	0	R	Indicates the EN bit status of DMA channel 1.
0	EN0	0	R	Indicates the EN bit status of DMA channel 0.

9.4.16 DMA Status EN Register (DSTAT_EN_8_15/8_15S)

This register indicates the EN bit status of the CHSTAT_n/nS register (n = 8 to 15).
Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	EN15	EN14	EN13	EN12	EN11	EN10	EN9	EN8
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	EN15	0	R	Indicates the EN bit status of DMA channel 15.
6	EN14	0	R	Indicates the EN bit status of DMA channel 14.
5	EN13	0	R	Indicates the EN bit status of DMA channel 13.
4	EN12	0	R	Indicates the EN bit status of DMA channel 12.
3	EN11	0	R	Indicates the EN bit status of DMA channel 11.
2	EN10	0	R	Indicates the EN bit status of DMA channel 10.
1	EN9	0	R	Indicates the EN bit status of DMA channel 9.
0	EN8	0	R	Indicates the EN bit status of DMA channel 8.

9.4.17 DMA Status ER Register (DSTAT_ER_0_7/0_7S)

This register indicates the ER bit status of the CHSTAT_n/nS register (n = 0 to 7).
Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	ER7	0	R	Indicates the ER bit status of DMA channel 7.
6	ER6	0	R	Indicates the ER bit status of DMA channel 6.
5	ER5	0	R	Indicates the ER bit status of DMA channel 5.
4	ER4	0	R	Indicates the ER bit status of DMA channel 4.
3	ER3	0	R	Indicates the ER bit status of DMA channel 3.
2	ER2	0	R	Indicates the ER bit status of DMA channel 2.
1	ER1	0	R	Indicates the ER bit status of DMA channel 1.
0	ER0	0	R	Indicates the ER bit status of DMA channel 0.

9.4.18 DMA Status ER Register (DSTAT_ER_8_15/8_15S)

This register indicates the ER bit status of the CHSTAT_n/nS register (n = 8 to 15).
Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ER15	ER14	ER13	ER12	ER11	ER10	ER9	ER8
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	ER15	0	R	Indicates the ER bit status of DMA channel 15.
6	ER14	0	R	Indicates the ER bit status of DMA channel 14.
5	ER13	0	R	Indicates the ER bit status of DMA channel 13.
4	ER12	0	R	Indicates the ER bit status of DMA channel 12.
3	ER11	0	R	Indicates the ER bit status of DMA channel 11.
2	ER10	0	R	Indicates the ER bit status of DMA channel 10.
1	ER9	0	R	Indicates the ER bit status of DMA channel 9.
0	ER8	0	R	Indicates the ER bit status of DMA channel 8.

9.4.19 DMA Status END Register (DSTAT_END_0_7/0_7S)

This register indicates the END bit status of the CHSTAT_n/nS register (n = 0 to 7).
Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	END7	END6	END5	END4	END3	END2	END1	END0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	END7	0	R	Indicates the END bit status of DMA channel 7.
6	END6	0	R	Indicates the END bit status of DMA channel 6.
5	END5	0	R	Indicates the END bit status of DMA channel 5.
4	END4	0	R	Indicates the END bit status of DMA channel 4.
3	END3	0	R	Indicates the END bit status of DMA channel 3.
2	END2	0	R	Indicates the END bit status of DMA channel 2.
1	END1	0	R	Indicates the END bit status of DMA channel 1.
0	END0	0	R	Indicates the END bit status of DMA channel 0.

9.4.20 DMA Status END Register (DSTAT_END_8_15/8_15S)

This register indicates the END bit status of the CHSTAT_n/nS register (n = 8 to 15).
Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	END15	END14	END13	END12	END11	END10	END9	END8
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	END15	0	R	Indicates the END bit status of DMA channel 15.
6	END14	0	R	Indicates the END bit status of DMA channel 14.
5	END13	0	R	Indicates the END bit status of DMA channel 13.
4	END12	0	R	Indicates the END bit status of DMA channel 12.
3	END11	0	R	Indicates the END bit status of DMA channel 11.
2	END10	0	R	Indicates the END bit status of DMA channel 10.
1	END9	0	R	Indicates the END bit status of DMA channel 9.
0	END8	0	R	Indicates the END bit status of DMA channel 8.

9.4.21 DMA Status TC Register (DSTAT_TC_0_7/0_7S)

This register indicates the TC bit status of the CHSTAT_n/nS register (n = 0 to 7).
Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	TC7	0	R	Indicates the TC bit status of DMA channel 7.
6	TC6	0	R	Indicates the TC bit status of DMA channel 6.
5	TC5	0	R	Indicates the TC bit status of DMA channel 5.
4	TC4	0	R	Indicates the TC bit status of DMA channel 4.
3	TC3	0	R	Indicates the TC bit status of DMA channel 3.
2	TC2	0	R	Indicates the TC bit status of DMA channel 2.
1	TC1	0	R	Indicates the TC bit status of DMA channel 1.
0	TC0	0	R	Indicates the TC bit status of DMA channel 0.

9.4.22 DMA Status TC Register (DSTAT_TC_8_15/8_15S)

This register indicates the TC bit status of the CHSTAT_n/nS register (n = 8 to 15).
Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	TC15	0	R	Indicates the TC bit status of DMA channel 15.
6	TC14	0	R	Indicates the TC bit status of DMA channel 14.
5	TC13	0	R	Indicates the TC bit status of DMA channel 13.
4	TC12	0	R	Indicates the TC bit status of DMA channel 12.
3	TC11	0	R	Indicates the TC bit status of DMA channel 11.
2	TC10	0	R	Indicates the TC bit status of DMA channel 10.
1	TC9	0	R	Indicates the TC bit status of DMA channel 9.
0	TC8	0	R	Indicates the TC bit status of DMA channel 8.

9.4.23 DMA Status SUS Register (DSTAT_SUS_0_7/0_7S)

This register indicates the SUS bit status of the CHSTAT_n/nS register (n = 0 to 7).
Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SUS7	SUS6	SUS5	SUS4	SUS3	SUS2	SUS1	SUS0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	SUS7	0	R	Indicates the SUS bit status of DMA channel 7.
6	SUS6	0	R	Indicates the SUS bit status of DMA channel 6.
5	SUS5	0	R	Indicates the SUS bit status of DMA channel 5.
4	SUS4	0	R	Indicates the SUS bit status of DMA channel 4.
3	SUS3	0	R	Indicates the SUS bit status of DMA channel 3.
2	SUS2	0	R	Indicates the SUS bit status of DMA channel 2.
1	SUS1	0	R	Indicates the SUS bit status of DMA channel 1.
0	SUS0	0	R	Indicates the SUS bit status of DMA channel 0.

9.4.24 DMA Status SUS Register (DSTAT_SUS_8_15/8_15S)

This register indicates the SUS bit status of the CHSTAT_n/nS register (n = 8 to 15).
Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SUS15	SUS14	SUS13	SUS12	SUS11	SUS10	SUS9	SUS8
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	SUS15	0	R	Indicates the SUS bit status of DMA channel 15.
6	SUS14	0	R	Indicates the SUS bit status of DMA channel 14.
5	SUS13	0	R	Indicates the SUS bit status of DMA channel 13.
4	SUS12	0	R	Indicates the SUS bit status of DMA channel 12.
3	SUS11	0	R	Indicates the SUS bit status of DMA channel 11.
2	SUS10	0	R	Indicates the SUS bit status of DMA channel 10.
1	SUS9	0	R	Indicates the SUS bit status of DMA channel 9.
0	SUS8	0	R	Indicates the SUS bit status of DMA channel 8.

9.4.25 DMA Extension Resource Selectors 0/0S to 7/7S (DMARS0/0S to DMARS7/7S)

DMARSn/nS (n = 0 to 7) are 32-bit readable/writable registers that specify the DMA transfer sources from peripheral modules in each channel. DMARS0/0S is for channels 0/0S and 1/1S, DMARS1/1S is for channels 2/2S and 3/3S, and so on.

Table 9.4 shows the specifiable combinations.

Some on-chip peripheral modules in this product use the same signal both for an interrupt request and for a DMA transfer request. If such a module is selected by a DMARSn/nS register, the signal works as a DMA transfer request signal and interrupt requests to the interrupt controller are masked. To enable the interrupt, clear the setting of DMARSn/nS (set all MID[7:0] and RID[1:0] to 0).

- DMARS0/0S

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH1 MID[7:0]								CH1 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH0 MID[7:0]								CH0 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS1/1S

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH3 MID[7:0]								CH3 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH2 MID[7:0]								CH2 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS2/2S

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH5 MID[7:0]								CH5 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH4 MID[7:0]								CH4 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS3/3S

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH7 MID[7:0]								CH7 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH6 MID[7:0]								CH6 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS4/4S

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH9 MID[7:0]								CH9 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH8 MID[7:0]								CH8 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS5/5S

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH11 MID[7:0]								CH11 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH10 MID[7:0]								CH10 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS6/6S

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH13 MID[7:0]								CH13 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH12 MID[7:0]								CH12 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS7/7S

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH15 MID[7:0]								CH15 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH14 MID[7:0]								CH14 RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

9.5 Operation

When there is a DMA transfer request, the DMAC starts the transfer according to the predetermined channel priority order; when the transfer end conditions are satisfied, it ends the transfer. Transfers can be requested in three modes: auto request, external request, and on-chip peripheral module request.

9.5.1 Transfer Flow

After the next source address register (N0SA_n/nS, N1SA_n/nS), next destination address register (N0DA_n/nS, N1DA_n/nS), next transaction byte register (N0TB_n/nS, N1TB_n/nS), channel control register (CHCTRL_n/nS), channel configuration register (CHCFG_n/nS), channel extension register (CHEXT_n/nS), DMA control register (DCTRL_0_7/0_7S, DCTRL_8_15/8_15S), and DMA extension resource selector (DMARSn/nS) are set for the target transfer conditions, the DMAC transfers data according to the following procedure:

1. Checks to see if transfer is enabled (EN = 0 and TACT = 0 in channel status register).
2. Clears the channel status register (set 1 in the SWRST bit of the channel control register).
3. Enables DMA transfer (set 1 in the SETEN bit of the channel control register).
4. When a transfer request comes and transfer is enabled, the DMAC transfers one transfer unit of data (depending on the DDS[3:0] and SDS[3:0] bit settings). For an auto request, the transfer begins automatically when 1 is set in the STG bit of the channel control register. The CRTB_n/nS value will be decremented by 1 for each transfer.
5. If 0 is set in the REN bit of the channel configuration register when transfer has been completed for the specified count (when CRTB_n/nS reaches 0), transfer ends normally. If the DEM bit of the channel configuration register is set to 0 at this time, a DMA transfer end interrupt is sent to the CPU. If the REN bit is 1 when CRTB_n/nS reaches 0, transfer operations are continued with the values of N0SA_n/nS, N1SA_n/nS, N0DA_n/nS, N1DA_n/nS, N0TB_n/nS, and N1TB_n/nS set by the RSEL bit of the channel configuration register until there are no more transfer requests.
6. When an address error in the DMAC is generated, the transfer is stopped. Transfers are also stopped when 1 is set in the CLREN bit of CHCTRL_n/nS.

9.5.2 DMA Transfer Requests

DMA transfer requests are basically generated in either the data transfer source or destination, but they can also be generated in external devices and on-chip peripheral modules that are neither the transfer source nor destination. Transfers can be requested in three modes: auto request, external request, and on-chip peripheral module request. External request or on-chip peripheral module request is selected by the DMARS0/0S to DMARS7/7S registers.

(1) Auto-Request Mode

When there is no transfer request signal from an external source, as in a memory-to-memory transfer or a transfer between memory and an on-chip peripheral module unable to request a transfer, auto-request mode allows the DMAC to automatically generate a transfer request signal internally. When the STG bit in channel control register n is set to 1, the transfer begins so long as the TACT bit in channel status register is 0.

(2) External Request Mode

In this mode a transfer is performed at the transfer request signal (DREQ0) of an external device of the LSI. When the DMA transfer is enabled, DMA transfer is performed upon a DREQ input.
Choose to detect DREQ0 by either the edge or level of the signal input with the LVL, HIEN, and LOEN bits in channel configuration register 0 as shown below. The source of the transfer request does not have to be the data transfer source or destination.
For the output level settings of the DACK0 and TEND0 pins, refer to section 8, Bus State Controller.

Table 9.3 Settings for External Request Detection

CHCFG_0/CHCFG_0S			
LVL	HIEN	LOEN	Detection of External Request
0	0	1	Falling edge detection
	1	0	Rising edge detection
1	0	1	Low level detection
	1	0	High level detection

When DREQ0 is accepted, the DREQ0 pin enters the request accept disabled state (non-sensitive period). After issuing an acknowledge DACK0 signal for the accepted DREQ0, the DREQ0 pin again enters the request accept enabled state.

(3) On-Chip Peripheral Module Request Mode

In this mode, the transfer is performed in response to the DMA transfer request signal from an on-chip peripheral module.
When a transfer request signal is sent in on-chip peripheral module request mode while DMA transfer is enabled, the DMA transfer is performed.
The DMA transfer request signals to be sent from on-chip peripheral modules or external pin input are listed in Table 9.4.
The transfer source or destination is fixed for some on-chip peripheral module requests. For details, see Table 9.4.

Table 9.4 On-Chip Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS *1						
				MID	RID	TM	AM[2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
OS timer channel 0	OSTM0TINT (compare match)	Arbitrary	Arbitrary	0000_1000	11	0/1	010	0	1	0	0/1	Ch0:000 Ch1:001 Ch2:010 Ch3:011 Ch4:100 Ch5:101 Ch6:110 Ch7:111 Ch8:000 Ch9:001 Ch10:010 Ch11:011 Ch12:100 Ch13:101 Ch14:110 Ch15:111
OS timer channel 1	OSTM1TINT (compare match)	Arbitrary	Arbitrary	0000_1001	11	0/1	010	0	1	0	0/1	
OS timer channel 2	OSTM2TINT (compare match)	Arbitrary	Arbitrary	0000_1010	11	0/1	010	0	1	0	0/1	
Multi-function timer pulse unit 3 channel 0	TGIA0 (input capture/compare match)	Arbitrary	Arbitrary	0001_0000	11	0/1	001	0	1	0	0/1	
	TGIB0 (input capture/compare match)	Arbitrary	Arbitrary	0001_0001	11	0/1	001	0	1	0	0/1	
	TGIC0 (input capture/compare match)	Arbitrary	Arbitrary	0001_0010	11	0/1	001	0	1	0	0/1	
	TGID0 (input capture/compare match)	Arbitrary	Arbitrary	0001_0011	11	0/1	001	0	1	0	0/1	
Multi-function timer pulse unit 3 channel 1	TGIA1 (input capture/compare match)	Arbitrary	Arbitrary	0001_0100	11	0/1	001	0	1	0	0/1	
	TGIB1 (input capture/compare match)	Arbitrary	Arbitrary	0001_0101	11	0/1	001	0	1	0	0/1	
Multi-function timer pulse unit 3 channel 2	TGIA2 (input capture/compare match)	Arbitrary	Arbitrary	0001_0110	11	0/1	001	0	1	0	0/1	
	TGIB2 (input capture/compare match)	Arbitrary	Arbitrary	0001_0111	11	0/1	001	0	1	0	0/1	

Table 9.4 On-Chip Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS *1						
				MID	RID	TM	AM[2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
Multi-function timer pulse unit 3 channel 3	TGIA3 (input capture/compare match)	Arbitrary	Arbitrary	0001_1000	11	0/1	001	0	1	0	0/1	Ch0:000 Ch1:001 Ch2:010 Ch3:011 Ch4:100 Ch5:101 Ch6:110 Ch7:111 Ch8:000 Ch9:001 Ch10:010 Ch11:011 Ch12:100 Ch13:101 Ch14:110 Ch15:111
	TGIB3 (input capture/compare match)	Arbitrary	Arbitrary	0001_1001	11	0/1	001	0	1	0	0/1	
	TGIC3 (input capture/compare match)	Arbitrary	Arbitrary	0001_1010	11	0/1	001	0	1	0	0/1	
	TGID3 (input capture/compare match)	Arbitrary	Arbitrary	0001_1011	11	0/1	001	0	1	0	0/1	
Multi-function timer pulse unit 3 channel 4	TGIA4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1100	11	0/1	001	0	1	0	0/1	
	TGIB4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1101	11	0/1	001	0	1	0	0/1	
	TGIC4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1110	11	0/1	001	0	1	0	0/1	
	TGID4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1111	11	0/1	001	0	1	0	0/1	
	TCIV4 (overflow and underflow of TCNT)	Arbitrary	Arbitrary	0010_0000	11	0/1	001	0	1	0	0/1	
Multi-function timer pulse unit 3 channel 5	TGIU5 (input capture/compare match)	Arbitrary	Arbitrary	0010_0001	11	0/1	001	0	1	0	0/1	
	TGIV5 (input capture/compare match)	Arbitrary	Arbitrary	0010_0010	11	0/1	001	0	1	0	0/1	
	TGIW5 (input capture/compare match)	Arbitrary	Arbitrary	0010_0011	11	0/1	001	0	1	0	0/1	
Multi-function timer pulse unit 3 channel 6	TGIA6 (input capture/compare match)	Arbitrary	Arbitrary	0010_0100	11	0/1	001	0	1	0	0/1	
	TGIB6 (input capture/compare match)	Arbitrary	Arbitrary	0010_0101	11	0/1	001	0	1	0	0/1	
	TGIC6 (input capture/compare match)	Arbitrary	Arbitrary	0010_0110	11	0/1	001	0	1	0	0/1	
	TGID6 (input capture/compare match)	Arbitrary	Arbitrary	0010_0111	11	0/1	001	0	1	0	0/1	
Multi-function timer pulse unit 3 channel 7	TGIA7 (input capture/compare match)	Arbitrary	Arbitrary	0010_1000	11	0/1	001	0	1	0	0/1	
	TGIB7 (input capture/compare match)	Arbitrary	Arbitrary	0010_1001	11	0/1	001	0	1	0	0/1	
	TGIC7 (input capture/compare match)	Arbitrary	Arbitrary	0010_1010	11	0/1	001	0	1	0	0/1	
	TGID7 (input capture/compare match)	Arbitrary	Arbitrary	0010_1011	11	0/1	001	0	1	0	0/1	

Table 9.4 On-Chip Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS *1						
				MID	RID	TM	AM[2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
Multi-function timer pulse unit 3 channel 7	TCIV7 (overflow and underflow of TCNT)	Arbitrary	Arbitrary	0010_1100	11	0/1	001	0	1	0	0/1	Ch0:000 Ch1:001 Ch2:010 Ch3:011 Ch4:100 Ch5:101 Ch6:110 Ch7:111 Ch8:000 Ch9:001 Ch10:010 Ch11:011 Ch12:100 Ch13:101 Ch14:110 Ch15:111
Multi-function timer pulse unit 3 channel 8	TGIA8 (input capture/compare match)	Arbitrary	Arbitrary	0010_1101	11	0/1	001	0	1	0	0/1	
	TGIB8 (input capture/compare match)	Arbitrary	Arbitrary	0010_1110	11	0/1	001	0	1	0	0/1	
	TGIC8 (input capture/compare match)	Arbitrary	Arbitrary	0010_1111	11	0/1	001	0	1	0	0/1	
	TGID8 (input capture/compare match)	Arbitrary	Arbitrary	0011_0000	11	0/1	001	0	1	0	0/1	
General PWM timer channel 0	CCMPA0 (input capture/compare match)	Arbitrary	Arbitrary	0011_0001	11	0	001	0	1	0	0/1	
	CCMPB0 (input capture/compare match)	Arbitrary	Arbitrary	0011_0010	11	0	001	0	1	0	0/1	
	CMPC0 (compare match)	Arbitrary	Arbitrary	0011_0011	11	0	001	0	1	0	0/1	
	CMPD0 (compare match)	Arbitrary	Arbitrary	0011_0100	11	0	001	0	1	0	0/1	
	CMPE0 (compare match)	Arbitrary	Arbitrary	0011_1000	11	0	001	0	1	0	0/1	
	CMPF0 (compare match)	Arbitrary	Arbitrary	0011_1001	11	0	001	0	1	0	0/1	
	ADTRGA0 (compare match)	Arbitrary	Arbitrary	0011_1010	11	0	001	0	1	0	0/1	
	ADTRGB0 (compare match)	Arbitrary	Arbitrary	0011_1011	11	0	001	0	1	0	0/1	
	OVF0 (overflow)	Arbitrary	Arbitrary	0011_1100	11	0	001	0	1	0	0/1	
	UNF0 (underflow)	Arbitrary	Arbitrary	0011_1101	11	0	001	0	1	0	0/1	
General PWM timer channel 1	CCMPA1 (input capture/compare match)	Arbitrary	Arbitrary	0011_1110	11	0	001	0	1	0	0/1	
	CCMPB1 (input capture/compare match)	Arbitrary	Arbitrary	0011_1111	11	0	001	0	1	0	0/1	
	CMPC1 (compare match)	Arbitrary	Arbitrary	0100_0000	11	0	001	0	1	0	0/1	
	CMPD1 (compare match)	Arbitrary	Arbitrary	0100_0001	11	0	001	0	1	0	0/1	
	CMPE1 (compare match)	Arbitrary	Arbitrary	0100_0101	11	0	001	0	1	0	0/1	
	CMPF1 (compare match)	Arbitrary	Arbitrary	0100_0110	11	0	001	0	1	0	0/1	
	ADTRGA1 (compare match)	Arbitrary	Arbitrary	0100_0111	11	0	001	0	1	0	0/1	
	ADTRGB1 (compare match)	Arbitrary	Arbitrary	0100_1000	11	0	001	0	1	0	0/1	
	OVF1(overflow)	Arbitrary	Arbitrary	0100_1001	11	0	001	0	1	0	0/1	
	UNF1(underflow)	Arbitrary	Arbitrary	0100_1010	11	0	001	0	1	0	0/1	

Table 9.4 On-Chip Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS *1						
				MID	RID	TM	AM[2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
General PWM timer channel 2	CCMPA2 (input capture/compare match)	Arbitrary	Arbitrary	0100_1011	11	0	001	0	1	0	0/1	Ch0:000 Ch1:001 Ch2:010 Ch3:011 Ch4:100 Ch5:101 Ch6:110 Ch7:111 Ch8:000 Ch9:001 Ch10:010 Ch11:011 Ch12:100 Ch13:101 Ch14:110 Ch15:111
	CCMPB2 (input capture/compare match)	Arbitrary	Arbitrary	0100_1100	11	0	001	0	1	0	0/1	
	CMPC2 (compare match)	Arbitrary	Arbitrary	0100_1101	11	0	001	0	1	0	0/1	
	CMPD2 (compare match)	Arbitrary	Arbitrary	0100_1110	11	0	001	0	1	0	0/1	
	CMPE2 (compare match)	Arbitrary	Arbitrary	0101_0010	11	0	001	0	1	0	0/1	
	CMPF2 (compare match)	Arbitrary	Arbitrary	0101_0011	11	0	001	0	1	0	0/1	
	ADTRGA2 (compare match)	Arbitrary	Arbitrary	0101_0100	11	0	001	0	1	0	0/1	
	ADTRGB2 (compare match)	Arbitrary	Arbitrary	0101_0101	11	0	001	0	1	0	0/1	
	OVF2 (overflow)	Arbitrary	Arbitrary	0101_0110	11	0	001	0	1	0	0/1	
	UNF2 (underflow)	Arbitrary	Arbitrary	0101_0111	11	0	001	0	1	0	0/1	
General PWM timer channel 3	CCMPA3 (input capture/compare match)	Arbitrary	Arbitrary	0101_1000	11	0	001	0	1	0	0/1	
	CCMPB3 (input capture/compare match)	Arbitrary	Arbitrary	0101_1001	11	0	001	0	1	0	0/1	
	CMPC3 (compare match)	Arbitrary	Arbitrary	0101_1010	11	0	001	0	1	0	0/1	
	CMPD3 (compare match)	Arbitrary	Arbitrary	0101_1011	11	0	001	0	1	0	0/1	
	CMPE3 (compare match)	Arbitrary	Arbitrary	0101_1111	11	0	001	0	1	0	0/1	
	CMPF3 (compare match)	Arbitrary	Arbitrary	0110_0000	11	0	001	0	1	0	0/1	
	ADTRGA3 (compare match)	Arbitrary	Arbitrary	0110_0001	11	0	001	0	1	0	0/1	
	ADTRGB3 (compare match)	Arbitrary	Arbitrary	0110_0010	11	0	001	0	1	0	0/1	
	OVF3 (overflow)	Arbitrary	Arbitrary	0110_0011	11	0	001	0	1	0	0/1	
	UNF3 (underflow)	Arbitrary	Arbitrary	0110_0100	11	0	001	0	1	0	0/1	

Table 9.4 On-Chip Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS *1						
				MID	RID	TM	AM[2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
General PWM timer channel 4	CCMPA4 (input capture/compare match)	Arbitrary	Arbitrary	0110_0101	11	0	001	0	1	0	0/1	Ch0:000 Ch1:001 Ch2:010 Ch3:011 Ch4:100 Ch5:101 Ch6:110 Ch7:111 Ch8:000 Ch9:001 Ch10:010 Ch11:011 Ch12:100 Ch13:101 Ch14:110 Ch15:111
	CCMPB4 (input capture/compare match)	Arbitrary	Arbitrary	0110_0110	11	0	001	0	1	0	0/1	
	CMPC4 (compare match)	Arbitrary	Arbitrary	0110_0111	11	0	001	0	1	0	0/1	
	CMPD4 (compare match)	Arbitrary	Arbitrary	0110_1000	11	0	001	0	1	0	0/1	
	CMPE4 (compare match)	Arbitrary	Arbitrary	0110_1100	11	0	001	0	1	0	0/1	
	CMPF4 (compare match)	Arbitrary	Arbitrary	0110_1101	11	0	001	0	1	0	0/1	
	ADTRGA4 (compare match)	Arbitrary	Arbitrary	0110_1110	11	0	001	0	1	0	0/1	
	ADTRGB4 (compare match)	Arbitrary	Arbitrary	0110_1111	11	0	001	0	1	0	0/1	
	OVF4 (overflow)	Arbitrary	Arbitrary	0111_0000	11	0	001	0	1	0	0/1	
	UNF4 (underflow)	Arbitrary	Arbitrary	0111_0001	11	0	001	0	1	0	0/1	
General PWM timer channel 5	CCMPA5 (input capture/compare match)	Arbitrary	Arbitrary	0111_0010	11	0	001	0	1	0	0/1	
	CCMPB5 (input capture/compare match)	Arbitrary	Arbitrary	0111_0011	11	0	001	0	1	0	0/1	
	CMPC5 (compare match)	Arbitrary	Arbitrary	0111_0100	11	0	001	0	1	0	0/1	
	CMPD5 (compare match)	Arbitrary	Arbitrary	0111_0101	11	0	001	0	1	0	0/1	
	CMPE5 (compare match)	Arbitrary	Arbitrary	0111_1001	11	0	001	0	1	0	0/1	
	CMPF5 (compare match)	Arbitrary	Arbitrary	0111_1010	11	0	001	0	1	0	0/1	
	ADTRGA5 (compare match)	Arbitrary	Arbitrary	0111_1011	11	0	001	0	1	0	0/1	
	ADTRGB5 (compare match)	Arbitrary	Arbitrary	0111_1100	11	0	001	0	1	0	0/1	
	OVF5 (overflow)	Arbitrary	Arbitrary	0111_1101	11	0	001	0	1	0	0/1	
	UNF5 (underflow)	Arbitrary	Arbitrary	0111_1110	11	0	001	0	1	0	0/1	

Table 9.4 On-Chip Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS *1						
				MID	RID	TM	AM[2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
General PWM timer channel 6	CCMPA6 (input capture/compare match)	Arbitrary	Arbitrary	0111_1111	11	0	001	0	1	0	0/1	Ch0:000 Ch1:001 Ch2:010 Ch3:011 Ch4:100 Ch5:101 Ch6:110 Ch7:111 Ch8:000 Ch9:001 Ch10:010 Ch11:011 Ch12:100 Ch13:101 Ch14:110 Ch15:111
	CCMPB6 (input capture/compare match)	Arbitrary	Arbitrary	1000_0000	11	0	001	0	1	0	0/1	
	CMPC6 (compare match)	Arbitrary	Arbitrary	1000_0001	11	0	001	0	1	0	0/1	
	CMPD6 (compare match)	Arbitrary	Arbitrary	1000_0010	11	0	001	0	1	0	0/1	
	CMPE6 (compare match)	Arbitrary	Arbitrary	1000_0110	11	0	001	0	1	0	0/1	
	CMPF6 (compare match)	Arbitrary	Arbitrary	1000_0111	11	0	001	0	1	0	0/1	
	ADTRGA6 (compare match)	Arbitrary	Arbitrary	1000_1000	11	0	001	0	1	0	0/1	
	ADTRGB6 (compare match)	Arbitrary	Arbitrary	1000_1001	11	0	001	0	1	0	0/1	
	OVF6 (overflow)	Arbitrary	Arbitrary	1000_1010	11	0	001	0	1	0	0/1	
	UNF6 (underflow)	Arbitrary	Arbitrary	1000_1011	11	0	001	0	1	0	0/1	
General PWM timer channel 7	CCMPA7 (input capture/compare match)	Arbitrary	Arbitrary	1000_1100	11	0	001	0	1	0	0/1	
	CCMPB7 (input capture/compare match)	Arbitrary	Arbitrary	1000_1101	11	0	001	0	1	0	0/1	
	CMPC7 (compare match)	Arbitrary	Arbitrary	1000_1110	11	0	001	0	1	0	0/1	
	CMPD7 (compare match)	Arbitrary	Arbitrary	1000_1111	11	0	001	0	1	0	0/1	
	CMPE7 (compare match)	Arbitrary	Arbitrary	1001_0011	11	0	001	0	1	0	0/1	
	CMPF7 (compare match)	Arbitrary	Arbitrary	1001_0100	11	0	001	0	1	0	0/1	
	ADTRGA7 (compare match)	Arbitrary	Arbitrary	1001_0101	11	0	001	0	1	0	0/1	
	ADTRGB7 (compare match)	Arbitrary	Arbitrary	1001_0110	11	0	001	0	1	0	0/1	
	OVF7 (overflow)	Arbitrary	Arbitrary	1001_0111	11	0	001	0	1	0	0/1	
	UNF7 (underflow)	Arbitrary	Arbitrary	1001_1000	11	0	001	0	1	0	0/1	
A/D converter	S12ADI0 (completion of scanning)	ADDRnn = 0 to 7	Arbitrary	1001_1001	11	0/1	001	0	1	0	0	
	S12GBADI0 (completion of scanning)	ADDRnn = 0 to 7	Arbitrary	1001_1010	11	0/1	001	0	1	0	0	
	S12GCADI0 (completion of scanning)	ADDRnn = 0 to 7	Arbitrary	1001_1011	11	0/1	001	0	1	0	0	
Serial sound interface channel 0	INT_ssif_dma_rx_0 (Receive data full)	SSIFRDR_0	Arbitrary	1001_1100	10	0	010	0	1	0	0	
	INT_ssif_dma_tx_0 (Transmit data empty)	Arbitrary	SSIFTDR_0	1001_1100	01	0	010	0	1	0	1	

Table 9.4 On-Chip Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS *1						
				MID	RID	TM	AM[2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
Serial sound interface channel 1	INT_ssif_dma_rx_1 (Receive data full)	SSIFRDR_1	Arbitrary	1001_1101	10	0	010	0	1	0	0	Ch0:000 Ch1:001 Ch2:010 Ch3:011 Ch4:100 Ch5:101 Ch6:110 Ch7:111 Ch8:000 Ch9:001 Ch10:010 Ch11:011 Ch12:100 Ch13:101 Ch14:110 Ch15:111
	INT_ssif_dma_tx_1 (Transmit data empty)	Arbitrary	SSIFTDR_1	1001_1101	01	0	010	0	1	0	1	
Serial sound interface channel 2	INT_ssif_dma_rt_2 (Receive data full)	SSIFRDR_2	Arbitrary	1001_1110	11	0	010	0	1	0	0/1	
	INT_ssif_dma_rt_2 (Transmit data empty)	Arbitrary	SSIFTDR_2	1001_1110	11	0	010	0	1	0	0/1	
Serial sound interface channel 3	INT_ssif_dma_rx_3 (Receive data full)	SSIFRDR_3	Arbitrary	1001_1111	10	0	010	0	1	0	0	
	INT_ssif_dma_tx_3 (Transmit data empty)	Arbitrary	SSIFTDR_3	1001_1111	01	0	010	0	1	0	1	
Renesas SPDIF interface	SPDIFRXI (Transmitter DMA Transfer)	RDAD	Arbitrary	1010_0000	11	0	010	1	1	0	0	
	SPDIFTXI (Receiver DMA Transfer)	Arbitrary	TDAD	1010_0001	11	0	010	1	1	0	1	
I2C bus interface channel 0	INTRIICRI0 (Receive data full)	RIIC0DRR	Arbitrary	1010_0010	10	0	010	0	1	0	0	
	INTRIICTI0 (Transmit data empty)	Arbitrary	RIIC0DRT	1010_0010	01	0	010	0	1	0	1	
I2C bus interface channel 1	INTRIICRI1 (Receive data full)	RIIC1DRR	Arbitrary	1010_0011	10	0	010	0	1	0	0	
	INTRIICTI1 (Transmit data empty)	Arbitrary	RIIC1DRT	1010_0011	01	0	010	0	1	0	1	
I2C bus interface channel 2	INTRIICRI2 (Receive data full)	RIIC2DRR	Arbitrary	1010_0100	10	0	010	0	1	0	0	
	INTRIICTI2 (Transmit data empty)	Arbitrary	RIIC2DRT	1010_0100	01	0	010	0	1	0	1	
I2C bus interface channel 3	INTRIICRI3 (Receive data full)	RIIC3DRR	Arbitrary	1010_0101	10	0	010	0	1	0	0	
	INTRIICTI3 (Transmit data empty)	Arbitrary	RIIC3DRT	1010_0101	01	0	010	0	1	0	1	
FIFO on-chip serial communication interface channel 0	RXI0 (Receive FIFO data full)	FRDR0	Arbitrary	1010_0110	10	0	100	1	1	0	0	
	TXI0 (Transmit FIFO data empty)	Arbitrary	FTDR0	1010_0110	01	0	100	1	1	0	1	
FIFO on-chip serial communication interface channel 1	RXI1 (Receive FIFO data full)	FRDR1	Arbitrary	1010_0111	10	0	100	1	1	0	0	
	TXI1 (Transmit FIFO data empty)	Arbitrary	FTDR1	1010_0111	01	0	100	1	1	0	1	
FIFO on-chip serial communication interface channel 2	RXI2 (Receive FIFO data full)	FRDR2	Arbitrary	1010_1000	10	0	100	1	1	0	0	
	TXI2 (Transmit FIFO data empty)	Arbitrary	FTDR2	1010_1000	01	0	100	1	1	0	1	
FIFO on-chip serial communication interface channel 3	RXI3 (Receive FIFO data full)	FRDR3	Arbitrary	1010_1001	10	0	100	1	1	0	0	
	TXI3 (Transmit FIFO data empty)	Arbitrary	FTDR3	1010_1001	01	0	100	1	1	0	1	
FIFO on-chip serial communication interface channel 4	RXI4 (Receive FIFO data full)	FRDR4	Arbitrary	1010_1010	10	0	100	1	1	0	0	
	TXI4 (Transmit FIFO data empty)	Arbitrary	FTDR4	1010_1010	01	0	100	1	1	0	1	

Table 9.4 On-Chip Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS *1						
				MID	RID	TM	AM[2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
CANFD interface	RXF_DMA0	RSCFD0CFDRFDF0_0	Arbitrary	1010_1011	11	0	001	0	1	0	0/1	Ch0:000 Ch1:001 Ch2:010 Ch3:011 Ch4:100 Ch5:101 Ch6:110 Ch7:111 Ch8:000 Ch9:001 Ch10:010 Ch11:011 Ch12:100 Ch13:101 Ch14:110 Ch15:111
	RXF_DMA1	RSCFD0CFDRFDF0_1	Arbitrary	1010_1100	11	0	001	0	1	0	0/1	
	RXF_DMA2	RSCFD0CFDRFDF0_2	Arbitrary	1010_1101	11	0	001	0	1	0	0/1	
	RXF_DMA3	RSCFD0CFDRFDF0_3	Arbitrary	1010_1110	11	0	001	0	1	0	0/1	
	RXF_DMA4	RSCFD0CFDRFDF0_4	Arbitrary	1010_1111	11	0	001	0	1	0	0/1	
	RXF_DMA5	RSCFD0CFDRFDF0_5	Arbitrary	1011_0000	11	0	001	0	1	0	0/1	
	RXF_DMA6	RSCFD0CFDRFDF0_6	Arbitrary	1011_0001	11	0	001	0	1	0	0/1	
	RXF_DMA7	RSCFD0CFDRFDF0_7	Arbitrary	1011_0010	11	0	001	0	1	0	0/1	
	COM_DMA0 (Transmission and Reception FIFO receive mode only)	RSCFD0CFDCFDF0_0	Arbitrary	1011_0011	11	0	001	0	1	0	0/1	
	COM_DMA1 (Transmission and Reception FIFO receive mode only)	RSCFD0CFDCFDF0_3	Arbitrary	1011_0100	11	0	001	0	1	0	0/1	
Renesas serial peripheral interface channel 0	SPRI0 (Receive buffer full)	SPDR0	Arbitrary	1011_0101	10	0	010	1	1	0	0	
	SPTI0 (Transmit buffer empty)	Arbitrary	SPDR0	1011_0101	01	0	010	1	1	0	1	
Renesas serial peripheral interface channel 1	SPRI1 (Receive buffer full)	SPDR1	Arbitrary	1011_0110	10	0	010	1	1	0	0	
	SPTI1 (Transmit buffer empty)	Arbitrary	SPDR1	1011_0110	01	0	010	1	1	0	1	
Renesas serial peripheral interface channel 2	SPRI2 (Receive buffer full)	SPDR2	Arbitrary	1011_0111	10	0	010	1	1	0	0	
	SPTI2 (Transmit buffer empty)	Arbitrary	SPDR2	1011_0111	01	0	010	1	1	0	1	
Serial communication interface channel 0	RXI0 (Receive FIFO data full)	RDR0	Arbitrary	1011_1000	10	0	010	0	1	0	0	
	TXI0 (Transmit FIFO data empty)	Arbitrary	TDR0	1011_1000	01	0	010	0	1	0	1	
Serial communication interface channel 1	RXI1 (Receive FIFO data full)	RDR1	Arbitrary	1011_1001	10	0	010	0	1	0	0	
	TXI1 (Transmit FIFO data empty)	Arbitrary	TDR1	1011_1001	01	0	010	0	1	0	1	
Ethernet MAC controller	IPLS (pulse timer output detection)	Arbitrary	Arbitrary	1011_1010	11	0	001	0	1	0	0/1	

Table 9.4 On-Chip Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS *1						
				MID	RID	TM	AM[2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
Trusted Secure IP *2	RDRDY1	TSIP_REG	Arbitrary	1011_1011	11	0	001	0	1	0	0/1	Ch0:000 Ch1:001 Ch2:010 Ch3:011 Ch4:100 Ch5:101 Ch6:110 Ch7:111 Ch8:000 Ch9:001 Ch10:010 Ch11:011 Ch12:100 Ch13:101 Ch14:110 Ch15:111
	RDRDY0	TSIP_REG	Arbitrary	1011_1100	11	0	001	0	1	0	0/1	
	WRRDY4	Arbitrary	TSIP_REG	1011_1101	11	0	001	0	1	0	0/1	
	WRRDY1	Arbitrary	TSIP_REG	1011_1111	11	0	001	0	1	0	0/1	
	WRRDY0	Arbitrary	TSIP_REG	1111_0000	11	0	001	0	1	0	0/1	
	IRDRDY	TSIP_REG	Arbitrary	1111_0001	11	0	001	0	1	0	0/1	
	IWRRDY	Arbitrary	TSIP_REG	1111_0010	11	0	001	0	1	0	0/1	
Dynamic Reconfigurable Processor (DRP) *3	PAE0 (Tile 0 Programable Almost Empty)	FIFO_ DATA0	Arbitrary	1111_0011	10	0	010	1	1	0	0	
	PAF0 (Tile 0 Programable Almost Full)	Arbitrary	FIFO_ DATA0	1111_0011	01	0	010	1	1	0	1	
	PAE1 (Tile 1 Programable Almost Empty)	FIFO_ DATA1	Arbitrary	1111_0100	10	0	010	1	1	0	0	
	PAF1 (Tile 1 Programable Almost Full)	Arbitrary	FIFO_ DATA1	1111_0100	01	0	010	1	1	0	1	
	PAE2 (Tile 2 Programable Almost Empty)	FIFO_ DATA2	Arbitrary	1111_0101	10	0	010	1	1	0	0	
	PAF2 (Tile 2 Programable Almost Full)	Arbitrary	FIFO_ DATA2	1111_0101	01	0	010	1	1	0	1	
	PAE3 (Tile 3 Programable Almost Empty)	FIFO_ DATA3	Arbitrary	1111_0110	10	0	010	1	1	0	0	
	PAF3 (Tile 3 Programable Almost Full)	Arbitrary	FIFO_ DATA3	1111_0110	01	0	010	1	1	0	1	
	PAE4 (Tile 4 Programable Almost Empty)	FIFO_ DATA4	Arbitrary	1111_0111	10	0	010	1	1	0	0	
	PAF4 (Tile 4 Programable Almost Full)	Arbitrary	FIFO_ DATA4	1111_0111	01	0	010	1	1	0	1	
	PAE5 (Tile 5 Programable Almost Empty)	FIFO_ DATA5	Arbitrary	1111_1000	10	0	010	1	1	0	0	
	PAF5 (Tile 5 Programable Almost Full)	Arbitrary	FIFO_ DATA5	1111_1000	01	0	010	1	1	0	1	
External request	DREQ0	Arbitrary	Arbitrary	0000_0000	11	0/1	001/ 010/ 100	001: Falling edge detection 010: Rising edge detection 101: Low level detection 110: High level detection			0/1	000

Note 1. CHCFG_n/nS setting value
- TM 0: Single transfer
  1: Block transfer
- AM 001: ACK level output
  010: ACK bus cycle output
  100: No ACK
- LVL 0: REQ edge detection
  1: REQ level detection
- REQD 0: ACK output at read
  1: ACK output at write

Note 2. Only in products with a Trusted Secure IP
Note 3. Only in products with a DRP

9.6 DMA Mode

9.6.1 Mode Setting

The DMS field of the CHCFG_n/nS register can be used to toggle between register mode and link mode.

Table 9.5 DMA Mode Setting

DMS (CHCFG_n/nS)	Mode	Description
0	Register mode	A DMA transfer is executed using the values set in the Next Register Set.
1	Link mode	A DMA transfer is executed using the descriptor set in the Current register. The DMAC repeatedly loads the descriptor and executes the DMA transfer unless otherwise set by the descriptor or stopped by the control register.

9.6.2 Register Mode

In register mode, a DMA transfer is executed using the values set in the internal registers.
Two sets of the source address, destination address, and transfer byte count (Next0 Register Set and Next1 Register Set) can be set. It is possible to select the Next register set to be used for the DMA transfer, as well as to execute two Next register sets continuously for the DMA transfer.

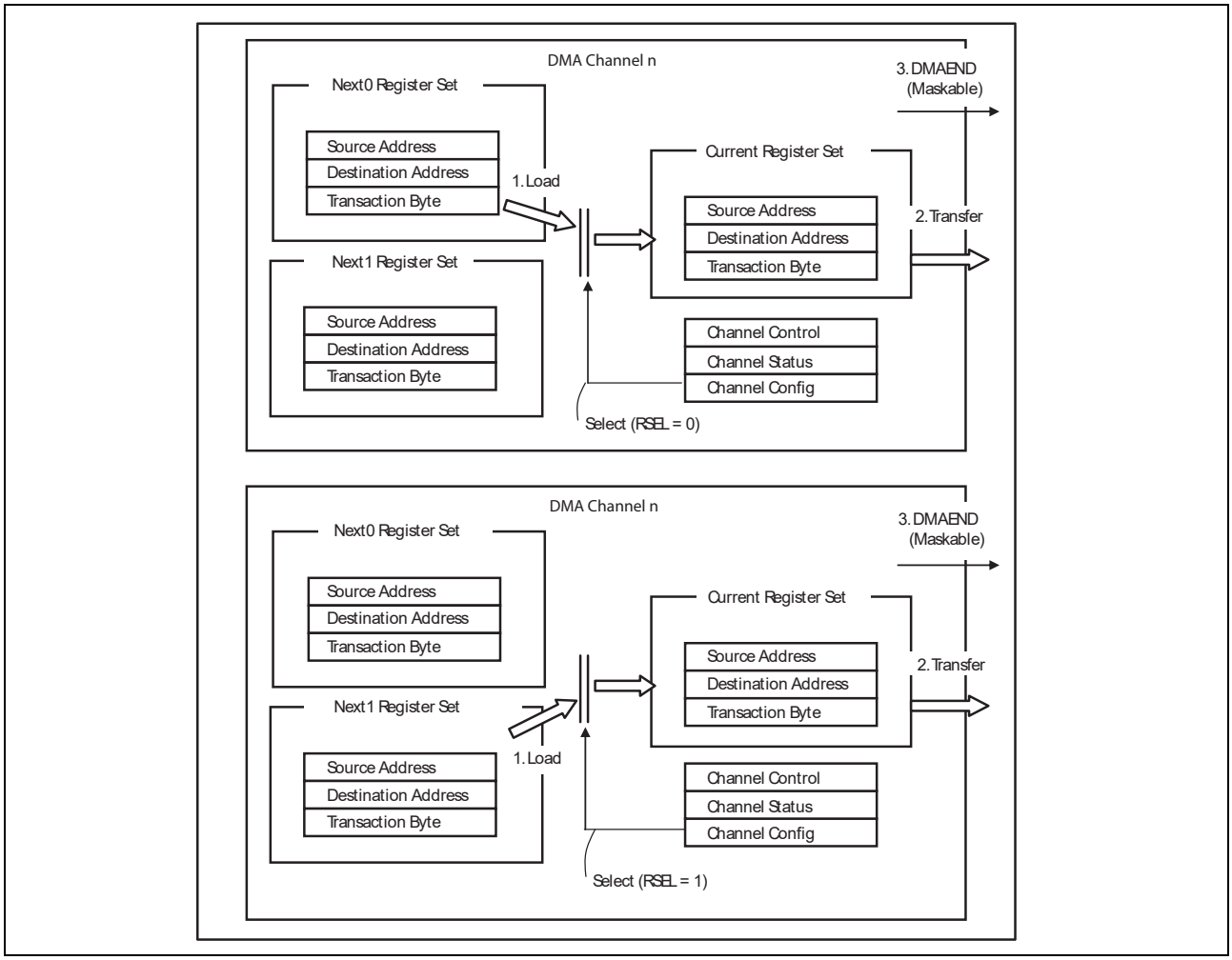


Figure 9.2 Outline of Normal Register Mode

The above figure shows how the transfer is executed when the Next0 Register Set is used (upper part of the figure) and when the Next1 Register Set is used (lower part of the figure).

(1) Operation Flow

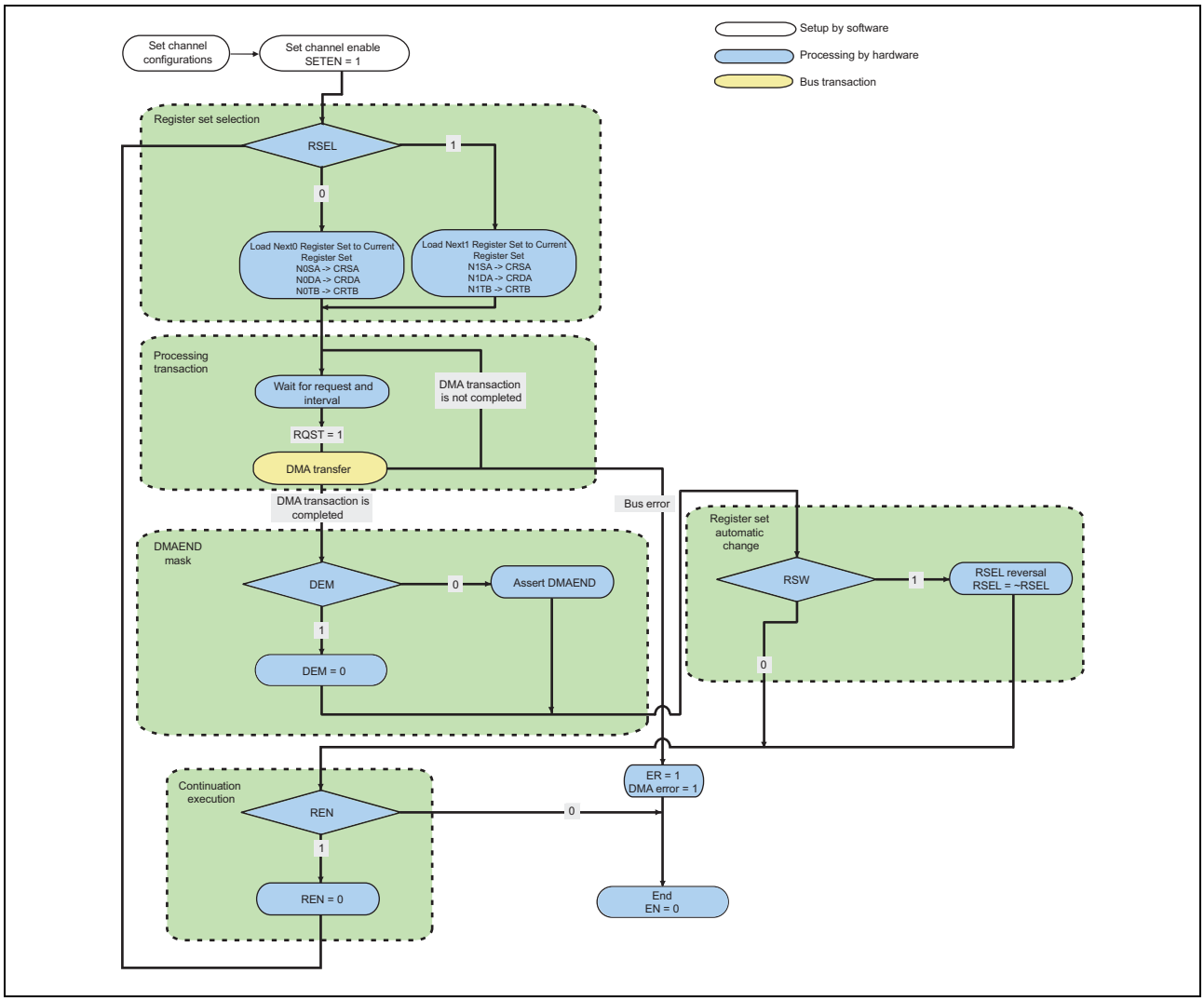


Figure 9.3 Register Mode Flow

<Explanation of the register mode flow>

1. Channel setting (set channel configuration)
   The Next0 or Next1 register set (destination address, source address, and total transfer byte count) is set.
   In the Channel register set, the DMA register set (REQ, DMAACK, transfer size, etc.) is set. (See section 9.7, DMA Transfer.)

2. Register set selection (register set selection)
   When 1 is set in EN, the values set in the Next register set selected by RSEL are loaded to the Current register set.

3. DMA transaction (processing transaction)
   A DMA transfer is executed according to the set values. For details of the transfer, see section 9.7, DMA Transfer.

4. DMA transfer end interrupt mask (DMAINT mask)
   The DMA transfer end interrupt is masked according to the value set in the DEM bit of CHCFG_n/nS.
   When 1 is set in DEM, the DMA transfer end interrupt is not output. Also, immediately after that, DEM is automatically cleared to 0.

5. Automatic register set change (register set automatic change)
   Whether to use the other Next register set is determined by the value set in the RSW bit of CHCFG_n/nS.

6. Continuation of execution (continuation execution)
   Whether to continue the execution of the DMA transfer is determined by the value set in the REN bit of CHCFG_n/nS. When 1 is set in REN, the execution of the DMA transfer is continued. Also, immediately after that, REN is automatically cleared to 0.

(2) Register Setting

(a) Register mode setting

Select the register set to be executed.

Table 9.6 Register Mode Setting

DMS (CHCFG_n/nS)	RSEL (CHCFG_n/nS)	Description
0	0	Executes the Next0 Register Set.
	1	Executes the Next1 Register Set.

(b) DMA transfer end interrupt mask setting

The DMA transfer end interrupt can be masked individually for each register set.

Table 9.7 DMAINT Mask Setting

DEM (CHCFG_n/nS)	Operation
0	When the DMA transaction is completed, a DMA transfer end interrupt is issued.
1	Even when the DMA transaction is completed, a DMA transfer end interrupt is not issued. After the DMA transaction is completed, DEM is cleared to 0 by hardware.

(c) Automatic register set execution setting

After DMA transfers, the DMA transaction of the selected register set is automatically executed.

Table 9.8 Automatic Register Set Execution Setting

REN (CHCFG_n/nS)	Operation	Remark
0	When the DMA transaction of the register set selected by RSEL is completed, the EN bit is cleared and the DMA operation ends.	Set this value to execute a DMA transaction once.
1	When a DMA transaction is completed, the DMAC continues to execute a DMA transfer by using the data set in the selected register set. When continuous transfers are successful, REN is cleared to 0.	Set this value to continuously execute DMA transfers by using the data set in separate register sets.

(d) Automatic register set change setting

When 1 is set in REN, the DMAC can automatically change to the register set to be executed next, after a DMA transaction is completed.

Table 9.9 Automatic Register Set Change Setting

RSW (CHCFG_n/nS)	Operation	Remark
0	If 1 is set in REN when a DMA transaction is completed, the register set is not changed.	Set this value to use only one register set.
1	If 1 is set in REN when a DMA transaction is completed, the value of RSEL is automatically inverted and the other register set is selected.	Set this value to change the register set.

(3) Setting Examples

(a) When only the Next0 register set is used

Table 9.10 Register Mode Setting Example 1

DMS (CHCFG_n/nS)	RSEL (CHCFG_n/nS)	DEM (CHCFG_n/nS)	RSW (CHCFG_n/nS)	REN (CHCFG_n/nS)
0 (Register mode)	0 (Next0)	0 (not masked)	0 (not switched)	0 (not continuously executed)

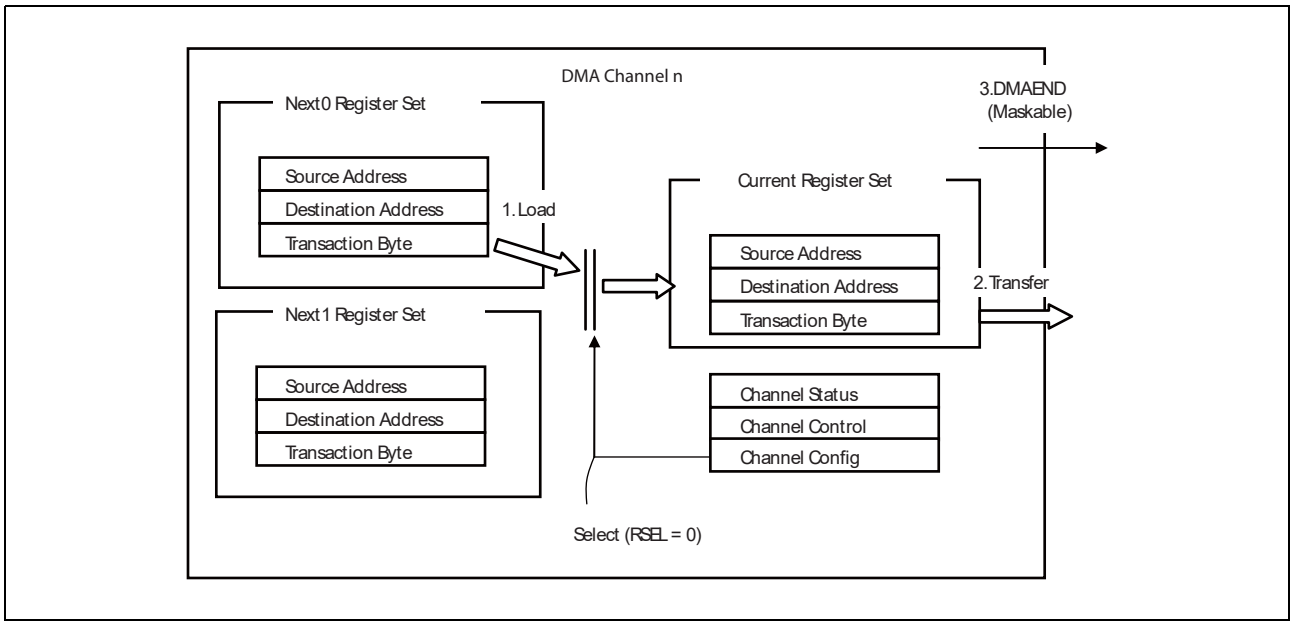


Figure 9.4 Register Mode Setting Example 1

1. 1 is set in EN (SETEN = 1), and the Next0 register set is loaded to the Current register set.
2. A DMA transaction is executed according to the values set in the Current register set and Channel register set.
3. Because 0 is set in DEM, the DMA transfer end interrupt is issued after the DMA transaction is completed.
4. Because 0 is set in REN, EN is cleared to 0 and the DMA transaction ends.

(b) When two register sets are used continuously

Table 9.11 Automatic Register Set Execution Setting

DMS (CHCFG_n/nS)	RSEL (CHCFG_n/nS)	DEM (CHCFG_n/nS)	RSW (CHCFG_n/nS)	REN (CHCFG_n/nS)
0 (Register mode)	0 (Next0)	1 (masked)	1 (switched)	1 (continuously executed)

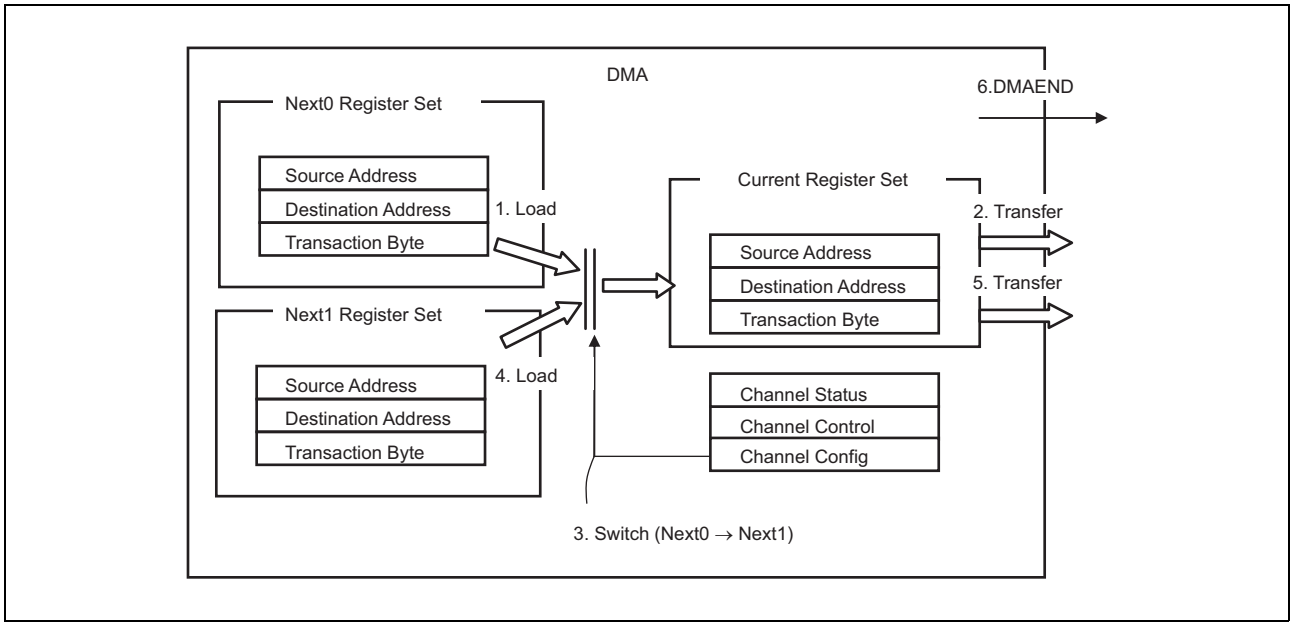


Figure 9.5 Register Mode Setting Example 2

1. 1 is set in EN (SETEN = 1), and the Next0 register set is loaded to the Current register set.
2. A DMA transaction is executed according to the values set in the Current register set and Channel register set.
3. Because 1 is set in DEM, DMA transfer end interrupt is not output after the DMA transaction is completed. Also, DEM is automatically cleared to 0.
4. Because 1 is set in REN, the execution is continued. Also, REN is automatically cleared to 0.
5. Because 1 is set in RSW, the register set to be executed next is switched (RSEL = 0 → 1).
6. The Next1 register set is loaded to the Current register set.
7. A DMA transaction is executed according to the values set in the Current register set and Channel register set.
8. Because 0 is set in DEM, the DMA transfer end interrupt is issued after the DMA transaction is completed.
9. Because 0 is set in REN, EN is cleared to 0 and the DMA transaction ends.

9.6.3 Link Mode

In link mode, a descriptor stored in external memory is loaded as set values and a DMA transaction is executed using the loaded values. The DMAC contains a Next Link address and a Current Link address for each channel, and these addresses are used to set the descriptor address to be executed next and to indicate the descriptor address of the currently executed DMA transaction, respectively.

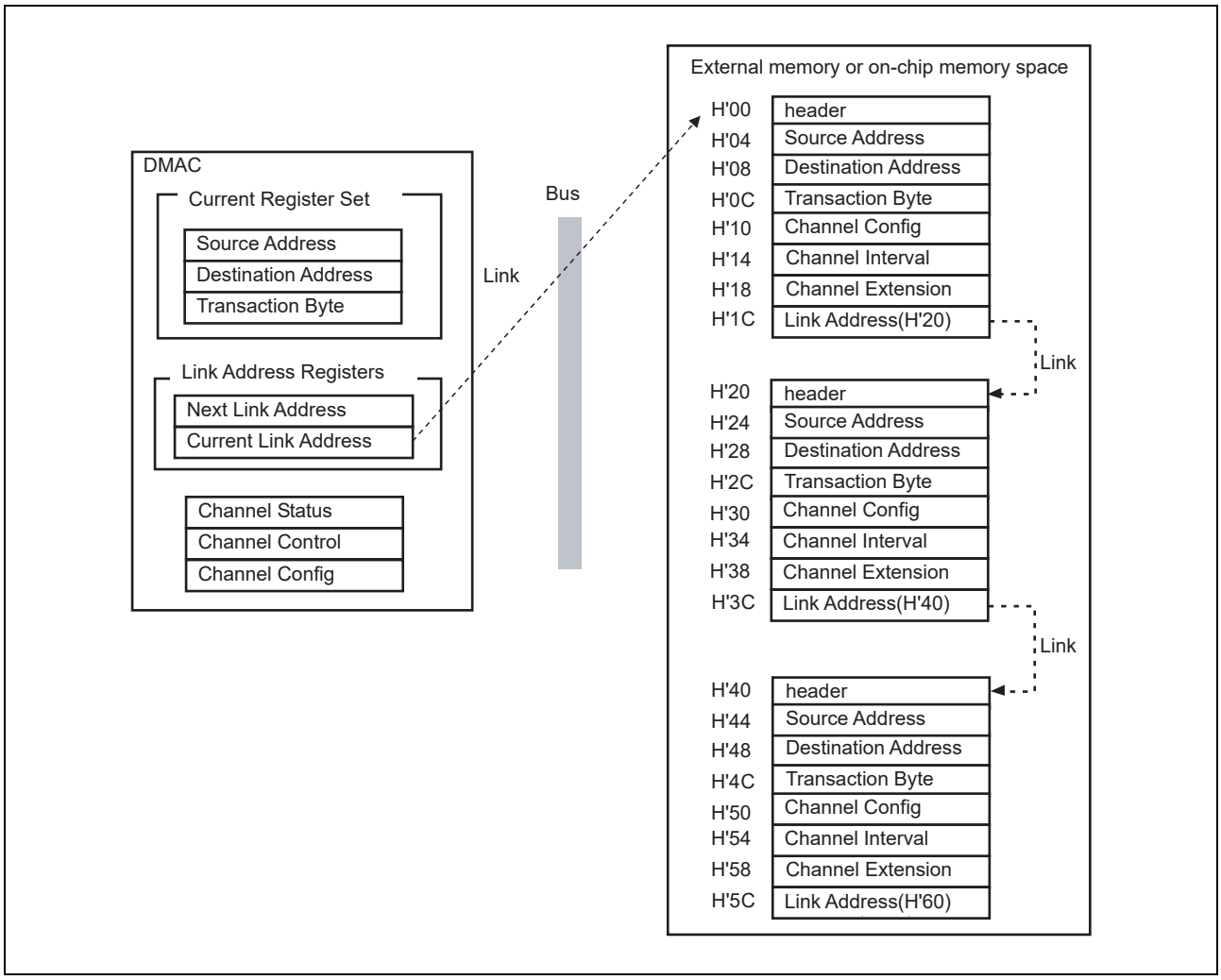


Figure 9.6 Link Mode Outline

(1) Operation Flow

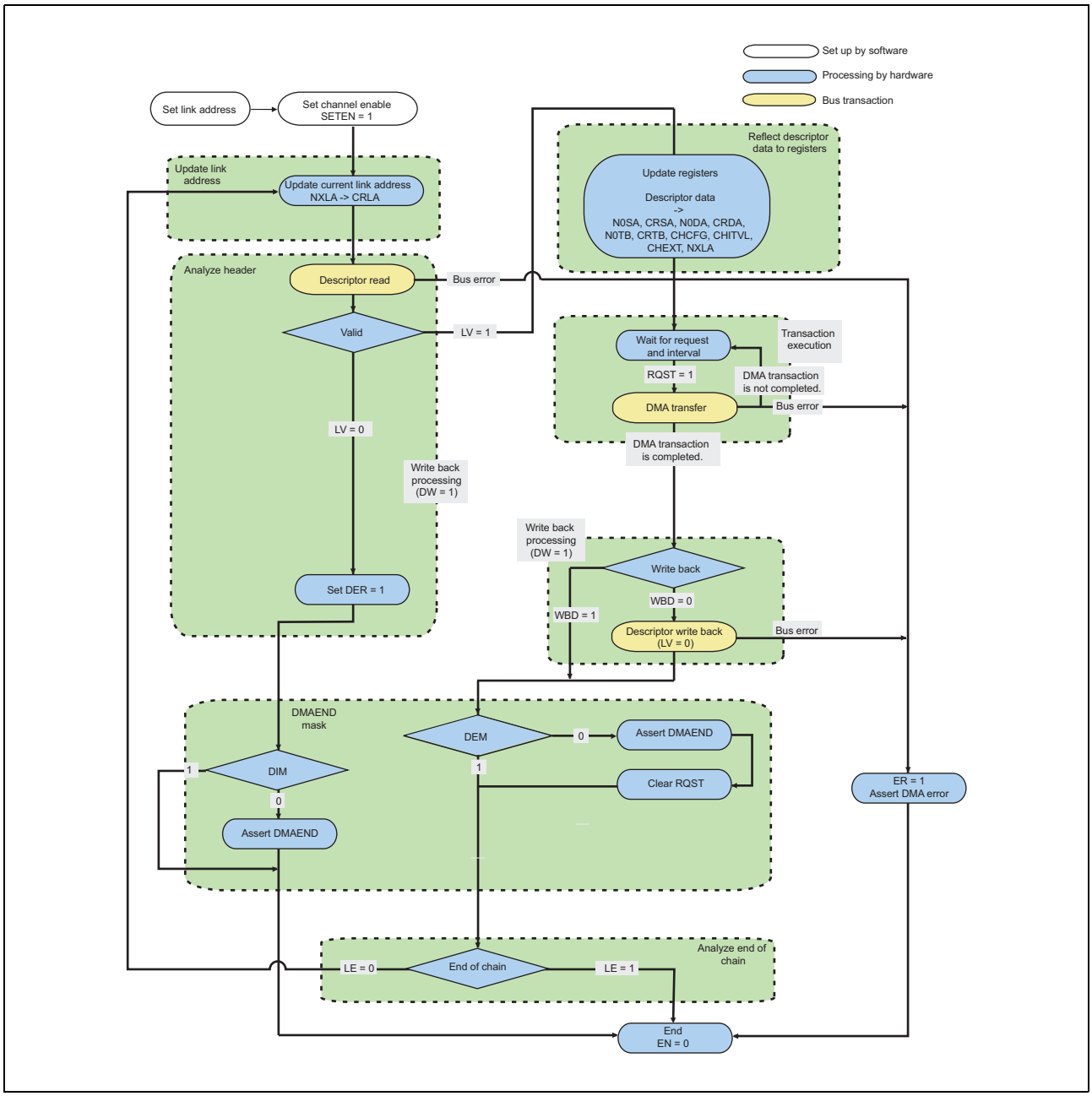


Figure 9.7 Link Mode Flow

<Explanation of the link mode flow>

1. Channel setting
   The start address of the link destination is set in NXLA_n/nS.

2. Link address update
   When 1 is set in EN (1 is set in SETEN), the Link address set in NXLA_n/nS is loaded to CRLA_n/nS.

3. Descriptor load and header analysis
   The DMAC begins to load the descriptor and then analyzes the content of the header. When LV is 0, the DMAC discards the loaded descriptor and sets 1 in DER to end the operation (EN = 0). In this case, if 0 is set in DIM of the header, DMAEND is issued.

4. Descriptor setting
   The loaded descriptor is set in the Current register set and Channel register set. Also, the next link address is set in NXLA_n/nS.

5. DMA transaction
   A DMA transaction is executed according to the set values.

6. Header writeback
   When 0 is set in WBD of the header, the DMAC writes back the header with 0 set in its LV bit.

7. DMAINT mask
   When 0 is set in the DEM bit of CHCFG_n/nS, the DMA transfer end interrupt is issued.

8. Link end analysis
   When 1 is set in LE of the header, the operation is ended by clearing EN to 0, after transfer using the settings of the descriptor is completed. If the setting of LE is 0, the Current registers are then updated and loading of the next descriptor begins. The TEND signal is issued after the transfer of each descriptor.

(2) Register Setting

(a) Link mode setting

To use the link mode, set 1 in the DMS bit of the CHCFG_n/nS register.

Table 9.12 Link Mode Setting

DMS (CHCFG_n/nS)	Description
1	Operates in link mode. This bit cannot be changed using a descriptor.

(b) Link address setting

There are two registers that indicate a link address: Next Link address register and Current Link address register.
To start the link mode, set a link address in the Next Link address register.
The Next Link address indicates the next link address after a descriptor is loaded. The Current Link address indicates the currently executed link address.

Table 9.13 Link Address Register Set

Register	Description
Next Link Address Register (NXLA_n/nS)	Sets and indicates the next link address. Before starting the link mode, set a link address in this register.
Current Link Address Register (CRLA_n/nS)	Indicates the currently executed link address. This register is read-only.

<Caution> In link mode, the settings can be changed by reading a descriptor. It is not possible, however, to synchronize the change of the settings with a peripheral module request or external request. Therefore, when using a peripheral module request or external request, set AM, LVL, HIEN, LOEN, and SEL of the CHCFG_n/nS register before setting Enable and do not change any of these bits in the descriptor.

(3) Descriptor Setting

In a link address, prepare a descriptor with data arranged in the order shown below.
The DMAC reads the descriptor in burst mode.

(a) Descriptor data arrangement

Table 9.14 Descriptor Data Arrangement

Address	Data	Remark
Link address + H'00	header	
Link address + H'04	Source Address	
Link address + H'08	Destination Address	
Link address + H'0C	Transaction Byte	
Link address + H'10	Config	The register mode cannot be set.
Link address + H'14	Interval	
Link address + H'18	Extension	
Link address + H'1C	Next Link Address	

Remark: As a link address, set an address aligned along the 32-bit boundary.

(b) header

The header indicates the status of the descriptor, as shown below.
The DMAC reads this area when a DMA transfer is started in link mode. Also, after a DMA transaction is completed, the DMAC writes back the transfer status to the area.

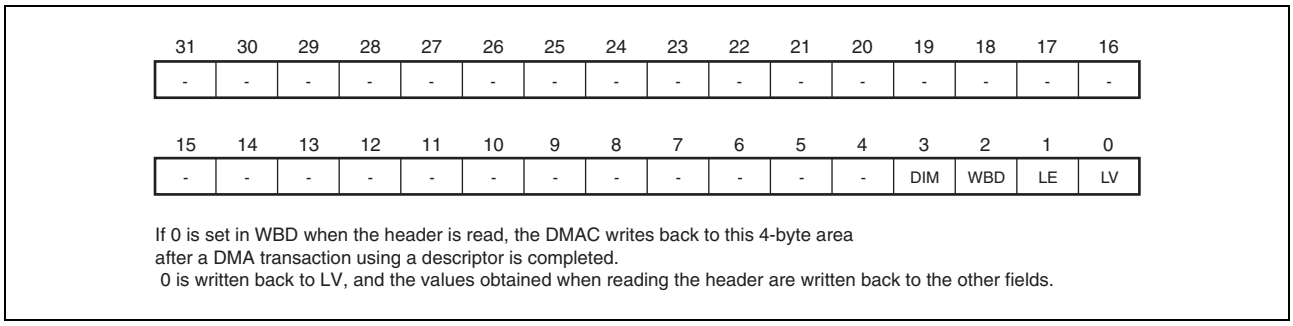


Figure 9.8 Header Area

Table 9.15 Header Area

Bit Position	Bit Name	Meaning
31 to 4	—	—
3	DIM	Descriptor Interrupt Mask Sets whether to mask the DMA transfer end interrupt if 0 is set in LV when the header is loaded. 0: Issues a DMA transfer end interrupt. 1: Does not issue a DMA transfer end interrupt.
2	WBD	Write Back Disable Sets whether to mask LV bit writeback. When 1 is set in this bit, the DMAC does not perform writeback. 0: Writes the LV bit back to 0. 1: Does not write back the LV bit.
1	LE	Link End Indicates whether the link ends with the DMA transaction of this descriptor. Set 1 in this bit to indicate the end of the link. 0: The link continues. 1: The link ends.
0	LV	Link Valid Indicates whether this descriptor is valid. If 0 is set in WBD, the DMAC writes 0 in this bit after the DMA transaction written in the descriptor is executed. When setting the header, set 1 in this bit. 0: Descriptor invalid 1: Descriptor valid

(c) Descriptor data other than the header

The data items of the descriptor other than the header are the same as defined in the internal register specifications (note that the DMS bit of the CHCFG_n/nS register cannot be changed using the descriptor). For information about the internal register specifications, see section 9.4, Register Descriptions.
For descriptor setting examples, see section 9.8, DMA Setting Examples.

(d) CACHE settings for descriptor access

The CACHE settings for descriptor access can be set in LWCA and LDCA of the DMA control register (DCTRL_0_7/0_7S, DCTRL_8_15/8_15S). Make these settings as appropriate for the access destination in which the descriptor is prepared.

(e) Descriptor area and DMA transfer area

The following figure outlines the descriptor area and DMA transfer area that are accessed by the DMAC.

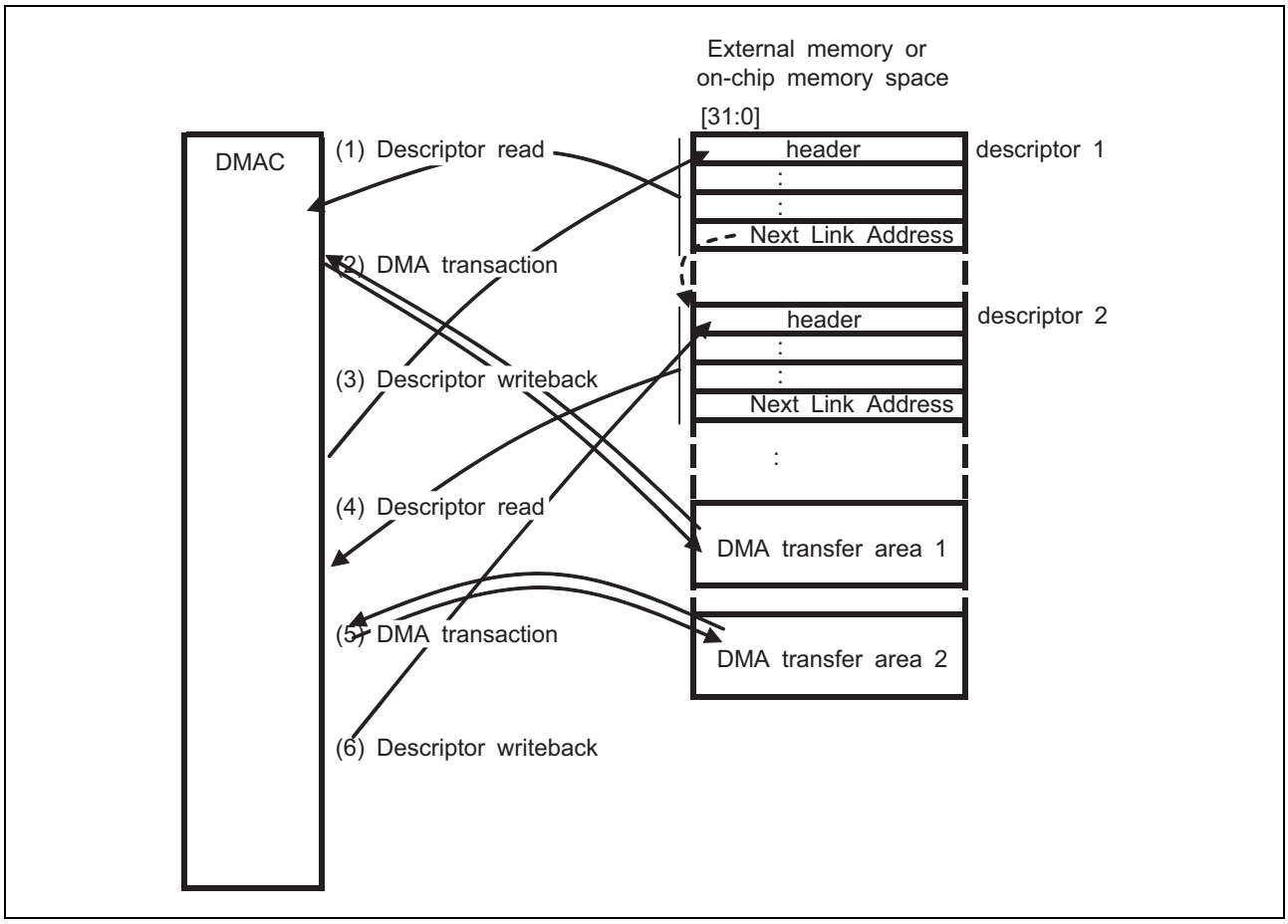


Figure 9.9 Outline of the Descriptor Area and DMA Transfer Area

1. Descriptor read
   The values set in the internal Next Link Address register are loaded to the Current Link Address register, and a descriptor is read from the external memory space (descriptor1) pointed to by the Current Link Address register.
2. DMA transfer
   When 1 is set in the LV bit of the header in the descriptor, a DMA transfer is executed according to the descriptor data.
3. Descriptor writeback
   When 0 is set in the WBD bit of the header after the DMA transfer of the set number of bytes is completed, the DMAC writes back data in word size to the header of descriptor1, with 0 set in LV and the other bits containing the values read in <1>.
4. Descriptor read
   When 0 is set in the LE bit of the header in the last read descriptor (<1>), the next descriptor is read from the address (descriptor2) indicated by Next Link Address in the descriptor.
5. DMA transfer
   When 1 is set in the LV bit of the header in the descriptor, a DMA transfer is executed according to the descriptor data.
6. Descriptor writeback
   When 0 is set in the WBD bit of the header after the DMA transfer of the set number of bytes is completed, the DMAC writes back data in word size to the header of descriptor2, with 0 set in LV and the other bits containing the values read in <4>.

4 through 6 are repeated.
When the header contains 1 in LE and 0 in WBD, the DMAC executes a DMA transfer using the settings of that descriptor, writes back data with 0 set in the LV bit of the header and ends the operation.
When the header contains 1 in both LE and WBD, the DMAC executes a DMA transfer using the settings of that descriptor and ends the operation (without writing back).
When the header contains 0 in LV, the DMAC ends the operation (without executing a DMA transfer).

(4) Descriptor Configuration Examples

In link mode, a descriptor can be configured as shown below.

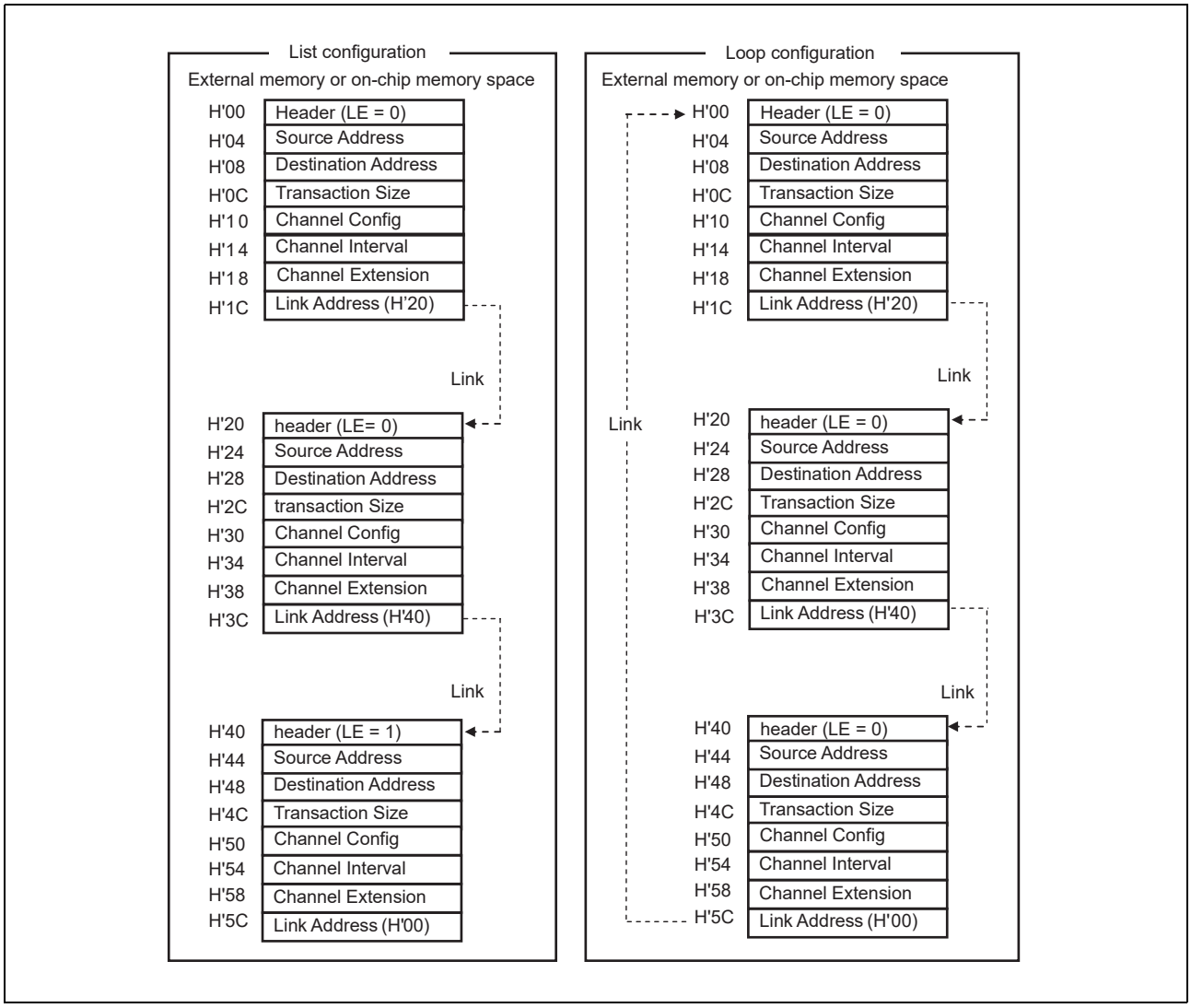


Figure 9.10 Descriptor Configuration Examples

- List configuration
  The link is ended by setting 1 in the LE bit of the header in the last descriptor.

- Loop configuration
  A descriptor can be created with a loop configuration, by setting the address of the top descriptor in the link address of the last descriptor. To end the loop, change the value of the LE bit of the header to 1 before the DMAC reads the descriptor, or follow the transfer suspension procedure.

9.7 DMA Transfer

The basic operation of DMA transfer is described here.

9.7.1 Transfer Mode

Two transfer modes are supported: single transfer mode and block transfer mode.
To select a transfer mode, set the TM bit of CHCFG_n/nS for each channel.

Table 9.16 Basic Transfer Setting

Transfer Mode	TM (CHCFG_n/nS)	Function
Single transfer	0	A single DMA transfer is executed in response to a DMAREQ.
Block transfer	1	In response to a DMAREQ, the DMAC continues to execute the transfer until the DMA transaction is completed.

(1) Single Transfer Mode

When a DMA transfer request is received, a DMA transfer is executed once in the direction indicated by REQD (source or destination). A DMA transfer is executed once each time a transfer request is received, and this operation continues until the number of bytes loaded to CRTB_n/nS is reached (arbitration between channels is accomplished for each DMA transfer).

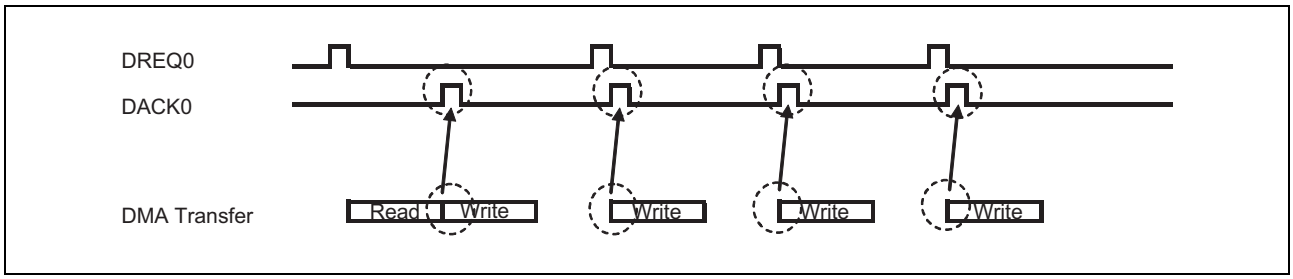


Figure 9.11 Single Transfer Mode (REQD = 1, SDS > DDS)

(2) Block Transfer Mode

Once a DMA transfer request is received, the DMAC continues to execute the transfer until data equivalent to the number of bytes loaded to the DMA transfer byte register (CRTB_n/nS register) is transferred (the DMA transaction is completed) (arbitration between channels is accomplished for each DMA transfer).

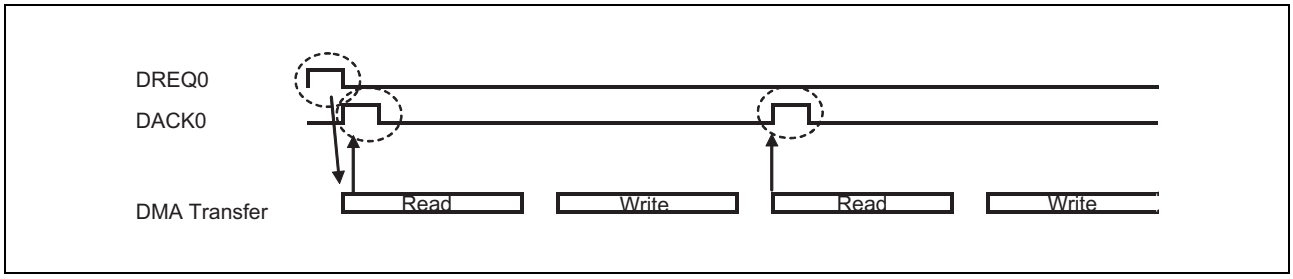


Figure 9.12 Block Transfer Mode (REQD = 0, SDS < DDS)

9.7.2 Priority Control for DMA Channels

Within channels 0 to 7 and 8 to 15, two priority control modes are supported: fixed priority mode and round robin mode. Only round robin mode is supported for priority control between the group of channels 0 to 7 and the group of channels 8 to 15. To select a priority control mode, use the PR bit of the DMA control register (DCTRL register). The fixed priority mode is selected when 0 is set in the PR bit, and the round robin mode is selected when 1 is set.
Read priority and write priority are controlled independently.
The DMAC issues transfer requests to different channels concurrently without waiting for the completion of any particular transfer and processes responses in the order it receives them. Therefore, the order in which the channels start transactions is not necessarily consistent with the order in which the transactions end.

Table 9.17 Priority Control Setting

Mode	PR (DCTRL)	Function	Purpose
Fixed priority	0	Requests are controlled based on the fixed order of priority for channels 0 to 7 and 8 to 15 (High: CH0 (CH8) > CH1 (CH9) > CH2 (CH10) > CH3 (CH11) > CH4 (CH12) > CH5 (CH13) > CH6 (CH14) > CH7 (CH15): Low).	Use this mode when the channels have a specific order of priority.
Round robin	1	Requests are controlled in a round robin fashion.	Use this mode to execute all requests evenly.

(1) Fixed Priority Mode

In fixed priority mode, the channels have a fixed order of priority in channels 0 to 7 and 8 to 15. Round robin mode is used to determine the priority between the group of channels 0 to 7 and the group of channels 8 to 15.

Immediately after a reset, the order of priority is as follows.
High CH0 > CH8 > CH1 > CH9 > CH2 > CH10 > CH3 > CH11 > CH4 > CH12 > CH5 > CH13 > CH6 > CH14 > CH7 > CH15 Low

If there is a transfer request from DMA channel 0 in this state, a transfer is executed on DMA channel 0. After the transfer is completed, the order of priority is as follows.
High CH8 > CH0 > CH9 > CH1 > CH10 > CH2 > CH11 > CH3 > CH12 > CH4 > CH13 > CH5 > CH14 > CH6 > CH15 > CH7 Low

If a DMA transfer request occurs on multiple channels simultaneously, the DMA transfer request of the channel having the smallest channel number is given priority. The following figure shows an example where a DMA transfer request occurs on a channel having a higher priority while a DMA transfer is being executed in fixed priority mode.

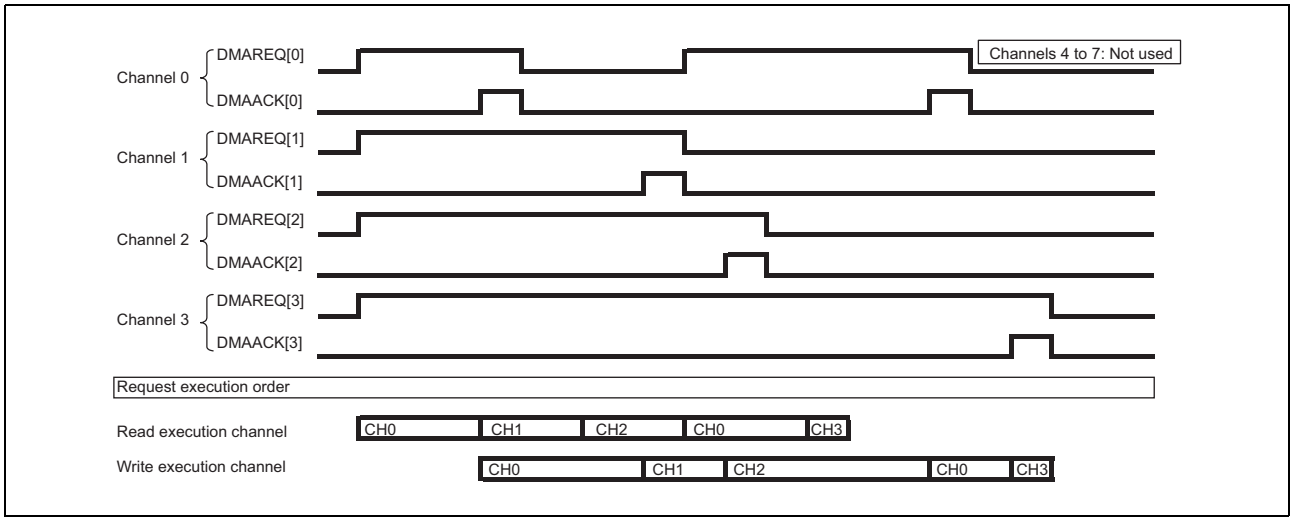


Figure 9.13 Fixed Priority Mode (Number of Channels = 4, REQD = 1)

(2) Round Robin Mode

In round robin mode, each time a transfer request is received from a channel in the group of channels 0 to 7 and the group of channels 8 to 15, the order of priority is changed in such a way that the channel that executed a transfer last has the lowest priority.
Round robin mode is used to determine the priority between the group of channels 0 to 7 and the group of channels 8 to 15.

Immediately after a reset, the order of priority is the same as that of the fixed priority mode, which is as follows.
High CH0 > CH8 > CH1 > CH9 > CH2 > CH10 > CH3 > CH11 > CH4 > CH12 > CH5 > CH13 > CH6 > CH14 > CH7 > CH15 Low

If a transfer request is received from DMA channel 2 in this state, a transfer is executed on DMA channel 2. After the transfer is completed, the order of priority is as follows.
High CH8 > CH3 > CH9 > CH4 > CH10 > CH5 > CH11 > CH6 > CH12 > CH7 > CH13 > CH0 > CH14 > CH1 > CH15 > CH2 Low

The following figure shows an example where DMA transfers are executed in round robin mode.

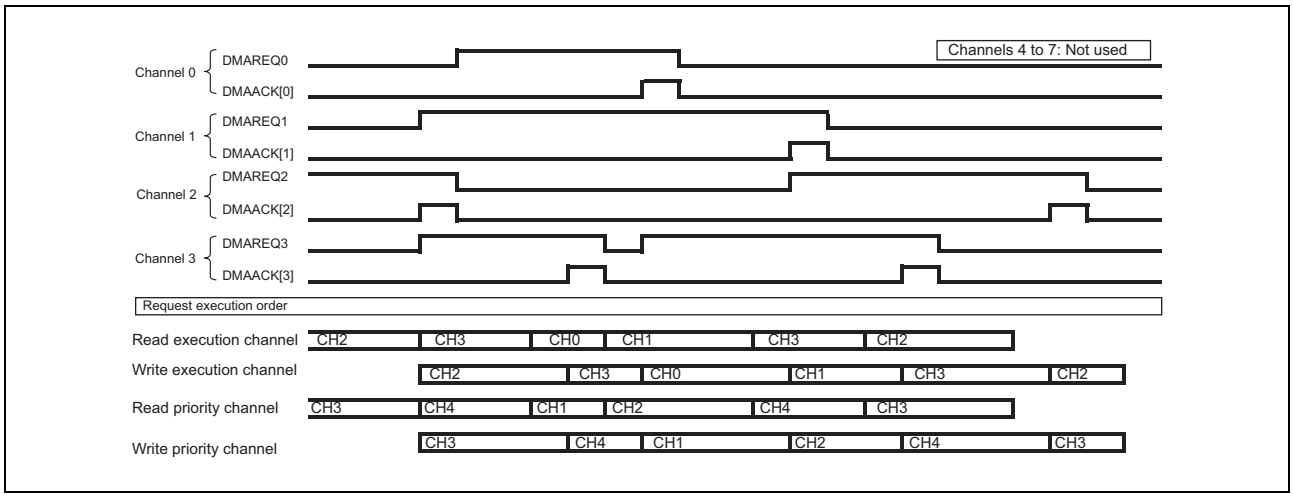


Figure 9.14 Round Robin Mode (Number of Channels = 4, REQD = 0)

The channel whose channel number is the number of the currently transferring channel + 1 gets to execute a DMA transfer next. If there is no transfer request from this channel, the channel whose channel number is the number of this channel + 1 gets to execute a DMA transfer.

9.7.3 Number of States of an External Bus Cycle

When this module is the bus master, the number of states of an external bus cycle is controlled by the bus state controller as when the CPU is the bus master. For details, refer to section 8, Bus State Controller.

9.7.4 DMA Transfer Request

Edge detection or level detection can be selected using the LVL bit of the CHCFG_n/nS register.
The HIEN and LOEN bits of the CHCFG_n/nS register are used to select either the rising edge or falling edge in the case of edge detection or either the high level or low level in the case of level detection.
When the transfer request is by an on-chip peripheral module, set the CHCFG_n/nS register according to Table 9.4.
When the transfer request is by the external pin (DREQ0), set the detection conditions (rising/falling edge and high/low level) according to Table 9.18.

Table 9.18 Setting for Detection of External Pin Request

Mode	LVL (CHCFG_0/0S)	HIEN (CHCFG_0/0S)	LOEN (CHCFG_0/0S)	Function
Edge detection	0	0	0	Specify this value when auto request triggers are in use.
			1	Detects external pin request (DREQ0) at its falling edge.
		1	0	Detects external pin request (DREQ0) at its rising edge.
			1	Setting prohibited
Level detection	1	0	0	Setting prohibited
			1	Detects external pin request (DREQ0) in Low level mode.
		1	0	Detects external pin request (DREQ0) in High level mode.
			1	Setting prohibited

(1) Edge Detection

Setting 0 in the LVL bit of the CHCFG_n/nS register enables edge detection.
When 1 is set in the HIEN bit of the CHCFG_n/nS register, rising edge detection is enabled. When 1 is set in the LOEN bit, falling edge detection is enabled.
Wait for DACK0 to be detected, before issuing the next DREQ0 request.

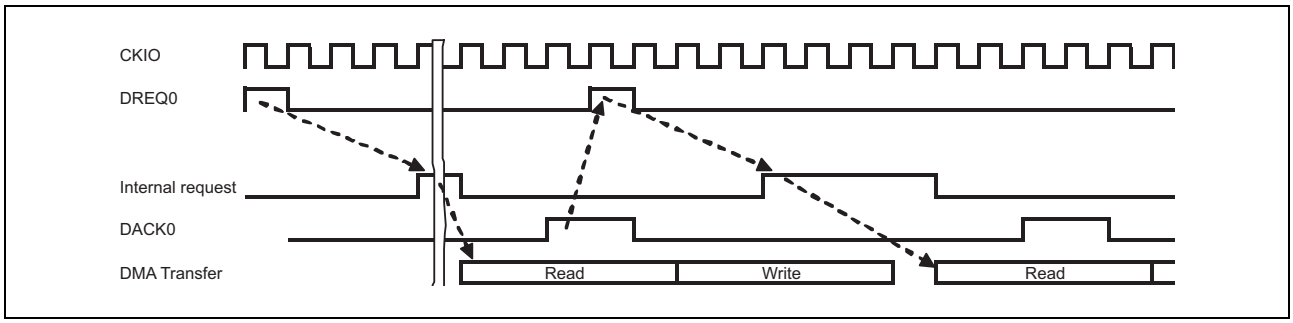


Figure 9.15 Edge Detection Timing (HIEN = 1, REQD = 0)

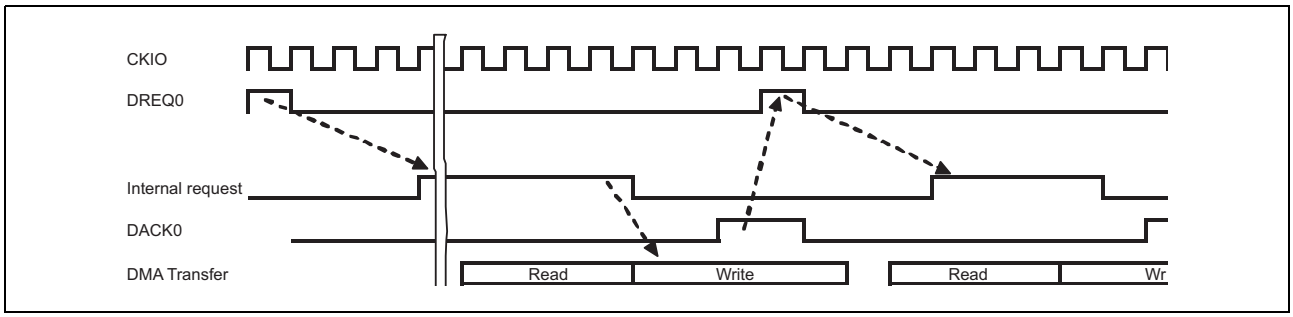


Figure 9.16 Edge Detection Timing (HIEN = 1, REQD = 1)

(2) Level Detection

Setting 1 in the LVL bit of the CHCFG_n/nS register enables level detection.
DREQ0 is regarded as valid when it remains active for two consecutive clock cycles or more (depending on the HIEN and LOEN settings).
When the level mode is selected for DACK0, it remains at the High level until DREQ0 is deasserted.
When the next DMA transfer request is to be issued, DACK0 needs to be deasserted before that DREQ0 can be asserted.

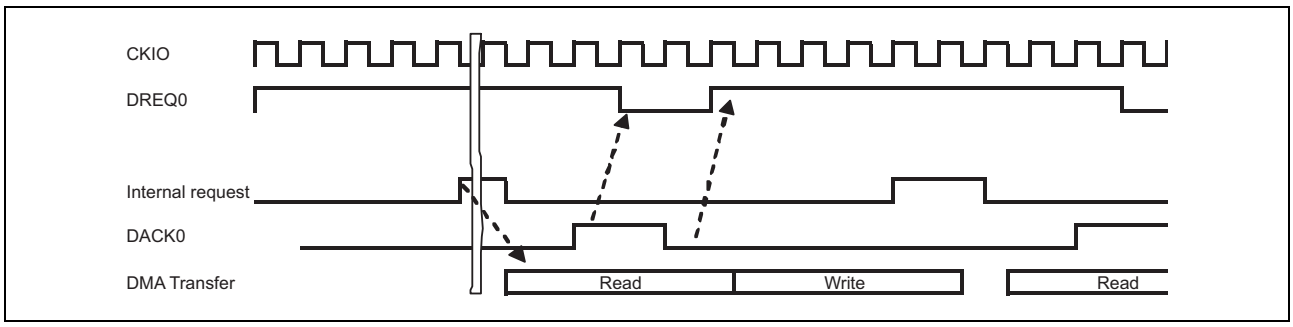


Figure 9.17 Level Detection Timing (HIEN = 1, REQD = 0, AM[2:0] = 001)

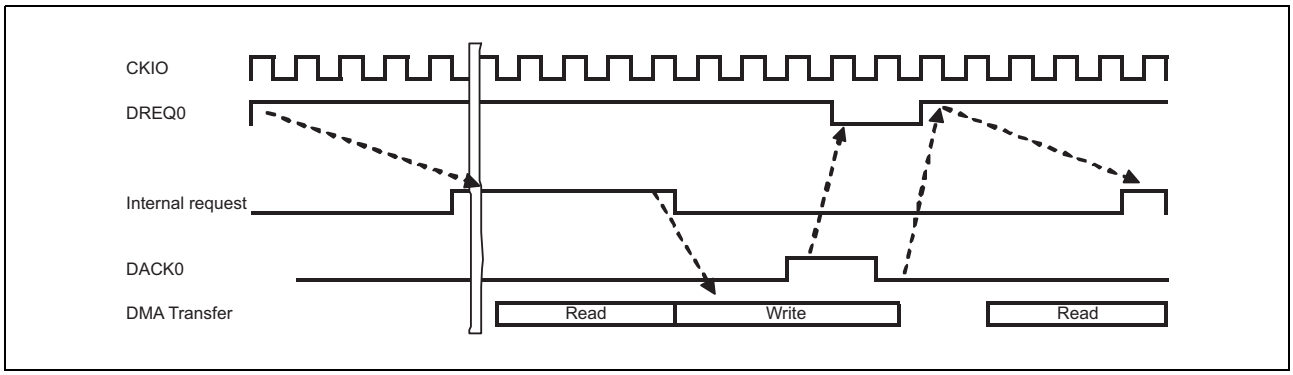


Figure 9.18 Level Detection Timing (HIEN = 1, REQD = 1, AM[2:0] = 001)

9.7.5 DMA Acknowledge Output Function

DACK0 is an acknowledge signal that is sent to DREQ0. Level output and bus cycle output settings are supported as the DACK0 output mode. DACK0 is asserted at the same time as CS# assertion except for the MPX-IO interface. For details, refer to section 8, Bus State Controller.

(1) DMA Acknowledge Signal Output Timing Setting

Upon receiving a DMA transfer request, the DACK0 pin becomes active (High level output). By using the REQD and AM[2:0] bits of the CHCFG_n/nS register, the DACK0 output timing can be set as shown below.

Table 9.19 DACK0 Output Timing Setting

Mode	AM[2] (CHCFG_0/0S)	AM[1:0] (CHCFG_0/0S)	REQD (CHCFG_0/0S)	Purpose
Pulse	0	00	0	Setting prohibited
			1	
Level	0	01	0 (Active during read)	DACK0 is output as a level. DACK0 remains asserted until DREQ0 is deasserted.
			1 (Active during write)	
Bus cycle	0	10 11	0 (Active during read)	DACK0 is output for the duration of a bus cycle. Use this mode to keep DACK0 asserted until the end of the bus cycle.
			1 (Active during write)	
Mask	1	—	—	Make this setting when using auto request trigger or when performing the SCIFA transfer.

(2) Level Output

Setting 001 in the AM bits of the CHCFG_n/nS register enables level output. DACK0 remains asserted until DREQ0 is deasserted.

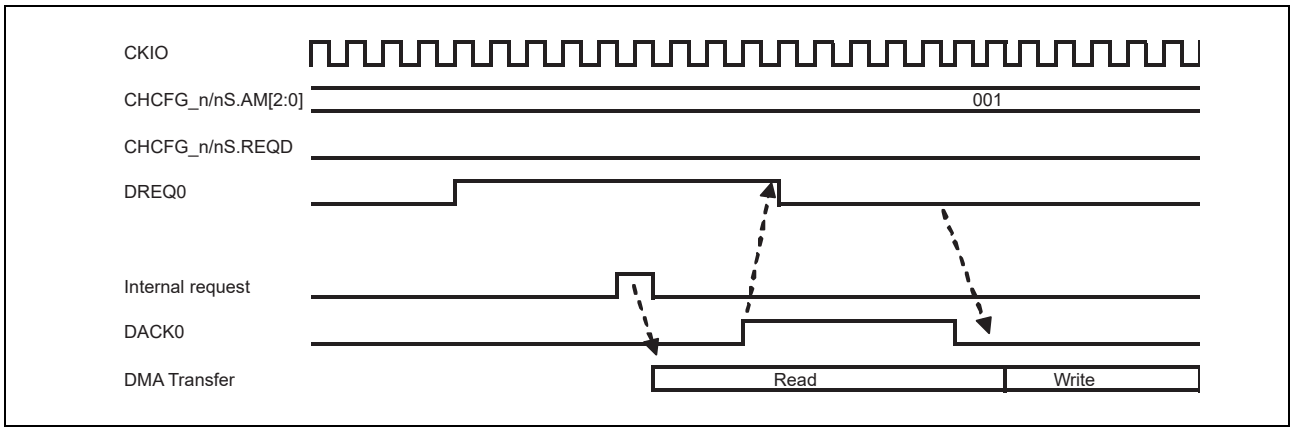


Figure 9.19 DACK0 Output Timing (AM[2:0] = 001, REQD = 0)

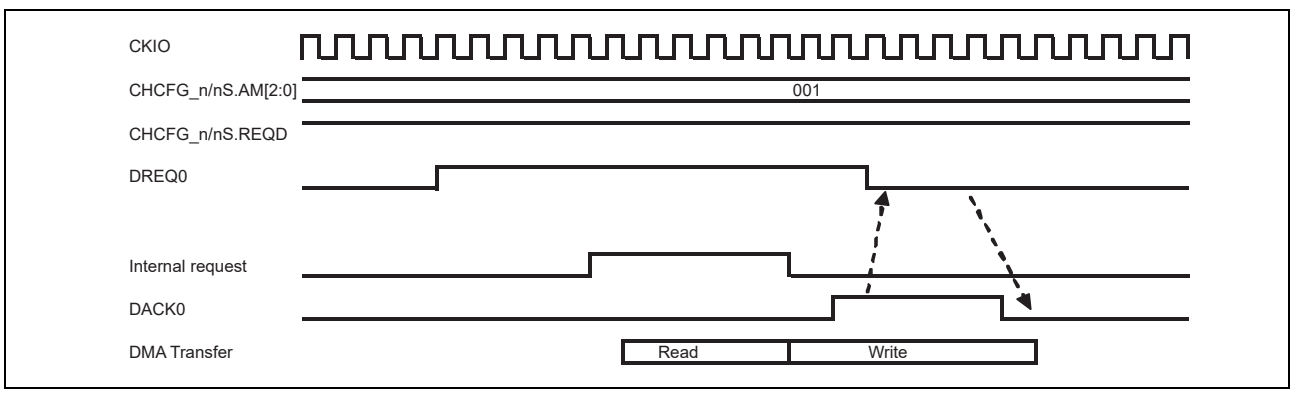


Figure 9.20 DACK0 Output Timing (AM[2:0] = 001, REQD = 1)

(3) Bus Cycle Output

Setting 010 in the AM bits of the CHCTRL_n/nS register enables bus cycle output. DACK0 remains active for the duration of a bus cycle.

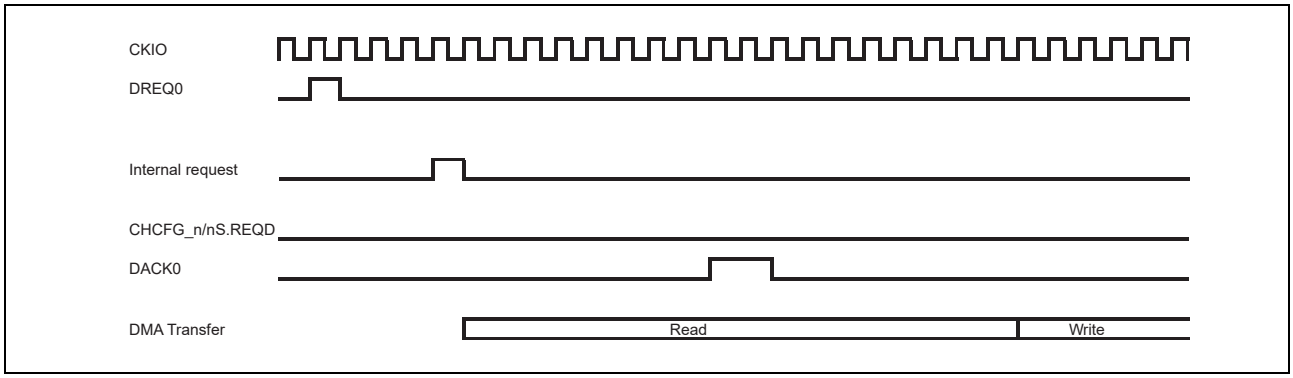


Figure 9.21 Bus Cycle Output Timing (REQD = 0)

- In the read active mode (REQD = 0), DACK0 remains active from the time when a read request is output on the bus until one cycle after the final read data.
- When level detection is selected for DREQ0, DREQ0 remains disabled until the cycle following the end of the bus cycle.

The following signals trigger the rise and fall of DACK0:
Rise: Transfer start (MARVALID = 1)
Fall: Transfer end (MRLAST & MRREADY = 1)

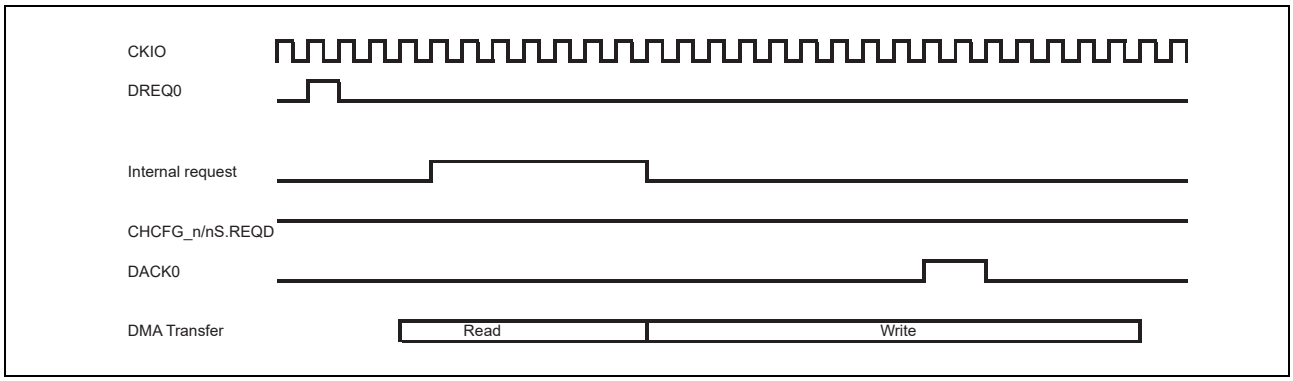


Figure 9.22 Bus Cycle Output Timing (REQD = 1)

- In the write active mode (REQD = 1), DACK0 remains active from the time when a write request is output until one cycle after the response to the final data is returned.
- When level detection is selected for DREQ0, DREQ0 remains disabled until the cycle following the end of the bus cycle.

The following signals trigger the rise and fall of DACK0:
Rise: Transfer start (MAWVALID = 1)
Fall: Transfer end (MBVALID & MBREADY = 1)

9.7.6 DMA Transfer End Output Function

TEND0 is a transaction completion signal that is sent to the source of a DMA transfer request. TEND0 is asserted as the same time as DACK0 for the last transfer transaction. Figure 9.23 shows the TEND0 output timing.

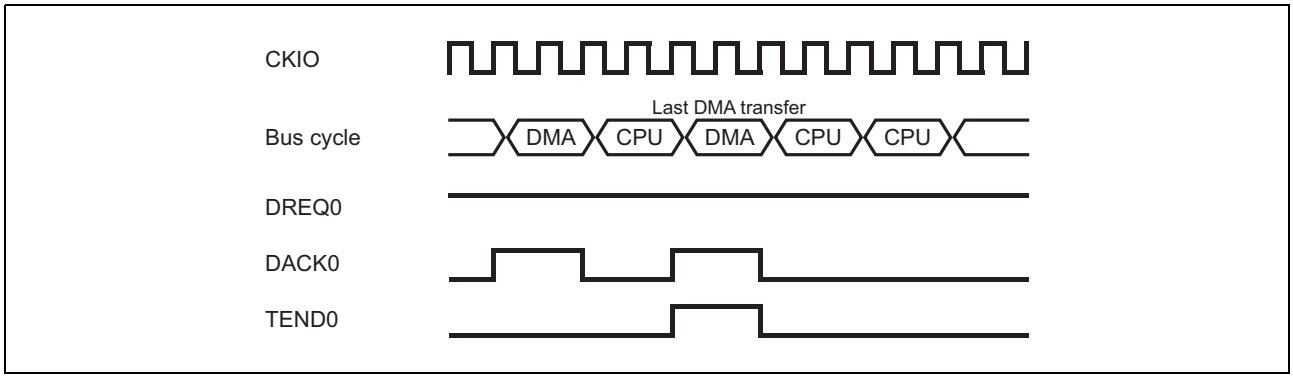


Figure 9.23 TEND0 Output Timing

9.7.7 DMA Transfer End Interrupt

The DMA transfer end interrupt is an interrupt request signal that indicates that a DMA transaction is completed.
There is an independent DMA transfer end interrupt for each channel.
When the transfer of data equivalent to the total transfer byte count loaded to the CRTB (Current Transaction Byte) is completed, 1 is set in END of the CHSTAT_n/nS register. In this case, when 0 is set in DEM of the CHCFG_n/nS register, the DMA transfer end interrupt is output (n = 0 to 15). (When writeback is performed in link mode, the signal is output after the writeback operation.)
When 0 is set in LV of the header in the read descriptor in link mode, 1 is set in DER of the CHSTAT_n/nS register. In this case, when 0 is set in DIM of the header, the DMA transfer end interrupt is output.

Table 9.20 Assertion Conditions of DMA Transfer End Interrupt

Source	Condition	DMA Transfer End Interrupt Mask Signal
DMA transaction end	When the transfer of data equivalent to the total transfer byte count loaded to the CRTB (Current Transaction Byte) is completed with an OKAY response (or after the writeback operation when writeback is performed in link mode)	DEM bit of the CHCFG_n/nS register
Descriptor invalid	When 0 is set in LV of the header in the read descriptor in link mode while 0 is set in DIM of the header	DIM bit of the header

9.7.8 DMA Error Interrupt

If an error response is received for a DMA transfer or descriptor access, the DMAC regards it as an error and stops the transfer. Upon receiving an error response, the EN bit of the CHSTAT_n/nS register of transferring channel n is cleared to 0 and 1 is set in the ER bit (n = 0 to 15). Also, the DMA error interrupt is output.
The DMA error interrupt cannot be masked.
Once an error occurs, the data of the whole transfer cannot be guaranteed. Be sure to start the transaction again from the beginning by following the procedure below.

1. Set 1 in the SWRST bit of the CHCTRL_n/nS register.
2. Set each register again.

9.7.9 Interval Count Function

The interval at which a DMA transfer is executed can be adjusted by setting the ITVL bit of the channel interval register (CHITVL_n/nS). This function is intended to prevent the DMA controller from occupying the bus all the time.
When a read or write operation is completed, a countdown starts from the value set in CHITVL_n/nS. The next internal request is not executed until the count value reaches 0.
The following figure shows an example of how this works.

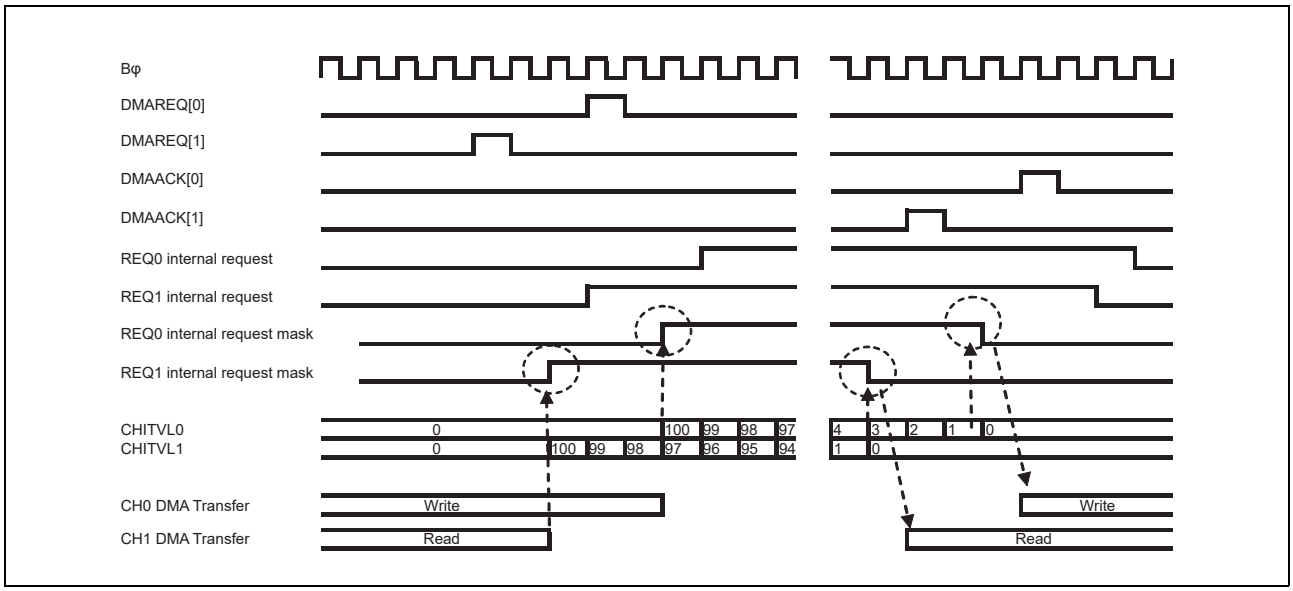


Figure 9.24 Interval Count

9.7.10 Difference in Operation Due to the Transfer Size

(1) When the Source Transfer Size Is Smaller

When the read of data equivalent to the destination data size is completed, the data is written to the destination.
The following figure shows a timing chart where the source transfer size is 8 bits and the destination transfer size is 32 bits (in the case of rising edge detection).

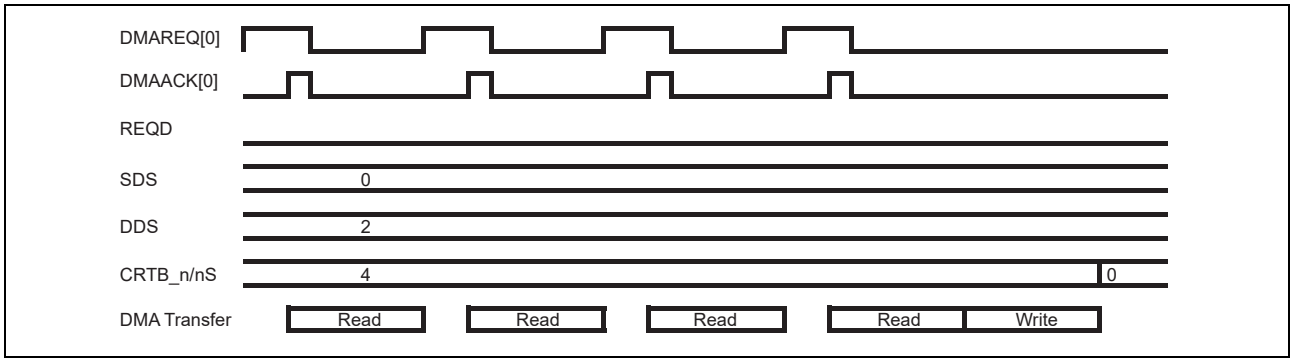


Figure 9.25 When the Source Transfer Size Is Smaller (LVL = 0, HIEN = 1, REQD = 0, SDS < DDS in CHCFG_n/nS)

(2) When the Destination Transfer Size Is Smaller

Since the source transfer size is larger, multiple destination writes occur after a single source read. The following figure shows a timing chart where the source transfer size is 64 bits and the destination transfer size is 16 bits (in the case of rising edge detection) (1 is set in REQD of the CHCFG_n/nS register).

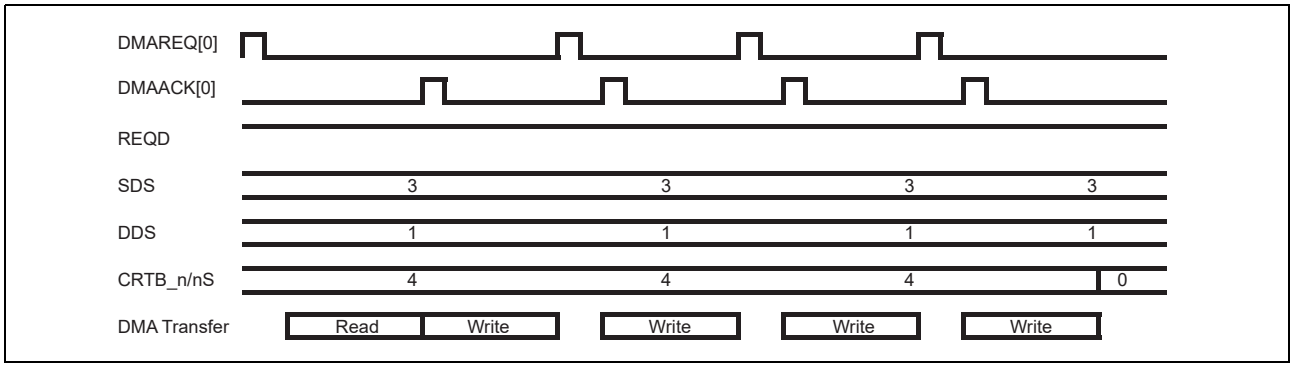


Figure 9.26 When the Destination Transfer Size Is Smaller (LVL = 0, HIEN = 1, REQD = 1, SDS > DDS in CHCFG_n/nS)

(3) When the Source Transfer Size Is the Same as the Destination Transfer Size

Every time a DMA transfer request is detected, a source read and a destination write occur.
The following figure shows a timing chart where the source transfer size and the destination transfer size are both 8 bits (in the case of rising edge detection, with 1 set in REQD of the CHCFG_n/nS register).

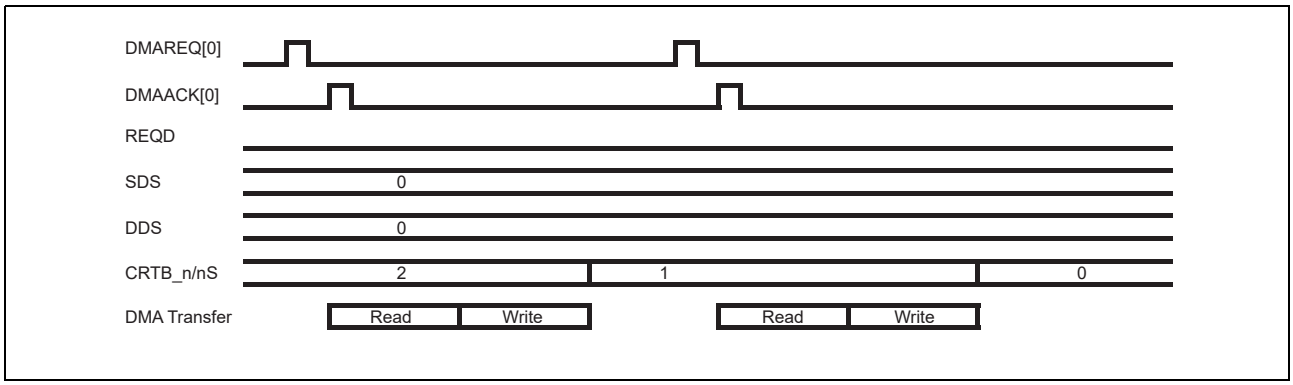


Figure 9.27 When the Source Transfer Size Is the Same as the Destination Transfer Size (LVL = 0, HIEN = 1, REQD = 0, SDS = DDS in CHCFG_n/nS)

9.7.11 Transfer Status

The channel status register indicates the status of DMA transfer execution on a channel.

(1) Suspend

A DMA transfer can be suspended by using the SETSUS bit of CHCTRL_n/nS. In this case, if an ongoing bus cycle exists, the DMAC waits for that cycle to end before suspending the transfer. Writing 1 in the CLRSUS bit restores the DMA transfer from the suspend status.

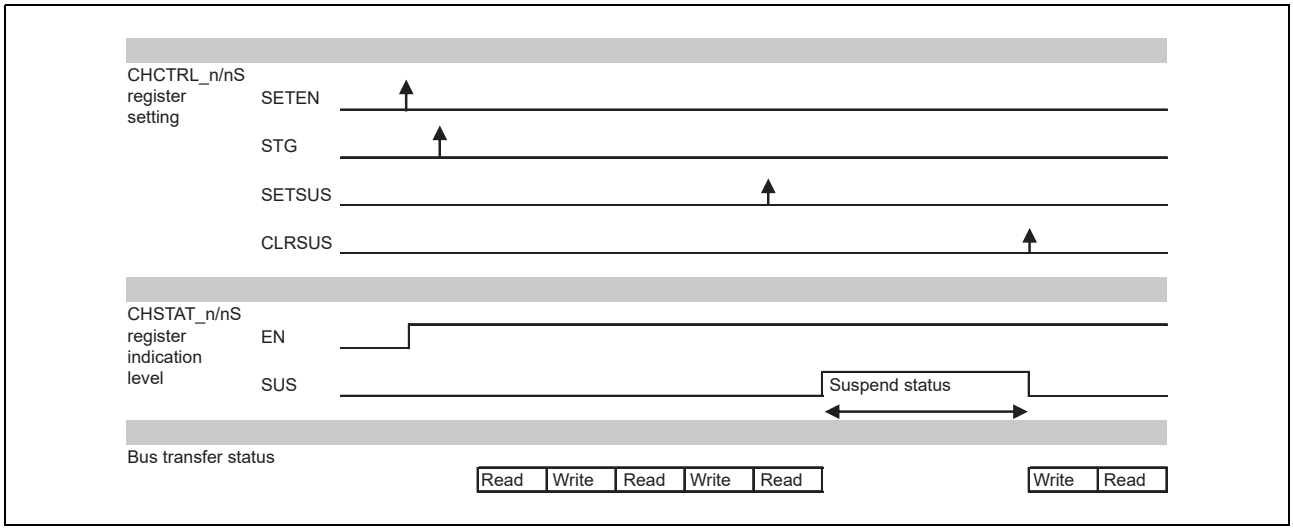


Figure 9.28 DMAC Suspend Status (Auto Request/Block Transfer)

In the above case, the DMA transfer is suspended after the read transfer is completed.
If there is any ongoing DMA transfer, the suspend status starts when that transfer is completed. To make sure that the transfer is suspended, read the CHSTAT or DSTAT_SUS register, after setting SETSUS, and check that 1 is set in the SUS bit for the relevant channel.

(2) Transfer Stop

If 1 is written to CLREN while a DMA transaction is in progress, the DMA transaction for the corresponding channel can be stopped. For the post-stop processing, two modes are supported: one sweeps out the data remaining in the buffer when the transaction is stopped (SBE = 1) and the other does not (SBE = 0). One of these modes can be selected using the SBE bit of the CHCFG_n/nS register. By default, SBE is set to 0.
When this sweep mode is enabled and CLREN is set to 1, and if a DMA transaction is stopped with data remaining in the DMAC buffer, the transaction is completed after the DMAC sweeps the data.

(a) Transfer Stop (Buffer Sweep Disabled - SBE = 0)

If 1 is set in CLREN during a DMA transfer, the DMA transfer is stopped. The stop timing depends on the value set in REQD. After stopping a DMA transfer, be sure to set 1 in SWRST to clear the DMA internal status before setting the next transfer.

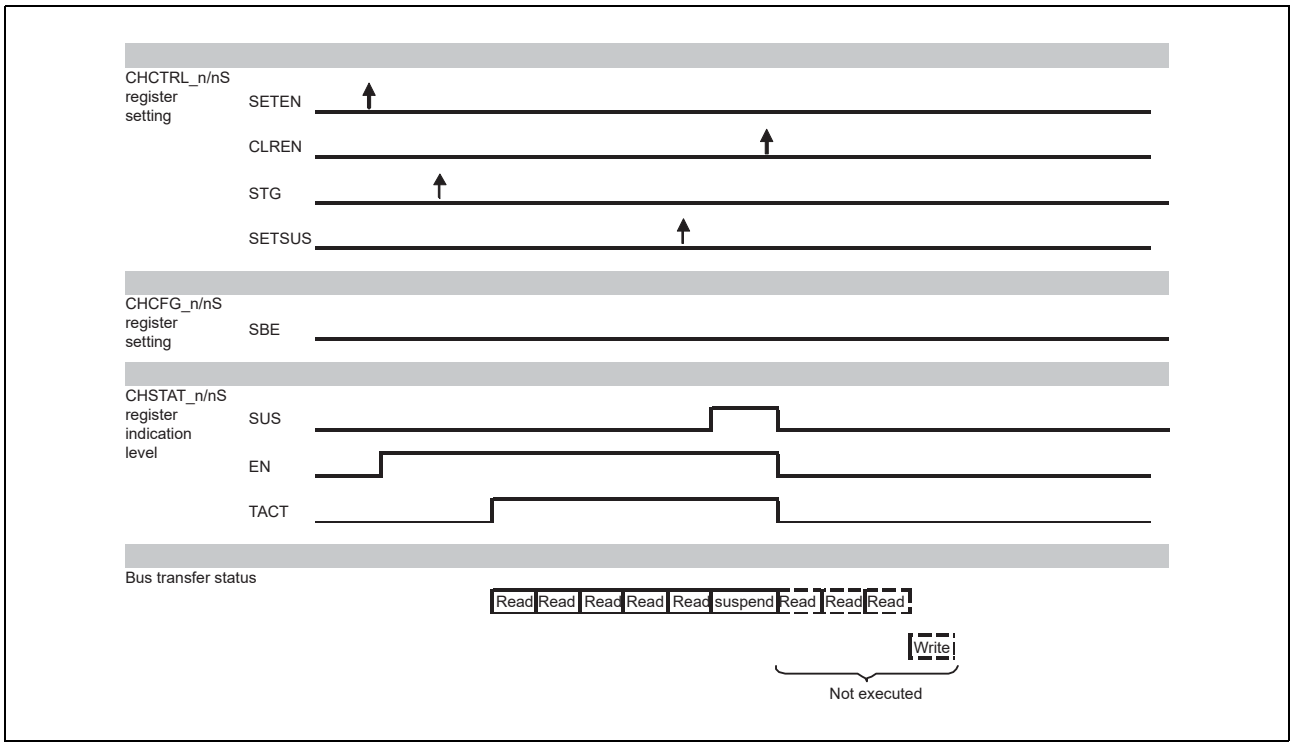


Figure 9.29 DMAC Transfer Stop

- The setting of the TACT bit being 0 indicates that the channel has been brought to a complete stop.
- If an ongoing DMA transfer is stopped before it is completed, the DMA transfer end interrupt is not asserted.
- If 0 is set in REQD, the DMA transfer is stopped when the next read is completed. (If the buffer contains any data that can be written, the DMA transfer is stopped after the data is written.)
- If 1 is set in REQD, the DMA transfer is stopped when the next write is completed.

(b) Transfer Stop (Buffer Sweep Enabled - SBE = 1)

If 1 is set in CLREN during a DMA transfer, the DMA transfer is stopped. When 0 is set in REQD, the DMA transfer is stopped after the DMAC sweeps (writes) the already read data. If 1 is set in REQD to use hardware requests, do not use the sweep mode. After stopping a DMA transfer, be sure to set 1 in SWRST to clear the DMAC internal status before setting the next transfer.

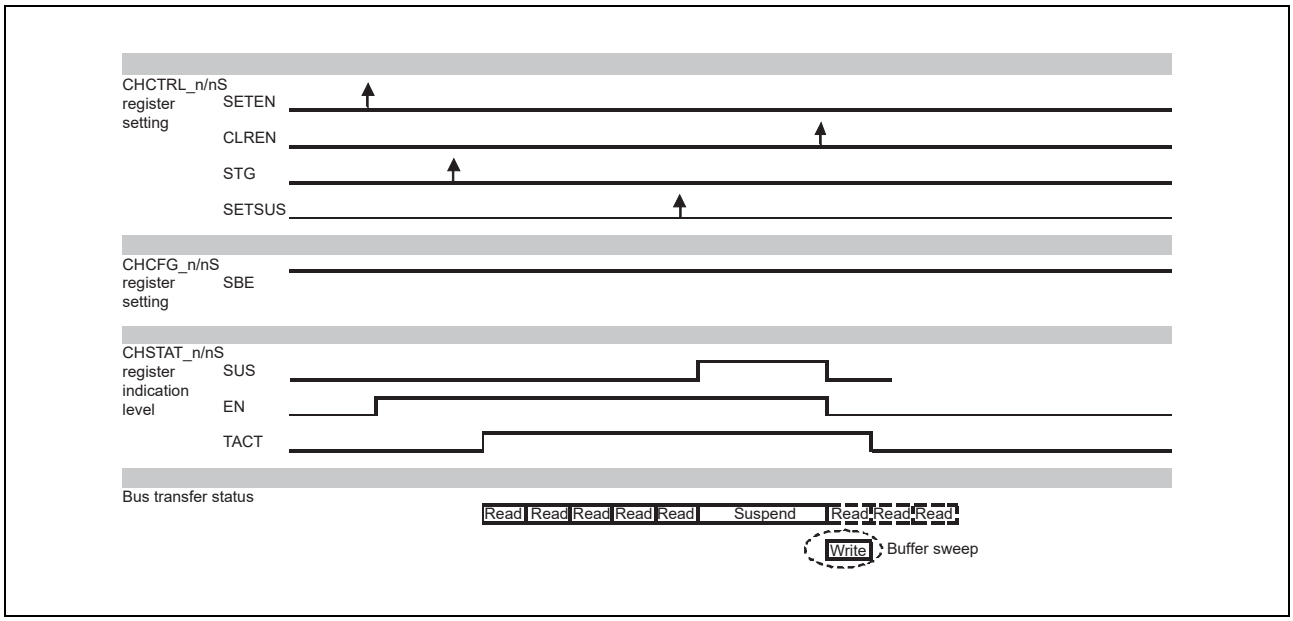


Figure 9.30 DMA Transfer Stop (Buffer Sweep Mode)

- The setting of the TACT bit being 0 indicates that the channel has been brought to a complete stop.
- If a transfer is stopped in sweep mode (SBE = 1) during the fifth read transfer by setting SETSUS and then CLREN, the read data is written before the DMA transfer is stopped.

(c) Channel Stop Check Method

Even when the EN bit is cleared to 0, the DMA transfer cannot be stopped immediately, if the bus is already executing the transfer. Therefore, in order to make sure that the DMAC has been brought to a complete stop, check that the EN bit and TACT bit are both set to 0.

(d) Transfer Stop Procedure

The transfer stop procedure is described below.

1. Set 1 in SETSUS of CHCTRL_n/nS.
2. Repeat polling until the SUS bit of CHSTAT_n/nS is set to 1. (If EN is already set to 0, the DMAC has already been stopped. Go to step 6.)
3. Set 1 in CLREN of CHCTRL_n/nS.
4. When 0 is set in SBE, the transfer is stopped according to the value of REQD. When 1 is set in SBE, the sweep mode is enabled. When 1 is set in SBE, set 0 in REQD.
5. Read CHSTAT_n/nS to check that 0 is set in the TACT bit. When TACT is set to 0, it means that the DMAC has been brought to a complete stop. When TACT is set to 1, repeat polling until this bit is set to 0.
6. To execute the next DMA transfer after stopping a transfer, be sure to set 1 in the SWRST (software reset) bit of CHCTRL_n/nS before the next DMA transfer starts.

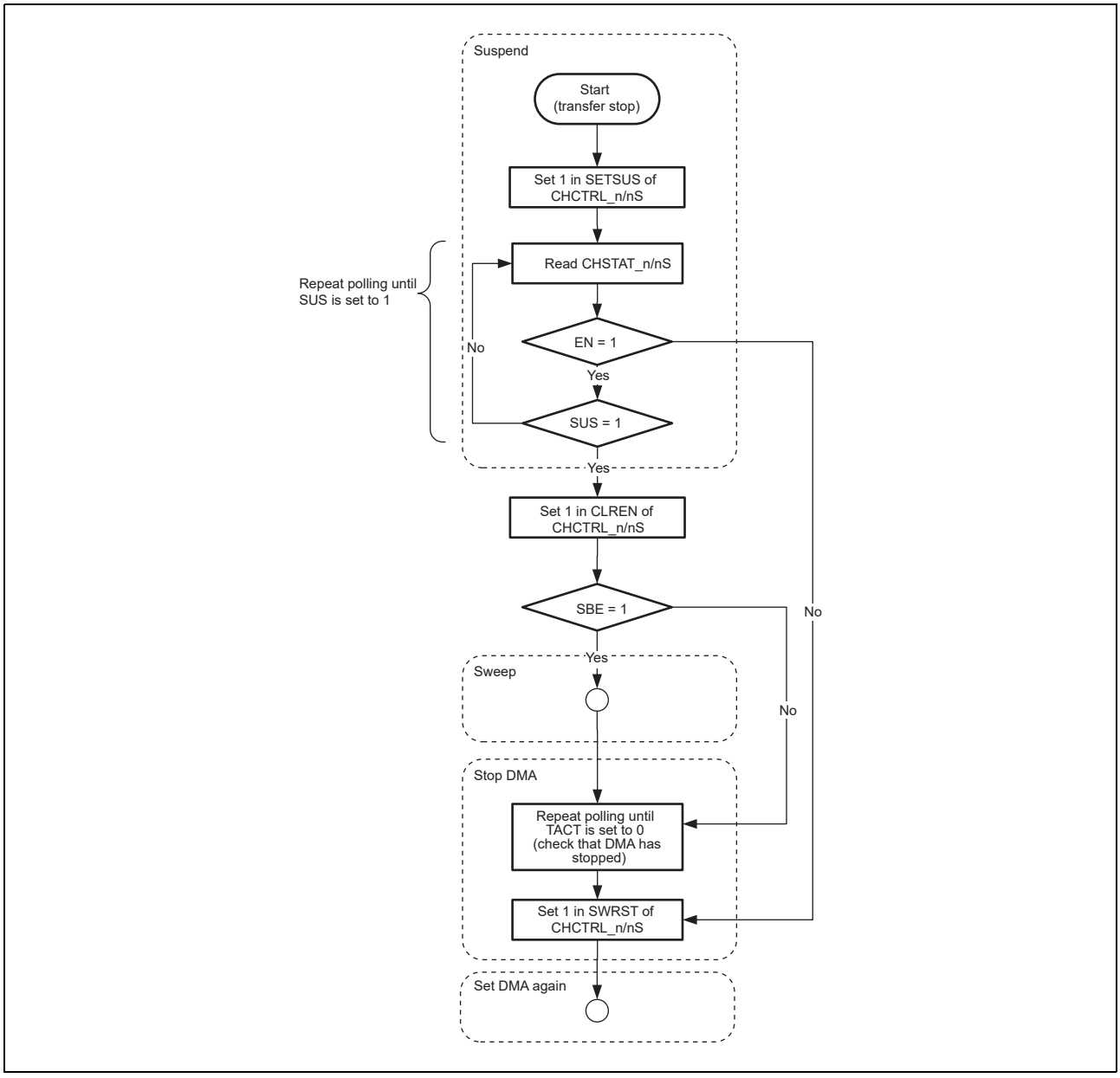


Figure 9.31 Transfer Stop Flow

9.8 DMA Setting Examples

Setting examples applicable when DMA transfer is executed using the direct memory access controller are shown in the following.
The transfer conditions for these setting examples are as follows.

Table 9.21 Transfer Condition List for DMA Transfer Setting Examples

	DMA Mode	Transfer Mode	Transfer Request
Setting example 1	Register	Single	Hardware
Setting example 2	Register	Block	Software
Setting example 3	Register (continuous execution)	Block	Software
Setting example 4	Link	Block	Software

For details of the settings, see the individual setting examples.

9.8.1 Setting Example 1 (Register Mode/Hardware Request)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 9.22 DMA Transfer Setting Example 1

Item		Description	
Channel used		3	
DMA mode		Register	
Transfer mode		Single transfer	
Register set used		Next0	
Source/destination		Source	Destination
	Start address	H'11110000	H'22220000
	Address direction	Increment	Increment
	Data size	32 bits	32 bits
DMA transfer byte count		64 bytes	
DMA transfer request		Rising edge detection by hardware	
DMAACK signal		Level output during read	
DMA transfer end interrupt mask		Not masked	
CACHE setting		Default value	

Setting example 1

N0SA = H'11110000 (source address)
N0DA = H'22220000 (destination address)
N0TB = H'00000040 (transfer byte count)

CHCFG = H'00022123 (configuration)
CHITVL = H'00000000 (interval)
CHEXT = H'00000000 (CACHE setting)

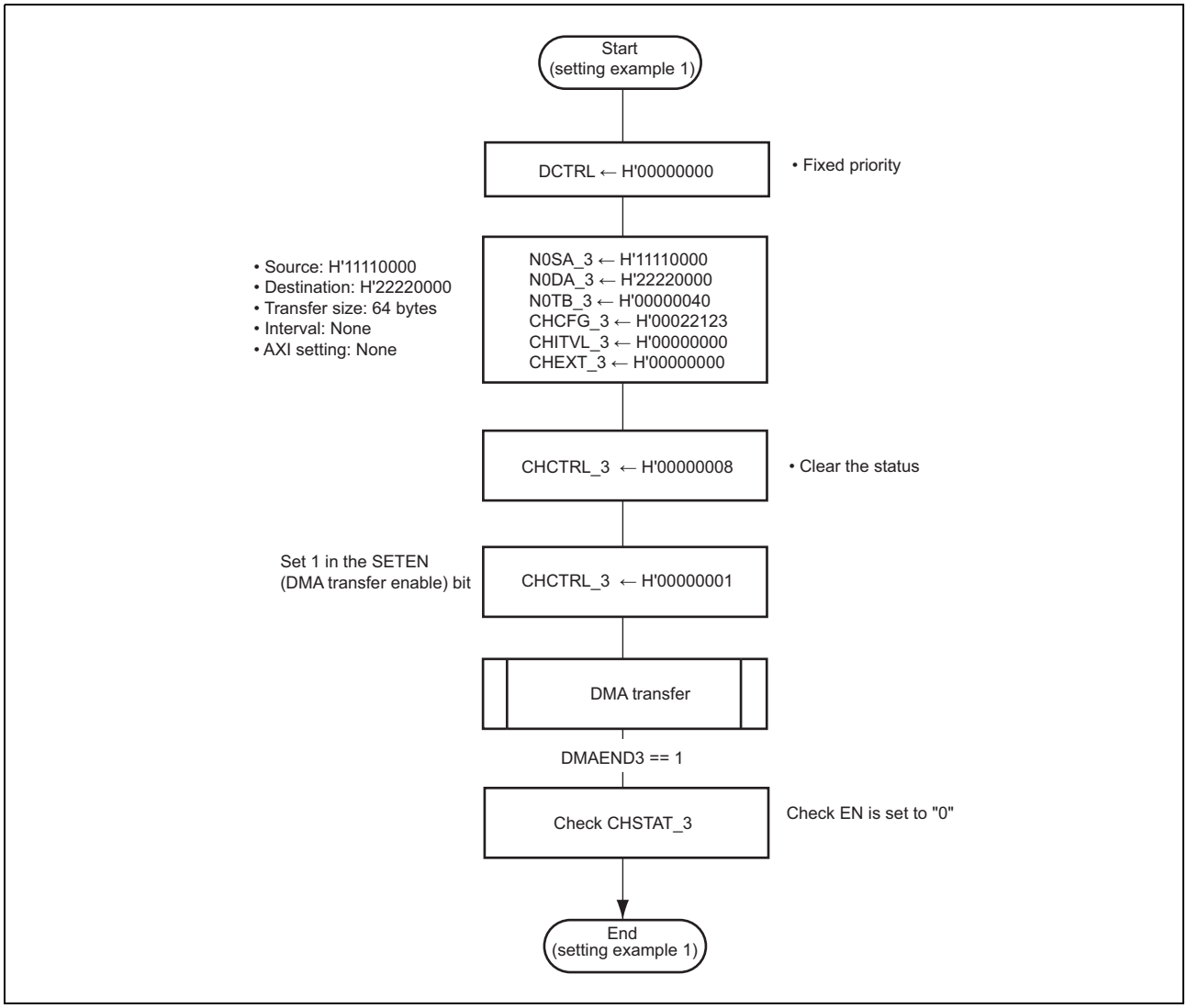


Figure 9.32 Setting Example 1

9.8.2 Setting Example 2 (Register Mode/Software Request)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 9.23 DMA Transfer Setting Example 2

Item		Description	
Channel used		2	
Priority control		Round robin	
DMA mode		Register	
Transfer mode		Block transfer	
Register set used		Next1	
Source/destination		Source	Destination
	Start address	H'0FFFE000	H'33330000
	Address direction	Increment	Increment
	Data size	8 bits	256 bits
DMA transfer byte count		128 bytes	
DMA transfer request		Auto request	
DMAACK signal		Masked	
DMA transfer end interrupt mask		Not masked	
CACHE setting		Default value	

Setting example 2

DCTRL = H'00000001 (DMA setting)

N1SA = H'0FFFE000 (source address)
N1DA = H'33330000 (destination address)
N1TB = H'00000080 (transfer byte count)

CHCFG = H'10450402 (configuration)
CHITVL = H'00000000 (interval)
CHEXT = H'00000000 (CACHE setting)

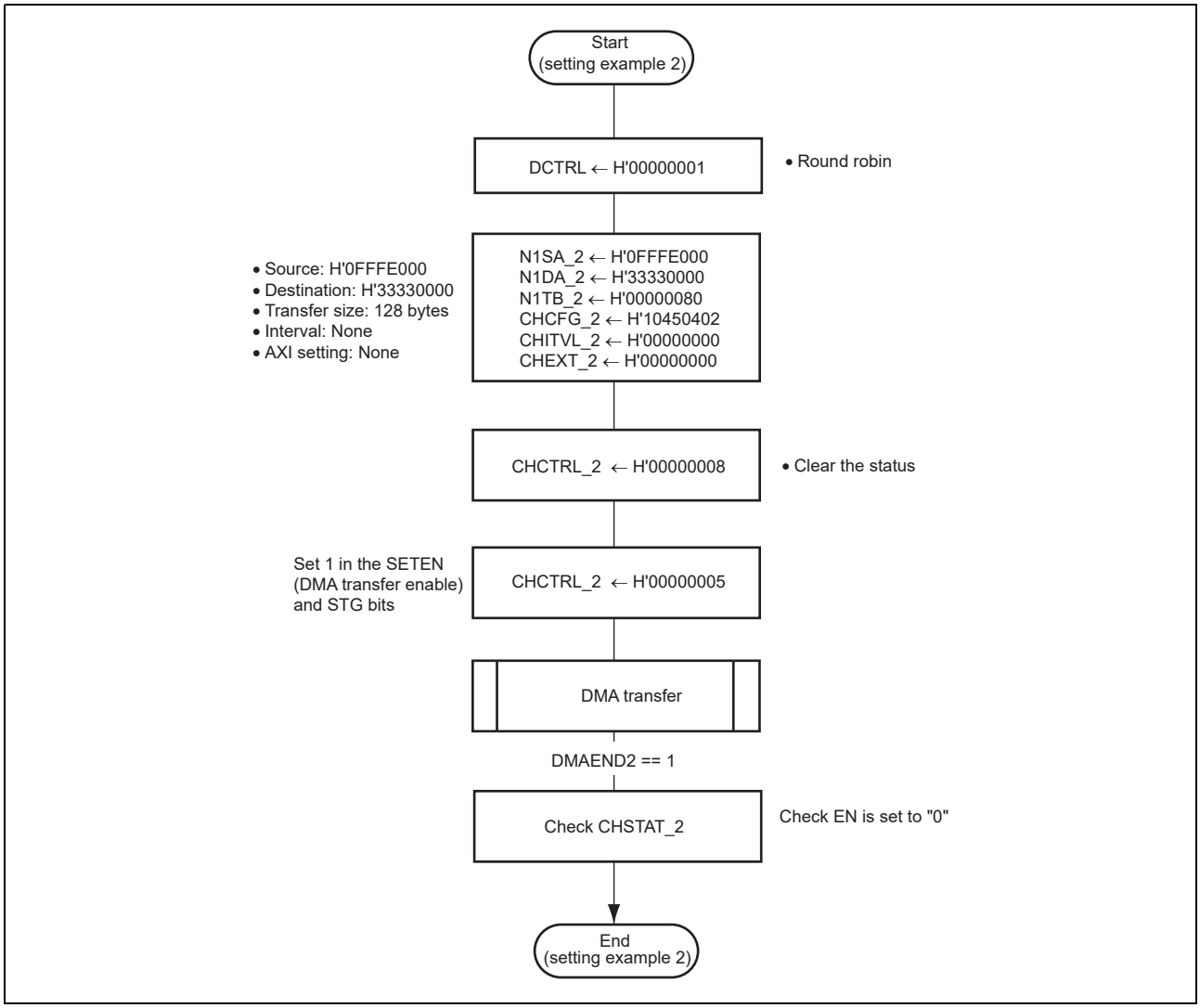


Figure 9.33 Setting Example 2

9.8.3 Setting Example 3 (Register Mode/Continuous Execution)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 9.24 DMA Transfer Setting Example 3

Item		Description	
Channel used		1	
Priority control		Round robin	
DMA mode		Register	
Transfer mode		Block transfer	
Register set used		Use Next0 and then Next1 continuously	
Next0		Source	Destination
	Start address	H'11110000	H'33330000
	Address direction	Fixed	Fixed
	Data size	32 bits	512 bits
	DMA transfer byte count	512 bytes	
Next1		Source	Destination
	Start address	H'22220000	H'44440000
	Address direction	Fixed	Fixed
	Data size	32 bits	512 bits
	DMA transfer byte count	2048 bytes	
DMA transfer request		Auto request	
DMAACK signal		Not output	
DMA transfer end interrupt mask		Mask the DMA transfer end interrupt upon completion of Next0	
CACHE setting		Default value	

Setting example 3

DCTRL = H'00000001 (DMA setting)

N0SA = H'11110000 (source address)
N0DA = H'33330000 (destination address)
N0TB = H'00000200 (transfer byte count)

N1SA = H'22220000 (source address)
N1DA = H'44440000 (destination address)
N1TB = H'00000800 (transfer byte count)

CHCFG = H'61762001 (configuration)
CHITVL = H'00000000 (interval)
CHEXT = H'00000000 (CACHE setting)

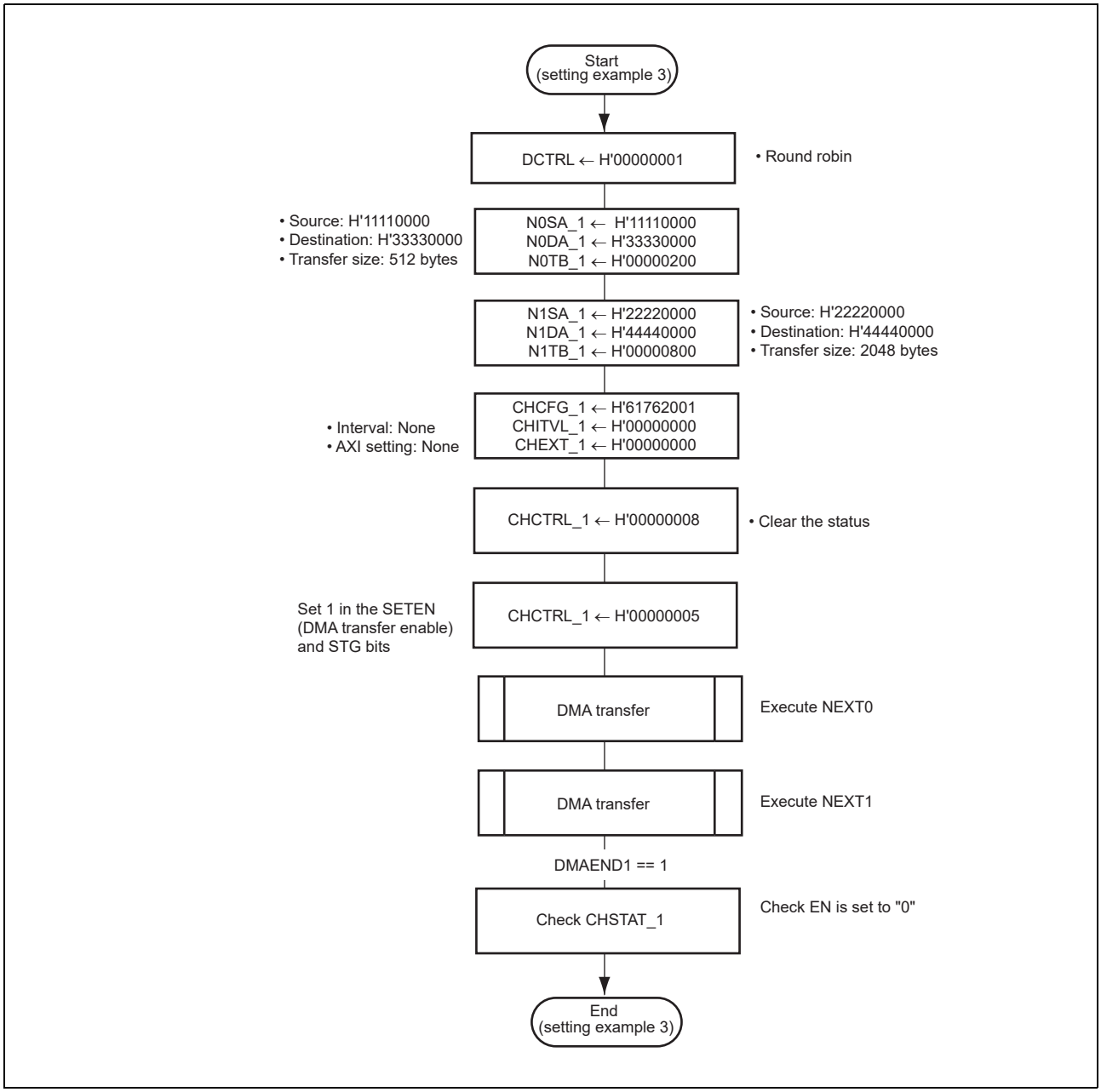


Figure 9.34 Setting Example 3

9.8.4 Setting Example 4 (Link Mode)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 9.25 DMA Transfer Setting Example 4

Item	Description
Channel used	0
Priority control	Round robin
DMA mode	Link
Transfer mode	Block transfer
Register set used	—
Descriptor start address	H'00001000

Table 9.26 DMA Transfer Setting Example 4 (Descriptor 1)

Item		Description	
Descriptor start address		H'00001000	
Next descriptor start address		H'00002000	
Transfer mode		Block transfer	
Next0		Source	Destination
	Start address	H'11110000	H'33330000
	Address direction	Increment	Increment
	Data size	32 bits	32 bits
	DMA transfer byte count	2048 bytes	
DMA transfer request		Auto request trigger (STG)	
DMAACK signal		Not output	
DMA transfer end interrupt mask		Masked	
CACHE setting		Default value	
header			
	DMA interrupt when LV = 1	Issued (DIM = 0)	
	LV writeback	Done (WBD = 0)	
	Next link address	Available (LE = 0)	
	Descriptor valid	Valid (LV = 1)	

Table 9.27 DMA Transfer Setting Example 4 (Descriptor 2)

Item		Description	
Descriptor start address		H'00002000	
Next descriptor start address		H'00005000	
Transfer mode		Block transfer	
Next0		Source	Destination
	Start address	H'44440000	H'55550000
	Address direction	Increment	Increment
	Data size	64 bits	256 bits
	DMA transfer byte count	1024 bytes	
DMA transfer request		Auto request trigger (STG)	
DMAACK signal		Not output	
DMA transfer end interrupt mask		Masked	
CACHE setting		Default value	
header			
	DMA interrupt when LV = 1	Issued (DIM = 0)	
	LV writeback	Done (WBD = 0)	
	Next link address	Available (LE = 0)	
	Descriptor valid	Valid (LV = 1)	

Table 9.28 DMA Transfer Setting Example 4 (Descriptor 3)

Item		Description	
Descriptor start address		H'00005000	
Next descriptor start address		—	
Transfer mode		Block transfer	
Next0		Source	Destination
	Start address	H'77770000	H'AAAA0000
	Address direction	Increment	Increment
	Data size	512 bits	512 bits
	DMA transfer byte count	4096 bytes	
DMA transfer request		Auto request trigger (STG)	
DMAACK signal		Not output	
DMA transfer end interrupt mask		Not masked	
CACHE setting		Default value	
header			
	DMA interrupt when LV = 1	Issued (DIM = 0)	
	LV writeback	Done (WBD = 0)	
	Next link address	Not available (LE = 1)	
	Descriptor valid	Valid (LV = 1)	

Setting example 4

DCTRL= H'00000001 (DMA setting)
NXLA = H'00001000 (descriptor start address)
CHCFG = H'80000000 (configuration)

Table 9.29 Descriptor Setting

	Descriptor 1	Descriptor 2	Descriptor 3
header	H'00000001	H'00000001	H'00000003
SA (Source Address)	H'11110000	H'44440000	H'77770000
DA (Destination Address)	H'33330000	H'55550000	H'AAAA0000
TB (Transaction Byte)	H'00000800	H'00000400	H'00001000
CFG (Configuration)	H'81422008	H'81453008	H'80466008
ITVL (Interval)	H'00000000	H'00000000	H'00000000
EXT (Extension)	H'00000000	H'00000000	H'00000000
NXLA (Next Link Address)	H'00002000	H'00005000	H'00000000

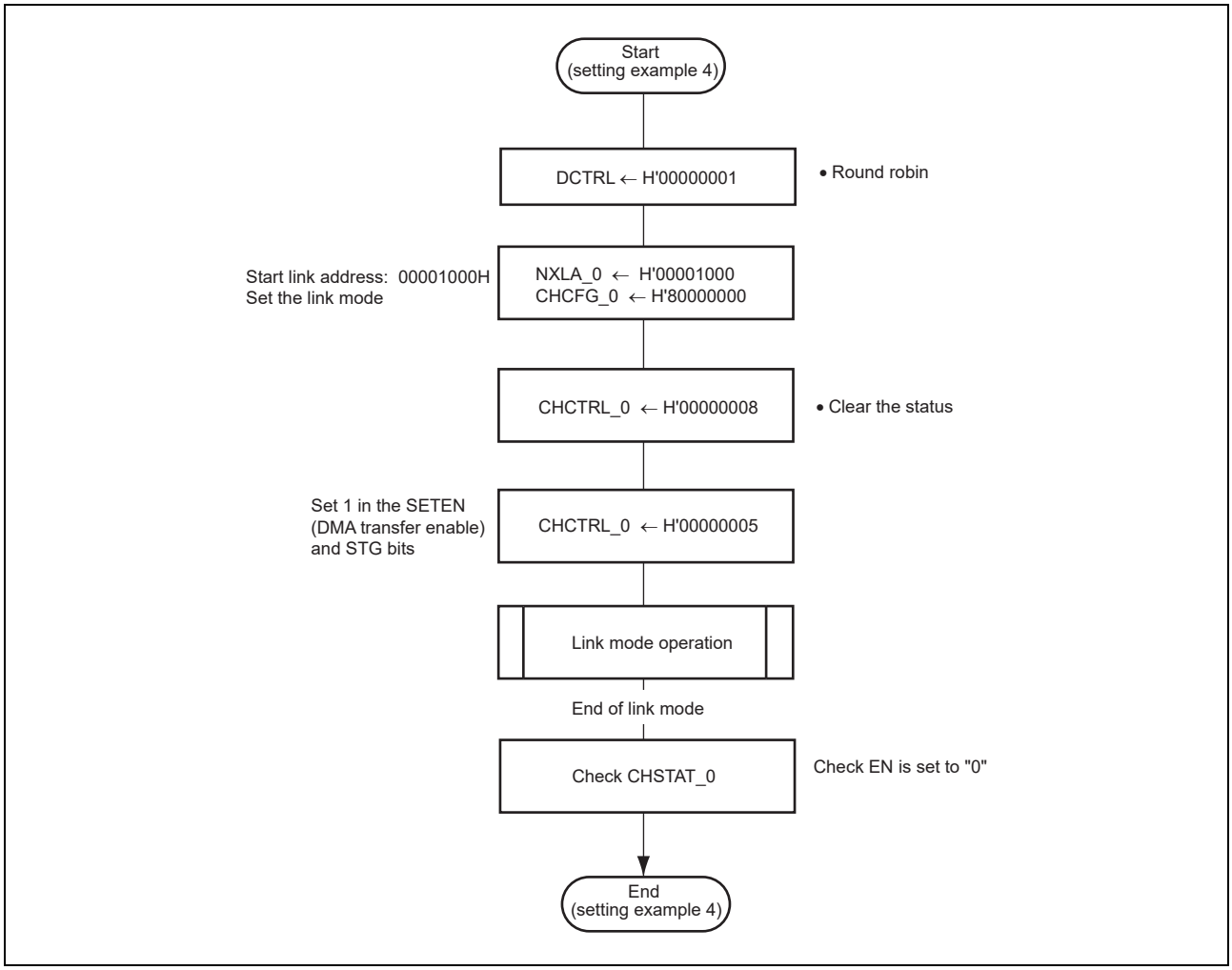


Figure 9.35 Setting Example 4

9.8.5 Next Register Set Continuous Execution Setting

The following figure shows the flowchart for executing DMA transfers continuously by using two Next register sets in register mode. While a DMA transaction is being executed using one Next register set, the other Next register set is set in order to continue to execute DMA transfers.

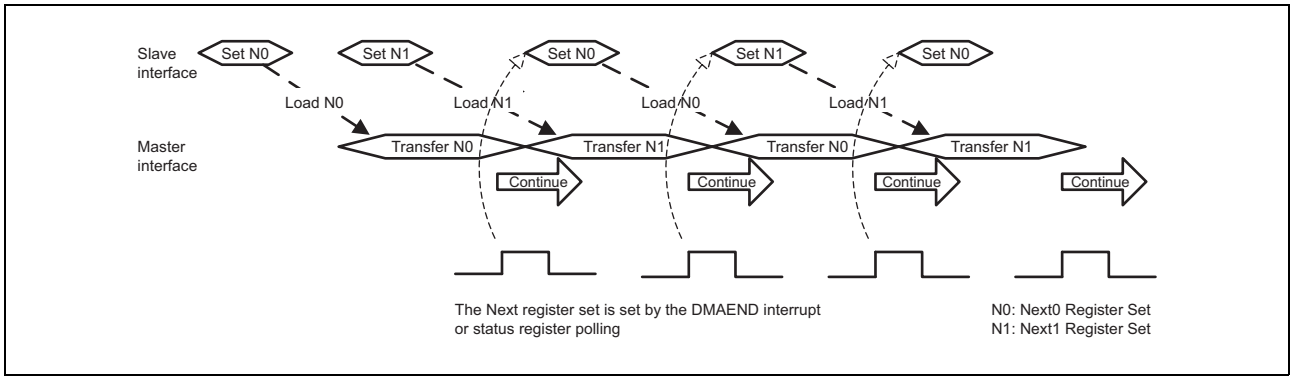


Figure 9.36 Image of Next Register Set Continuous Execution

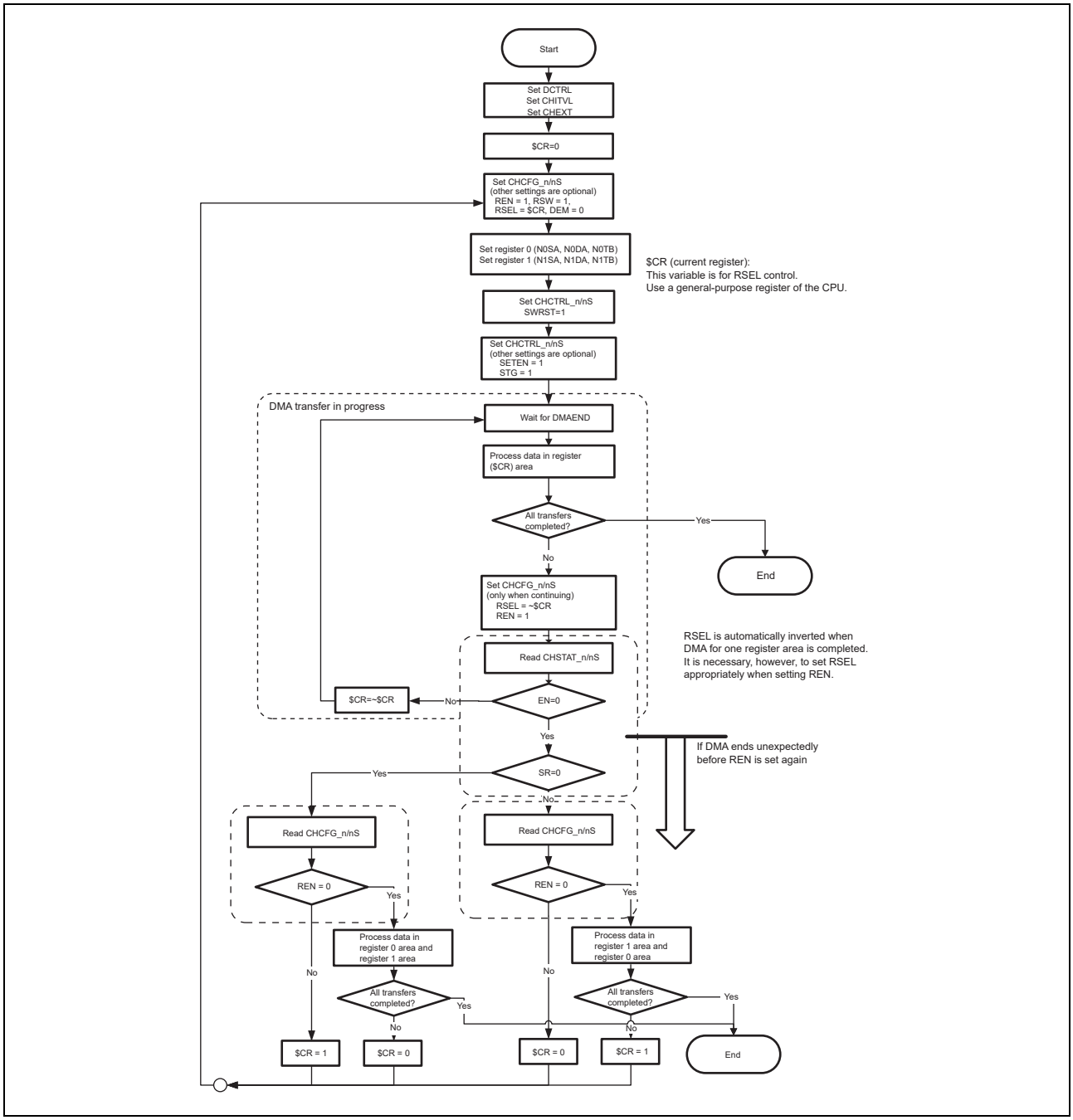


Figure 9.37 Example of Continuous DMA Execution by Using a Next Register Set

- Supplementary information
  First, save the data of the register sets to be used for DMA transfers (0 (N0SA, N0DA, and N0TB) and 1 (N1SA, N1DA, and N1TB)) to a general-purpose register of the CPU (the values of this register is referred to as $CR for the sake of convenience).
  Each time the DMA transfers for one register set are completed (the DMA transfer end interrupt is output), REN is automatically cleared to 0. In order to continue to execute DMA transfers, it is necessary to set REN of the CHCFG_n/nS register every time the DMA transfer end interrupt is asserted. This register also contains the RSEL bit, and the value of this bit needs to be set appropriately as well. Therefore, use $CR.
  In this mode, two Next register sets are executed continuously. However, if CLREN is not set before the DMA transaction is completed (the next DMA transfer end interrupt is output), continuous execution stops. In this case, how much of data has been transferred can be checked by reading the SR and EN bits of the CHSTAT_n/nS register and the REN bit of the CHCFG_n/nS register. To restart the DMA transaction, follow the flowchart shown above.

9.9 Note

9.9.1 Divided Output of DACK0 and TEND0

When transferring 4 bytes or more to an 8-bit or 16-bit external device or transferring 2 bytes or more to an 8-bit external device, each DMA transfer unit is divided into multiple bus cycles. Note that, if the setting is such that DMA transfer is divided into multiple bus cycles and CS# is negated between bus cycles, the DACK0 output and the TEND0 output is divided to align data as with CS#. Figure 9.38 shows an example.

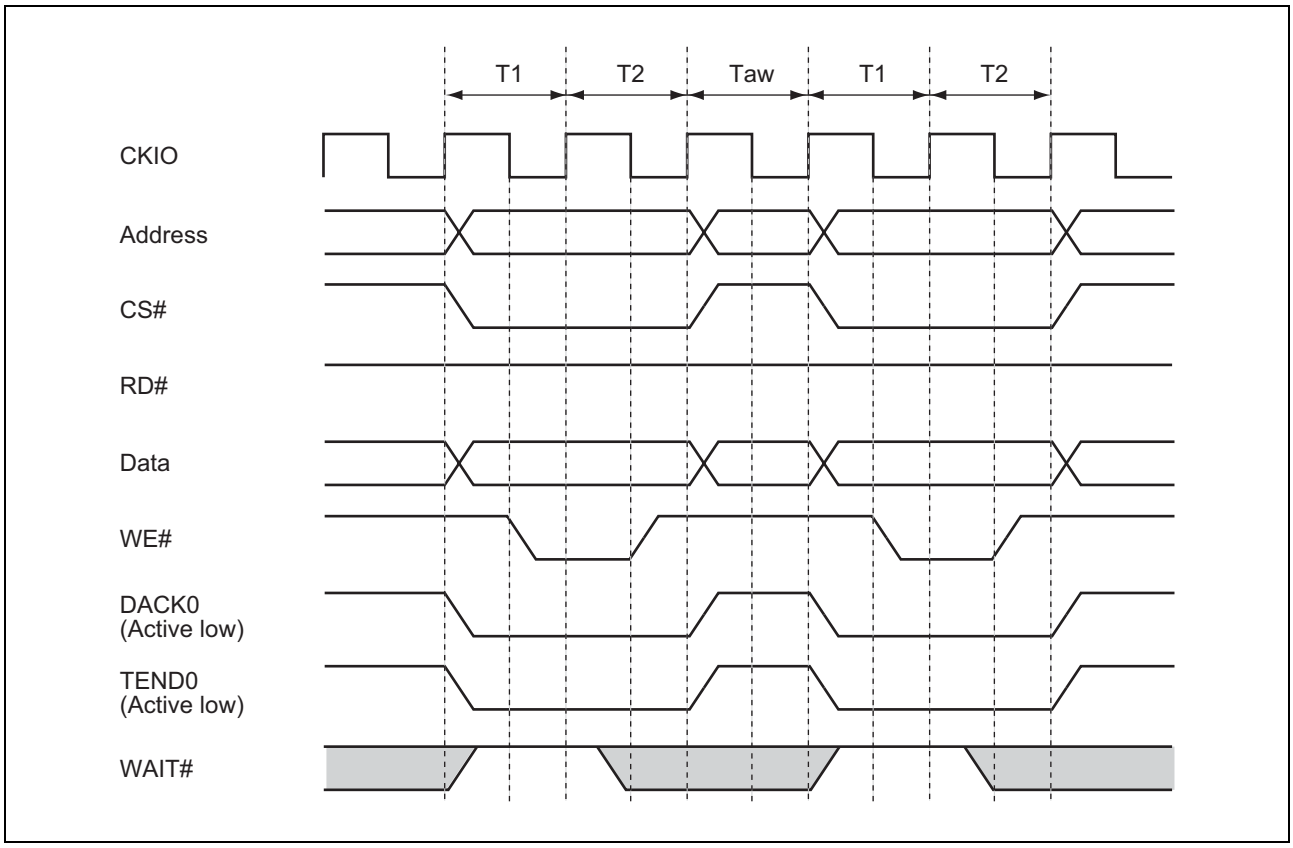


Figure 9.38 Example of TEND0 Divided Output Timing

9.9.2 TEND0 Not Output

Note that TEND0 may not be output depending on the combination of the bits DDS[3:0], SDS[3:0] and REQD in the CHCFG_0/0S register.

Table 9.30 shows when TEND0 is not output and Figure 9.39 shows an operation example.

Table 9.30 Bit Combination when TEND0 Is Not Output

CHCFG_0/0S Register			
REQD	DDS	SDS	TEND0 Output
1	—	—	Output
0	DDS > SDS		Output
	DDS = SDS		Output
	DDS < SDS		Not output

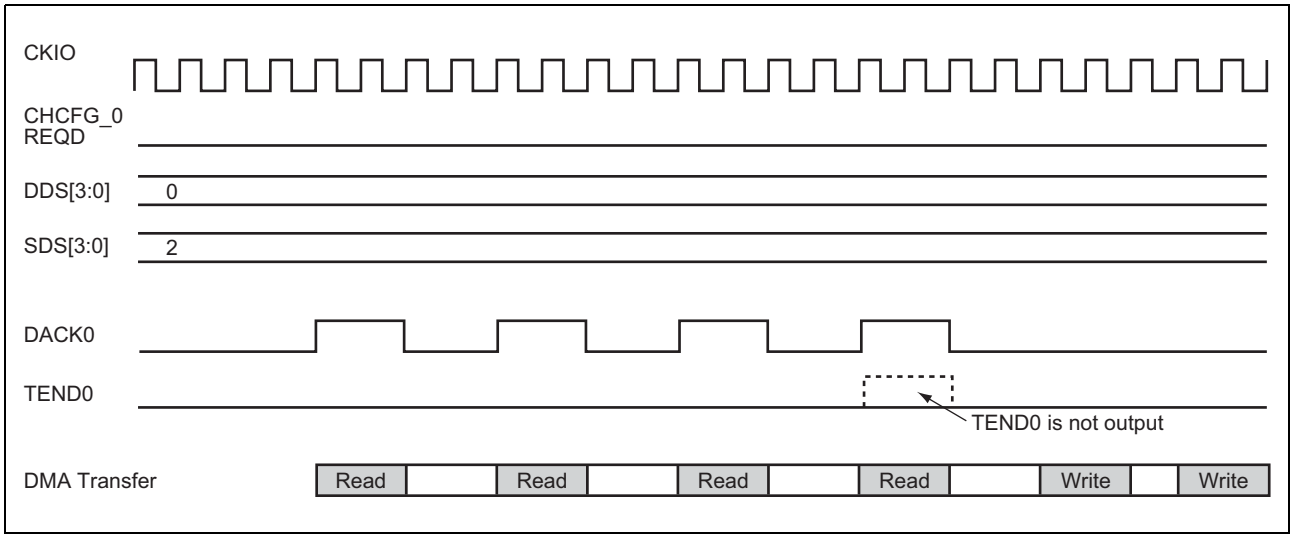


Figure 9.39 TEND0 Not Output

9.9.3 Atomic Access (ARLOCK[1:0] and AWLOCK[1:0])

This module does not support atomic (locked or exclusive) access, that is, it only supports normal access.
Signals ARLOCK[1:0] and AWLOCK[1:0] are fixed as follows and cannot be modified.
ARLOCK[1:0], AWLOCK[1:0]: 00 (normal access)

10. Multi-Function Timer Pulse Unit 3 (MTU3a)

10.1 Overview

This LSI has an on-chip multi-function timer pulse unit 3 (MTU3a), consisting of eight 16-bit timer channels and one 32-bit timer channel.

Table 10.1 shows the specifications of the MTU and Table 10.2 lists the functions of the MTU. Figure 10.1 and Figure 10.2 are block diagrams of the MTU.

Table 10.1 MTU Specifications

Item	Description
Pulse input/output	28 lines max.
Pulse input	3 lines
Count clock	11 clocks for each channel (14 clocks for MTU0, 12 clocks for MTU2, and 10 clocks for MTU5, four clocks for MTU1-MTU2 combination (when LWA = 1))
Operating frequency	Up to 66 MHz
Available operations	[MTU0 to MTU4, MTU6, MTU7, and MTU8] • Waveform output on compare match • Input capture function (noise filter setting available) • Counter-clearing operation • Simultaneous writing to multiple timer counters (TCNT) (excluding MTU8) • Simultaneous clearing on compare match or input capture (excluding MTU8) • Simultaneous input and output to registers in synchronization with counter operations (excluding MTU8) • Up to 12-phase PWM output in combination with synchronous operation (excluding MTU8)
	[MTU0 MTU3, MTU4, MTU6, MTU7, and MTU8] • Buffer operation specifiable
	[MTU1, MTU2] • Phase counting mode can be specified independently • 32-bit phase counting mode can be specified for interlocked operation of MTU1 and MTU2 (when TMDR3.LWA = 1) • Cascade connection operation available
	[MTU3, MTU4, MTU6, and MTU7] • Through interlocked operation of MTU3/4 and MTU6/7, the positive and negative signals in six phases (12 phases in total) can be output in complementary PWM and reset-synchronized PWM operation. • In complementary PWM mode, values can be transferred from buffer registers to temporary registers at crests and troughs of the timer-counter values or when the buffer registers (TGRD registers in MTU4 and MTU7) are written to. • Double-buffering selectable in complementary PWM mode
	[MTU3 and MTU4] • Through interlocking with MTU0, a mode for driving AC synchronous motors (brushless DC motors) by using complementary PWM output and reset-synchronized PWM output is settable and allows the selection of two types of waveform output (chopping or level)
	[MTU5] • Capable of operation as a dead-time compensation counter
	[MTU0/MTU5, MTU1, MTU2, and MTU8] 32-bit phase counting mode specifiable by combining MTU1 and MTU2 and through interlocked operation with MTU0/MTU5 and MTU8
Interrupt-skipping function	• In complementary PWM mode, interrupts on crests and troughs of counter values and triggers to start conversion by the A/D converter can be skipped
Interrupt sources	43 sources
Buffer operation	Automatic transfer of register data (transfer from the buffer register to the timer register)
Trigger generation	A/D converter start triggers can be generated
	A/D converter start request delaying function enables A/D converter to be started with any desired timing and to be synchronized with PWM output
Low power consumption function	The MTU3a can be placed in the module-stop state.

Table 10.2 MTU Functions (1/2)

Item		MTU0	MTU1	MTU2	MTU1 & MTU2 (LWA = 1)	MTU3	MTU4	MTU5	MTU6	MTU7	MTU8
Count clock		P1φ/1 P1φ/2 P1φ/4 P1φ/8 P1φ/16 P1φ/32 P1φ/64 P1φ/256 P1φ/1024 MTCLKA MTCLKB MTCLKC MTCLKD MTIOC1A	P1φ/1 P1φ/2 P1φ/4 P1φ/8 P1φ/16 P1φ/32 P1φ/64 P1φ/256 P1φ/1024 MTCLKA MTCLKB	P1φ/1 P1φ/2 P1φ/4 P1φ/8 P1φ/16 P1φ/32 P1φ/64 P1φ/256 P1φ/1024 MTCLKA MTCLKB MTCLKC	MTCLKA MTCLKB MTCLKC MTCLKD	P1φ/1 P1φ/2 P1φ/4 P1φ/8 P1φ/16 P1φ/32 P1φ/64 P1φ/256 P1φ/1024 MTCLKA MTCLKB	P1φ/1 P1φ/2 P1φ/4 P1φ/8 P1φ/16 P1φ/32 P1φ/64 P1φ/256 P1φ/1024 MTCLKA MTCLKB	P1φ/1 P1φ/2 P1φ/4 P1φ/8 P1φ/16 P1φ/32 P1φ/64 P1φ/256 P1φ/1024 MTIOC1A	P1φ/1 P1φ/2 P1φ/4 P1φ/8 P1φ/16 P1φ/32 P1φ/64 P1φ/256 P1φ/1024 MTCLKA MTCLKB	P1φ/1 P1φ/2 P1φ/4 P1φ/8 P1φ/16 P1φ/32 P1φ/64 P1φ/256 P1φ/1024 MTCLKA MTCLKB	P1φ/1 P1φ/2 P1φ/4 P1φ/8 P1φ/16 P1φ/32 P1φ/64 P1φ/256 P1φ/1024 MTCLKA MTCLKB
External clocks in phase-counting mode		—	MTCLKA MTCLKB	MTCLKA MTCLKB MTCLKC MTCLKD	MTCLKA MTCLKB MTCLKC MTCLKD	—	—	—	—	—	—
General registers (TGR)		TGRA TGRB TGRE	TGRA TGRB	TGRA TGRB	TGRALW TGRBLW	TGRA TGRB	TGRA TGRB	TGRU TGRV TGRW	TGRA TGRB	TGRA TGRB	TGRA TGRB
General registers/ buffer registers		TGRC TGRD TGRF	—	—	—	TGRC TGRD TGRE	TGRC TGRD TGRE TGRF	—	TGRC TGRD TGRE	TGRC TGRD TGRE TGRF	TGRC TGRD
I/O pins		MTIOC0A MTIOC0B MTIOC0C MTIOC0D	MTIOC1A MTIOC1B	MTIOC2A MTIOC2B	MTIOC1A MTIOC1B	MTIOC3A MTIOC3B MTIOC3C MTIOC3D	MTIOC4A MTIOC4B MTIOC4C MTIOC4D	MTIC5U MTIC5V MTIC5W	MTIOC6A MTIOC6B MTIOC6C MTIOC6D	MTIOC7A MTIOC7B MTIOC7C MTIOC7D	MTIOC8A MTIOC8B MTIOC8C MTIOC8D
Counter clear function		TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGRALW/ TGRBLW input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture
Compare match output	0 output	√	√	√	—	√	√	—	√	√	√
	1 output	√	√	√	—	√	√	—	√	√	√
	Toggle output	√	√	√	—	√	√	—	√	√	√
Input capture function		√	√	√	√*1	√	√	√	√	√	√*2
Synchronous operation		√	√	√	—	√	√	—	√	√	—
PWM mode 1		√	√	√	—	√	√	—	√	√	—
PWM mode 2		√	√	√	—	—	—	—	—	—	—
Complementary PWM mode		—	—	—	—	√	√	—	√	√	—
Reset-synchronized PWM mode		—	—	—	—	√	√	—	√	√	—
AC synchronous motor drive mode		√	—	—	—	√	√	—	—	—	—
Phase counting mode		—	√	√	√	—	—	—	—	—	—
Buffer operation		√	—	—	—	√	√	—	√	√	√
Dead time compensation counter function		—	—	—	—	—	—	√	—	—	—
DMAC activation		TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGRALW/ TGRBLW input capture	TGR compare match or input capture	TGR compare match or input capture and TCNT overflow/ underflow*3	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture and TCNT overflow/ underflow*3	TGR compare match or input capture

Table 10.2 MTU Functions (2/2)

Item	MTU0	MTU1	MTU2	MTU1 & MTU2 (LWA = 1)	MTU3	MTU4	MTU5	MTU6	MTU7	MTU8
A/D converter start trigger	TGRA compare match or input capture TGRE compare match	TGRA compare match or input capture	TGRA compare match or input capture	TGRALW input capture	TGRA compare match or input capture	TGRA compare match or input capture, or TCNT underflow (trough) in complementary PWM mode	—	TGRA compare match or input capture	TGRA compare match or input capture, or TCNT underflow (trough) in complementary PWM mode	—
Interrupt sources	Seven sources •Compare match or input capture 0A •Compare match or input capture 0B •Compare match or input capture 0C •Compare match or input capture 0D •Compare match0E •Compare match0F •Overflow	Four sources •Compare match or input capture 1A •Compare match or input capture 1B •Overflow •Underflow	Four sources •Compare match or input capture 2A •Compare match or input capture 2B •Overflow •Underflow	Four sources • Input capture 1A • Input capture 1B •Overflow •Underflow	Five sources •Compare match or input capture 3A •Compare match or input capture 3B •Compare match or input capture 3C •Compare match or input capture 3D •Overflow	Five sources •Compare match or input capture 4A •Compare match or input capture 4B •Compare match or input capture 4C •Compare match or input capture 4D •Overflow or underflow*3	Three sources •Compare match or input capture 5U •Compare match or input capture 5V •Compare match or input capture 5W	Five sources •Compare match or input capture 6A •Compare match or input capture 6B •Compare match or input capture 6C •Compare match or input capture 6D •Overflow	Five sources •Compare match or input capture 7A •Compare match or input capture 7B •Compare match or input capture 7C •Compare match or input capture 7D •Overflow or underflow*3	Five sources •Compare match or input capture 8A •Compare match or input capture 8B •Compare match or input capture 8C •Compare match or input capture 8D •Overflow
A/D converter start request delaying function	—	—	—	—	—	•A/D converter start request at a match between TADCORA and TCNT or A/D converter start request at a match between TADCORB and TCNT	—	—	•A/D converter start request at a match between TADCORA and TCNT or A/D converter start request at a match between TADCORB and TCNT	—
Interrupt skipping 1	—	—	—	—	•Skips TGRA compare match interrupts	•Skips TCIV interrupts	—	•Skips TGRA compare match interrupts	•Skips TCIV interrupts	—
Interrupt skipping 2	—	—	—	—	—	•Skipping in compare count between TADCORA and TCNT, and TADCORB and TCNT	—	—	•Skipping in compare count between TADCORA and TCNT, and TADCORB and TCNT	—
Module stop function	STBCR3.MSTP33*4									

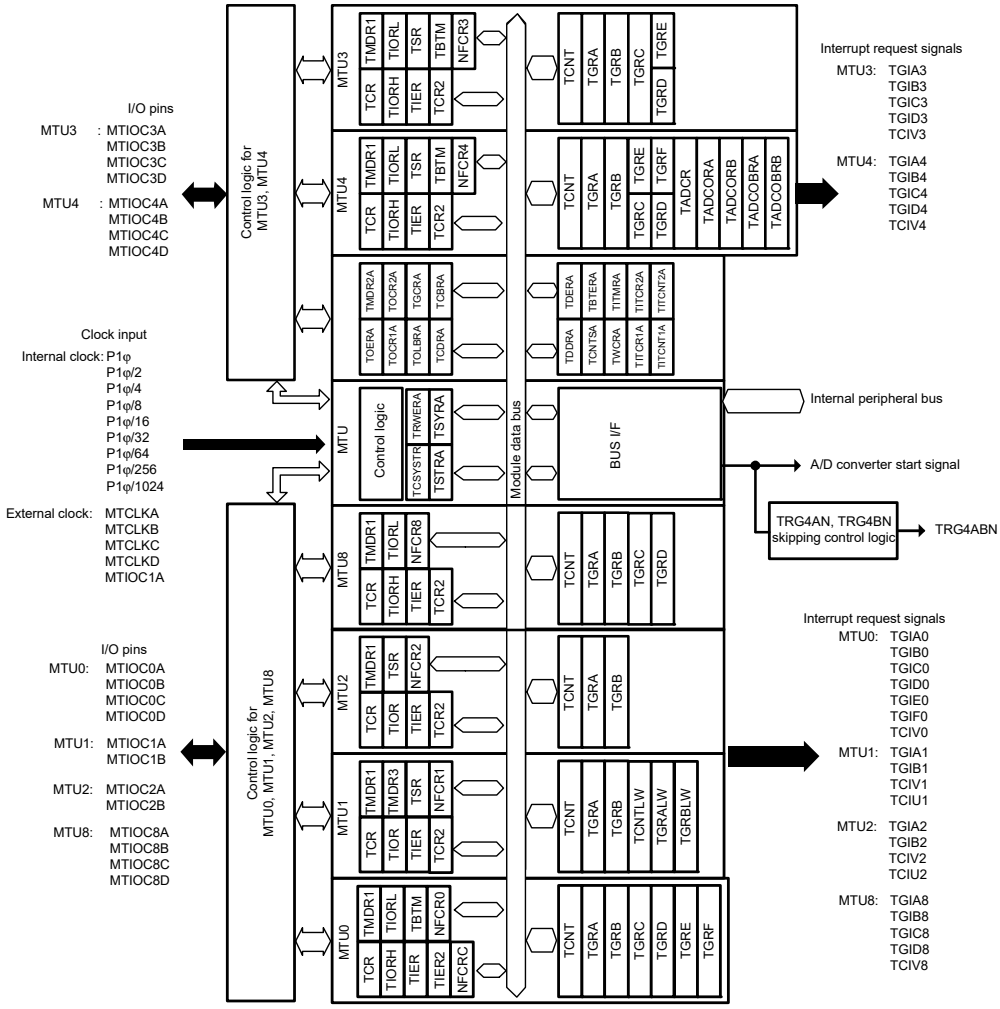
√: Possible —: Not possible

Note 1. When LWA is 1, the TGRALW capture source can be selected from either of the following: an input from MTIOC1A or MTU0.TGRA compare match/input capture event.
The TGRBLW capture source can be selected from any of the following: an input from MTIOC1B, MTU0.TGRC compare match/input capture event, or MTU8.TGRC compare match event.

Note 2. Capture in MTU8 is supported only in normal mode.

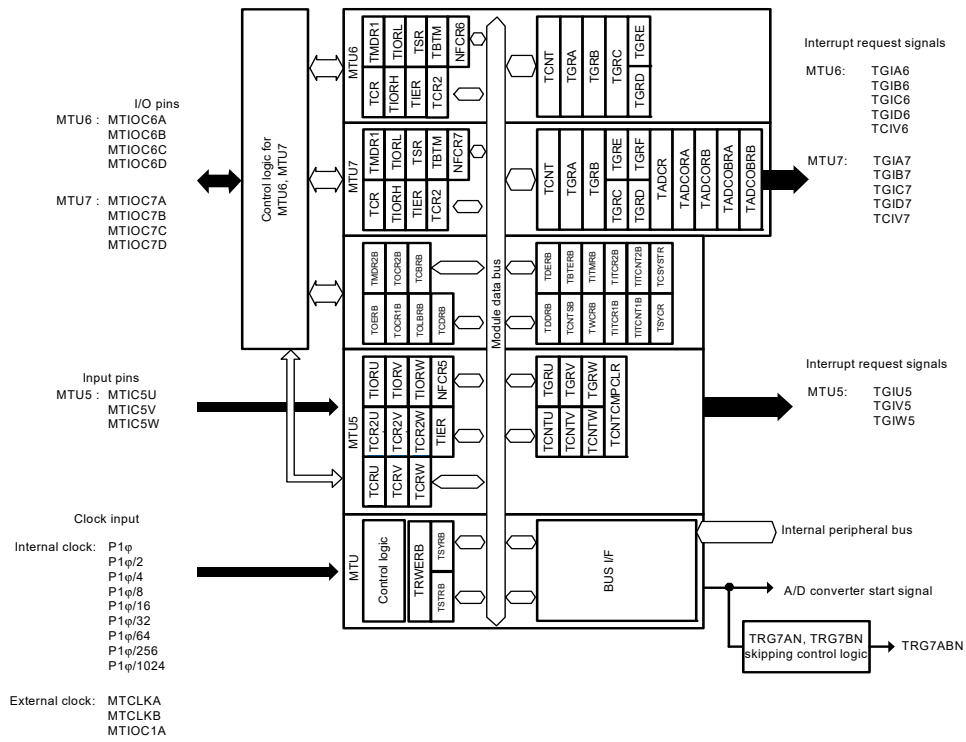
Note 3. The underflow interrupt source is valid only in complementary PWM mode.

Note 4. For details on the module stop function, refer to section 52, Power-Down Modes.



TCR: Timer control register
TCR2: Timer control register 2
TMDR: Timer mode register 1
TMDR2A: Timer mode register 2
TMDR3: Timer mode register 3
TIOR: Timer I/O control register
TIORH: Timer I/O control register H
TIORL: Timer I/O control register L
TIER: Timer interrupt enable register
TIER2: Timer interrupt enable register 2
TSR: Timer status register
TBTM: Timer buffer operation transfer mode register
TCNT: Timer counter
TCNTLW: Timer longword counter
TGRA: Timer general register A
TGRALW: Timer longword general register A
TGRB: Timer general register B
TGRBLW: Timer longword general register B
TGRC: Timer general register C
TGRD: Timer general register D
TGRE: Timer general register E
TGRF: Timer general register F
TSTRA: Timer start register
TSYRA: Timer synchronous register
TCSYSTR: Timer counter synchronous start register
TRWERA: Timer read/write enable register
TOERA: Timer output master enable register
TOCR1A: Timer output control register 1
TOCR2A: Timer output control register 2
TOLBRA: Timer output level buffer register
TGCRA: Timer gate control register A
TCNTSA: Timer subcounter
TCDRA: Timer cycle data register
TCBRA: Timer cycle buffer register
TDDRA: Timer dead time data register
TDERA: Timer dead time enable register
TBTERA: Timer buffer transfer set register
TWCRA: Timer waveform control register
TICCR: Timer input capture control register
TADCR: Timer A/D converter start request control register
TADCORA: Timer A/D converter start request cycle set register A
TADCORB: Timer A/D converter start request cycle set register B
TADCOBRA: Timer A/D converter start request cycle set buffer register A
TADCOBRB: Timer A/D converter start request cycle set buffer register B
TITMRA: Timer interrupt skipping mode register
TITCR1A: Timer interrupt skipping set register 1
TITCR2A: Timer interrupt skipping set register 2
TITCNT1A: Timer interrupt skipping counter 1
TITCNT2A: Timer interrupt skipping counter 2
NFCRn: Noise filter control register n

Figure 10.1 Block Diagram of MTU (MTU0 to MTU4, MTU8)



TCR: Timer control register
TCRU: Timer control register U
TCR2U: Timer control register 2U
TCRV: Timer control register V
TCR2V: Timer control register 2V
TCRW: Timer control register W
TCR2W: Timer control register 2W
TMDR1: Timer mode register 1
TMDR2B: Timer mode register 2
TIORH: Timer I/O control register H
TIORL: Timer I/O control register L
TIORU: Timer I/O control register U
TIORV: Timer I/O control register V
TIORW: Timer I/O control register W
TIER: Timer interrupt enable register
TSR: Timer status register
TCNT: Timer counter
TCNTU: Timer counter U
TCNTV: Timer counter V
TCNTW: Timer counter W
TGRA: Timer general register A
TGRB: Timer general register B
TGRC: Timer general register C
TGRD: Timer general register D
TGRE: Timer general register E
TGRF: Timer general register F
TGRU: Timer general register U
TGRV: Timer general register V
TGRW: Timer general register W
TCNTCMPCLR: Timer compare match clear register

TSTRB: Timer start register
TSYRB: Timer synchronous register
TSYCR: Timer synchronous clear register
TCSYSTR: Timer counter synchronous start register
TRWERB: Timer read/write enable register
TOERB: Timer output master enable register
TOCR1B: Timer output control register 1
TOCR2B: Timer output control register 2
TOLBRB: Timer output level buffer register
TCDRB: Timer cycle data register
TCBRB: Timer cycle buffer register
TDDRB: Timer dead time data register
TDERB: Timer dead time enable register
TCNTSB: Timer subcounter
TBTERB: Timer buffer transfer set register
TWCRB: Timer waveform control register
TADCR: Timer A/D converter start request control register
TADCORA: Timer A/D converter start request cycle set register A
TADCORB: Timer A/D converter start request cycle set register B
TADCOBRA: Timer A/D converter start request cycle set buffer register A
TADCOBRB: Timer A/D converter start request cycle set buffer register B
TITMRB: Timer interrupt skipping mode register
TITCR1B: Timer interrupt skipping set register 1
TITCR2B: Timer interrupt skipping set register 2
TITCNT1B: Timer interrupt skipping counter 1
TITCNT2B: Timer interrupt skipping counter 2
NFCRn: Noise filter control register n

Figure 10.2 Block Diagram of MTU (MTU5 to MTU7)

Table 10.3 shows the configuration of pins for the MTU.

Table 10.3 Pin Configuration of the MTU

Channel	Pin Name	I/O	Function
MTU	MTCLKA	Input	External clock A input pin (MTU1/MTU2 phase counting mode A phase input)
	MTCLKB	Input	External clock B input pin (MTU1/MTU2 phase counting mode B phase input)
	MTCLKC	Input	External clock C input pin (MTU2 phase counting mode A phase input)
	MTCLKD	Input	External clock D input pin (MTU2 phase counting mode B phase input)
MTU0	MTIOC0A	I/O	MTU0 TGRA input capture input/output compare output/PWM output pin
	MTIOC0B	I/O	MTU0 TGRB input capture input/output compare output/PWM output pin
	MTIOC0C	I/O	MTU0 TGRC input capture input/output compare output/PWM output pin
	MTIOC0D	I/O	MTU0 TGRD input capture input/output compare output/PWM output pin
MTU1	MTIOC1A	I/O	MTU1 TGRA input capture input/output compare output/PWM output pin
	MTIOC1B	I/O	MTU1 TGRB input capture input/output compare output/PWM output pin
MTU2	MTIOC2A	I/O	MTU2 TGRA input capture input/output compare output/PWM output pin
	MTIOC2B	I/O	MTU2 TGRB input capture input/output compare output/PWM output pin
MTU3	MTIOC3A	I/O	MTU3 TGRA input capture input/output compare output/PWM output pin
	MTIOC3B	I/O	MTU3 TGRB input capture input/output compare output/PWM output pin
	MTIOC3C	I/O	MTU3 TGRC input capture input/output compare output/PWM output pin
	MTIOC3D	I/O	MTU3 TGRD input capture input/output compare output/PWM output pin
MTU4	MTIOC4A	I/O	MTU4 TGRA input capture input/output compare output/PWM output pin
	MTIOC4B	I/O	MTU4 TGRB input capture input/output compare output/PWM output pin
	MTIOC4C	I/O	MTU4 TGRC input capture input/output compare output/PWM output pin
	MTIOC4D	I/O	MTU4 TGRD input capture input/output compare output/PWM output pin
MTU5	MTIC5U	Input	MTU5 TGRU input capture input/external pulse input pin
	MTIC5V	Input	MTU5 TGRV input capture input/external pulse input pin
	MTIC5W	Input	MTU5 TGRW input capture input/external pulse input pin
MTU6	MTIOC6A	I/O	MTU6 TGRA input capture input/output compare output/PWM output pin
	MTIOC6B	I/O	MTU6 TGRB input capture input/output compare output/PWM output pin
	MTIOC6C	I/O	MTU6 TGRC input capture input/output compare output/PWM output pin
	MTIOC6D	I/O	MTU6 TGRD input capture input/output compare output/PWM output pin
MTU7	MTIOC7A	I/O	MTU7 TGRA input capture input/output compare output/PWM output pin
	MTIOC7B	I/O	MTU7 TGRB input capture input/output compare output/PWM output pin
	MTIOC7C	I/O	MTU7 TGRC input capture input/output compare output/PWM output pin
	MTIOC7D	I/O	MTU7 TGRD input capture input/output compare output/PWM output pin
MTU8	MTIOC8A	I/O	MTU8.TGRA input capture input/output compare output pin
	MTIOC8B	I/O	MTU8.TGRB input capture input/output compare output pin
	MTIOC8C	I/O	MTU8.TGRC input capture input/output compare output pin
	MTIOC8D	I/O	MTU8.TGRD input capture input/output compare output pin

10.2 Register Descriptions

Table 10.4 shows the register configuration.

Table 10.4 Register configuration

Channel	Register Name	Abbreviation	Address	Access size
MTU0	Timer control register	TCR	H'E8041300	8
	Timer mode register 1	TMDR1	H'E8041301	8
	Timer I/O control register H	TIORH	H'E8041302	8
	Timer I/O control register L	TIORL	H'E8041303	8
	Timer interrupt enable register	TIER	H'E8041304	8
	Timer counter	TCNT	H'E8041306	16
	Timer general register A	TGRA	H'E8041308	16
	Timer general register B	TGRB	H'E804130A	16
	Timer general register C	TGRC	H'E804130C	16
	Timer general register D	TGRD	H'E804130E	16
	Timer general register E	TGRE	H'E8041320	16
	Timer general register F	TGRF	H'E8041322	16
	Timer interrupt enable register 2	TIER2	H'E8041324	8
	Timer buffer operation transfer mode register	TBTM	H'E8041326	8
	Timer control register 2	TCR2	H'E8041328	8
	Noise filter control register 0	NFCR0	H'E8041290	8
	Noise filter control register C	NFCRC	H'E8041299	8
MTU1	Timer control register	TCR	H'E8041380	8
	Timer mode register 1	TMDR1	H'E8041381	8
	Timer I/O control register	TIOR	H'E8041382	8
	Timer interrupt enable register	TIER	H'E8041384	8
	Timer status register	TSR	H'E8041385	8
	Timer counter	TCNT	H'E8041386	16
	Timer general register A	TGRA	H'E8041388	16
	Timer general register B	TGRB	H'E804138A	16
	Timer input capture control register	TICCR	H'E8041390	8
	Timer mode register 3	TMDR3	H'E8041391	8
	Timer control register 2	TCR2	H'E8041394	8
	Timer longword counter	TCNTLW	H'E80413A0	32
	Timer longword general register A	TGRALW	H'E80413A4	32
	Timer longword general register B	TGRBLW	H'E80413A8	32
	Noise filter control register 1	NFCR1	H'E8041291	8
MTU2	Timer control register	TCR	H'E8041400	8
	Timer mode register 1	TMDR1	H'E8041401	8
	Timer I/O control register	TIOR	H'E8041402	8
	Timer interrupt enable register	TIER	H'E8041404	8
	Timer status register	TSR	H'E8041405	8
	Timer counter	TCNT	H'E8041406	16
	Timer general register A	TGRA	H'E8041408	16
	Timer general register B	TGRB	H'E804140A	16
	Timer control register 2	TCR2	H'E804140C	8
	Noise filter control register 2	NFCR2	H'E8041292	8

Channel	Register Name	Abbreviation	Address	Access size
MTU3	Timer control register	TCR	H'E8041200	8
	Timer mode register 1	TMDR1	H'E8041202	8
	Timer I/O control register H	TIORH	H'E8041204	8
	Timer I/O control register L	TIORL	H'E8041205	8
	Timer interrupt enable register	TIER	H'E8041208	8
	Timer counter	TCNT	H'E8041210	16
	Timer general register A	TGRA	H'E8041218	16
	Timer general register B	TGRB	H'E804121A	16
	Timer general register C	TGRC	H'E8041224	16
	Timer general register D	TGRD	H'E8041226	16
	Timer control register 2	TCR2	H'E804124C	8
	Timer general register E	TGRE	H'E8041272	16
	Timer status register	TSR	H'E804122C	8
	Timer buffer operation transfer mode register	TBTM	H'E8041238	8
	Noise filter control register 3	NFCR3	H'E8041293	8
MTU4	Timer control register	TCR	H'E8041201	8
	Timer mode register 1	TMDR1	H'E8041203	8
	Timer I/O control register H	TIORH	H'E8041206	8
	Timer I/O control register L	TIORL	H'E8041207	8
	Timer interrupt enable register	TIER	H'E8041209	8
	Timer counter	TCNT	H'E8041212	16
	Timer general register A	TGRA	H'E804121C	16
	Timer general register B	TGRB	H'E804121E	16
	Timer general register C	TGRC	H'E8041228	16
	Timer general register D	TGRD	H'E804122A	16
	Timer control register 2	TCR2	H'E804124D	8
	Timer general register E	TGRE	H'E8041274	16
	Timer general register F	TGRF	H'E8041276	16
	Timer status register	TSR	H'E804122D	8
	Timer buffer operation transfer mode register	TBTM	H'E8041239	8
	Timer A/D converter start request control register	TADCR	H'E8041240	16
	Timer A/D converter start request cycle set register A	TADCORA	H'E8041244	16
	Timer A/D converter start request cycle set register B	TADCORB	H'E8041246	16
	Timer A/D converter start request cycle set buffer register A	TADCOBRA	H'E8041248	16
	Timer A/D converter start request cycle set buffer register B	TADCOBRB	H'E804124A	16
	Noise filter control register 4	NFCR4	H'E8041294	8
MTU5	Timer counter U	TCNTU	H'E8041C80	16
	Timer general register U	TGRU	H'E8041C82	16
	Timer control register U	TCRU	H'E8041C84	8
	Timer control register 2U	TCR2U	H'E8041C85	8
	Timer I/O control register U	TIORU	H'E8041C86	8
	Timer counter V	TCNTV	H'E8041C90	16
	Timer general register V	TGRV	H'E8041C92	16
	Timer control register V	TCRV	H'E8041C94	8
	Timer control register 2V	TCR2V	H'E8041C95	8
	Timer I/O control register V	TIORV	H'E8041C96	8

Channel	Register Name	Abbreviation	Address	Access size
MTU5	Timer counter W	TCNTW	H'E8041CA0	16
	Timer general register W	TGRW	H'E8041CA2	16
	Timer control register W	TCRW	H'E8041CA4	8
	Timer control register 2W	TCR2W	H'E8041CA5	8
	Timer I/O control register W	TIORW	H'E8041CA6	8
	Timer interrupt enable register	TIER	H'E8041CB2	8
	Timer start register	TSTR	H'E8041CB4	8
	Timer compare match clear register	TCNTCMPCLR	H'E8041CB6	8
	Noise filter control register 5	NFCR5	H'E8041A95	8
MTU6	Timer control register	TCR	H'E8041A00	8
	Timer mode register 1	TMDR1	H'E8041A02	8
	Timer I/O control register H	TIORH	H'E8041A04	8
	Timer I/O control register L	TIORL	H'E8041A05	8
	Timer interrupt enable register	TIER	H'E8041A08	8
	Timer counter	TCNT	H'E8041A10	16
	Timer general register A	TGRA	H'E8041A18	16
	Timer general register B	TGRB	H'E8041A1A	16
	Timer general register C	TGRC	H'E8041A24	16
	Timer general register D	TGRD	H'E8041A26	16
	Timer control register 2	TCR2	H'E8041A4C	8
	Timer general register E	TGRE	H'E8041A72	16
	Timer synchronous clear register	TSYCR	H'E8041A50	8
	Timer status register	TSR	H'E8041A2C	8
	Timer buffer operation transfer mode register	TBTM	H'E8041A38	8
	Noise filter control register 6	NFCR6	H'E8041A93	8
MTU7	Timer control register	TCR	H'E8041A01	8
	Timer mode register 1	TMDR1	H'E8041A03	8
	Timer I/O control register H	TIORH	H'E8041A06	8
	Timer I/O control register L	TIORL	H'E8041A07	8
	Timer interrupt enable register	TIER	H'E8041A09	8
	Timer counter	TCNT	H'E8041A12	16
	Timer general register A	TGRA	H'E8041A1C	16
	Timer general register B	TGRB	H'E8041A1E	16
	Timer general register C	TGRC	H'E8041A28	16
	Timer general register D	TGRD	H'E8041A2A	16
	Timer control register 2	TCR2	H'E8041A4D	8
	Timer general register E	TGRE	H'E8041A74	16
	Timer general register F	TGRF	H'E8041A76	16
	Timer status register	TSR	H'E8041A2D	8
	Timer buffer operation transfer mode register	TBTM	H'E8041A39	8
	Timer A/D converter start request control register	TADCR	H'E8041A40	16
	Timer A/D converter start request cycle set register A	TADCORA	H'E8041A44	16
	Timer A/D converter start request cycle set register B	TADCORB	H'E8041A46	16
	Timer A/D converter start request cycle set buffer register A	TADCOBRA	H'E8041A48	16
	Timer A/D converter start request cycle set buffer register B	TADCOBRB	H'E8041A4A	16
	Noise filter control register 7	NFCR7	H'E8041A94	8

Channel	Register Name	Abbreviation	Address	Access size
MTU8	Timer control register	TCR	H'E8041600	8
	Timer mode register 1	TMDR1	H'E8041601	8
	Timer I/O control register H	TIORH	H'E8041602	8
	Timer I/O control register L	TIORL	H'E8041603	8
	Timer interrupt enable register	TIER	H'E8041604	8
	Timer control register 2	TCR2	H'E8041606	8
	Timer counter	TCNT	H'E8041608	32
	Timer general register A	TGRA	H'E804160C	32
	Timer general register B	TGRB	H'E8041610	32
	Timer general register C	TGRC	H'E8041614	32
	Timer general register D	TGRD	H'E8041618	32
	Noise filter control register 8	NFCR8	H'E8041298	8
MTU	Timer output master enable register A	TOERA	H'E804120A	8
	Timer gate control register A	TGCRA	H'E804120D	8
	Timer output control register 1A	TOCR1A	H'E804120E	8
	Timer output control register 2A	TOCR2A	H'E804120F	8
	Timer cycle data register A	TCDRA	H'E8041214	16
	Timer dead time data register A	TDDRA	H'E8041216	16
	Timer subcounter A	TCNTSA	H'E8041220	16
	Timer cycle buffer register A	TCBRA	H'E8041222	16
	Timer interrupt skipping set register 1A	TITCR1A	H'E8041230	8
	Timer interrupt skipping counter 1A	TITCNT1A	H'E8041231	8
	Timer buffer transfer set register A	TBTERA	H'E8041232	8
	Timer dead time enable register A	TDERA	H'E8041234	8
	Timer output level buffer register A	TOLBRA	H'E8041236	8
	Timer interrupt skipping mode register A	TITMRA	H'E804123A	8
	Timer interrupt skipping set register 2A	TITCR2A	H'E804123B	8
	Timer interrupt skipping counter 2A	TITCNT2A	H'E804123C	8
	Timer waveform control register A	TWCRA	H'E8041260	8
	Timer mode register 2A	TMDR2A	H'E8041270	8
	Timer read/write enable register A	TRWERA	H'E8041284	8
	Timer start register A	TSTRA	H'E8041280	8
	Timer synchronous register A	TSYRA	H'E8041281	8
	Timer output master enable register B	TOERB	H'E8041A0A	8
	Timer output control register 1B	TOCR1B	H'E8041A0E	8
	Timer output control register 2B	TOCR2B	H'E8041A0F	8
	Timer cycle data register B	TCDRB	H'E8041A14	16
	Timer dead time data register B	TDDRB	H'E8041A16	16
	Timer subcounter B	TCNTSB	H'E8041A20	16
	Timer cycle buffer register B	TCBRB	H'E8041A22	16
	Timer interrupt skipping set register 1B	TITCR1B	H'E8041A30	8
	Timer interrupt skipping counter 1B	TITCNT1B	H'E8041A31	8
	Timer buffer transfer set register B	TBTERB	H'E8041A32	8
	Timer dead time enable register B	TDERB	H'E8041A34	8
	Timer output level buffer register B	TOLBRB	H'E8041A36	8
	Timer interrupt skipping mode register B	TITMRB	H'E8041A3A	8

Channel	Register Name	Abbreviation	Address	Access size
MTU	Timer interrupt skipping set register 2B	TITCR2B	H'E8041A3B	8
	Timer interrupt skipping counter 2B	TITCNT2B	H'E8041A3C	8
	Timer waveform control register B	TWCRB	H'E8041A60	8
	Timer mode register 2B	TMDR2B	H'E8041A70	8
	Timer start register B	TSTRB	H'E8041A80	8
	Timer synchronous register B	TSYRB	H'E8041A81	8
	Timer read/write enable register B	TRWERB	H'E8041A84	8
	Timer counter synchronous start register	TCSYSTR	H'E8041282	8

10.2.1 Timer Control Register (TCR)

- MTU0.TCR, MTU1.TCR, MTU2.TCR, MTU3.TCR, MTU4.TCR, MTU6.TCR, MTU7.TCR, MTU8.TCR

Address(es): MTU0.TCR H'E8041300, MTU1.TCR H'E8041380, MTU2.TCR H'E8041400, MTU3.TCR H'E8041200, MTU4.TCR H'E8041201, MTU6.TCR H'E8041A00, MTU7.TCR H'E8041A01, MTU8.TCR H'E8041600

	b7	b6	b5	b4	b3	b2	b1	b0
	CCLR[2:0]			CKEG[1:0]		TPSC[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TPSC[2:0]	Time Prescaler Select	See Table 10.7 to Table 10.10.	R/W
b4, b3	CKEG[1:0]	Clock Edge Select	b4 b3 0 0: Count at rising edge 0 1: Count at falling edge 1 x: Count at both edges	R/W
b7 to b5	CCLR[2:0]	Counter Clear Source Select	See Table 10.5 and Table 10.6.	R/W

x: Don't care

The TCR register controls the TCNT operation for each channel in combination with the TCR2 register. The MTU has a total of 11 TCR registers, one each for MTU0 to MTU4, MTU 6, MTU7, and MTU8 and three (TCRU, TCRV, and TCRW) for MTU5. TCR values should be specified only while TCNT operation is stopped.

TPSC[2:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. The clock source can be selected independently for each channel. See Table 10.7 to Table 10.10 for details.

CKEG[1:0] Bits (Clock Edge Select)

These bits select the count clock source edge, including the MTIOC1A pin. When the internal clock is counted at both edges, the count clock period is halved (e.g. P1φ/4 at both edges = P1φ/2 at rising edge). If phase counting mode is used on MTU1 and MTU2, the setting of these bits is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the count clock source is P1φ/2 or slower. When P1φ/1 or the overflow/underflow in another channel is selected for the count clock source, a value can be written to these bits but counter operation compiles with the initial value.

CCLR[2:0] Bits (Counter Clear Source Select)

These bits select the TCNT counter clearing source. See Table 10.5 and Table 10.6 for details.

Table 10.5 CCLR[2:0] (MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8)

Channel	Bit 7 CCLR2	Bit 6 CCLR1	Bit 5 CCLR0	Description
MTU0 MTU3 MTU4 MTU6 MTU7 MTU8	0	0	0	TCNT clearing disabled
	0	0	1	TCNT cleared by TGRA compare match/input capture
	0	1	0	TCNT cleared by TGRB compare match/input capture
	0	1	1	TCNT cleared by counter clearing in another channel where synchronous clearing or synchronous operation is selected*1
	1	0	0	TCNT clearing disabled
	1	0	1	TCNT cleared by TGRC compare match/input capture*2
	1	1	0	TCNT cleared by TGRD compare match/input capture*2
	1	1	1	TCNT cleared by counter clearing in another channel where synchronous clearing or synchronous operation is selected*1

Note 1. Synchronous operation is selected by setting a SYNC bit in TSYRA or TSYRB to 1 except for MTU8.
Note 2. When TGRC or TGRD is used as a buffer register, TCNT is not cleared because the buffer register setting has priority and compare match/input capture does not occur.

Table 10.6 CCLR[2:0] (MTU1 and MTU2)

Channel	Bit 7 Reserved*2	Bit 6 CCLR1	Bit 5 CCLR0	Description
MTU1 MTU2	0	0	0	TCNT clearing disabled
	0	0	1	TCNT cleared by TGRA compare match/input capture
	0	1	0	TCNT cleared by TGRB compare match/input capture
	0	1	1	TCNT cleared by counter clearing in another channel where synchronous clearing or synchronous operation is selected*1

Note 1. Synchronous operation is selected by setting a SYNC bit in TSYRA or TSYRB to 1.
Note 2. Bit 7 is reserved in MTU1 and MTU2. It is read as 0. The write value is ignored.

- MTU5.TCRU, MTU5.TCRV, MTU5.TCRW

Address(es): MTU5.TCRU H'E8041C84, MTU5.TCRV H'E8041C94, MTU5.TCRW H'E8041CA4

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	TPSC[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	TPSC[1:0]	Time Prescaler Select	See Table 10.11.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TPSC[1:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. See Table 10.11 for details.

10.2.2 Timer Control Register 2 (TCR2)

- MTU0.TCR2, MTU3.TCR2, MTU4.TCR2, MTU6.TCR2, MTU7.TCR2, MTU8.TCR2

Address(es): MTU0.TCR2 H'E8041328, MTU3.TCR2 H'E804124C, MTU4.TCR2 H'E804124D, MTU6.TCR2 H'E8041A4C, MTU7.TCR2 H'E8041A4D, MTU8.TCR2 H'E8041606

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TPSC2[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

- MTU1.TCR2, MTU2.TCR2

Address(es): MTU1.TCR2 H'E8041394, MTU2.TCR2 H'E804140C

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PCB[1:0]		TPSC2[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TPSC2[2:0]	Time Prescaler Select	See Table 10.7 to Table 10.10.	R/W
b4, b3	PCB[1:0]	Phase Counting Mode Function Expansion Control	Functional Expansion Control for Phase Counting Modes 2, 3, and 5	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The TCR2 register controls the TCNT operation for each channel in combination with the TCR register. The MTU has a total of 11 TCR2 registers, one each for MTU0 to MTU4, MTU 6, MTU7, and MTU8 and three (TCR2U, TCR2V, and TCR2W) for MTU5. TCR2 values should be specified only while TCNT operation is stopped.

TPSC2[2:0] Bits (Time Prescaler Select)

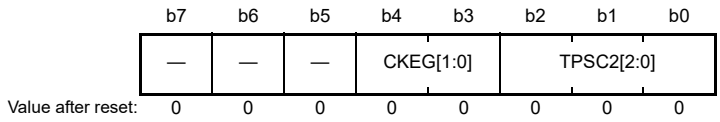
These bits select the TCNT count clock source. The clock source can be selected independently for each channel. See Table 10.7 to Table 10.10 for details.

PCB[1:0] Bits (Phase Counting Mode Function Expansion Control)

These bits control extended functions for phase counting mode 2, 3, and 5 in MTU1 and MTU2. See section 10.3.6, Phase Counting Mode.

- MTU5.TCR2U, MTU5.TCR2V, MTU5.TCR2W

Address(es): MTU5.TCR2U H'E8041C85, MTU5.TCR2V H'E8041C95, MTU5.TCR2W H'E8041CA5



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TPSC2[2:0]	Time Prescaler Select	See Table 10.11.	R/W
b4, b3	CKEG[1:0]	Clock Edge Select	b4 b3 0 0: Counts at the rising edge. 0 1: Counts at the falling edge. 1 x: Counts at both edges.	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

x: Don't care

TPSC2[2:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. See Table 10.11 for details.

CKEG[1:0] Bits (Clock Edge Select)

These bits select the edge of the count clock source signal output from the MTIOC1A pin.

Table 10.7 TPSC[2:0], TPSC2[2:0] (MTU0)

	TCR2[2:0]			TCR[2:0]			
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC2	TPSC1	TPSC0	Description
MTU0	0	0	0	0	0	0	Internal clock: counts on P1φ/1
	0	0	0	0	0	1	Internal clock: counts on P1φ/4
	0	0	0	0	1	0	Internal clock: counts on P1φ/16
	0	0	0	0	1	1	Internal clock: counts on P1φ/64
	0	0	0	1	0	0	External clock: counts on MTCLKA pin input
	0	0	0	1	0	1	External clock: counts on MTCLKB pin input
	0	0	0	1	1	0	External clock: counts on MTCLKC pin input
	0	0	0	1	1	1	External clock: counts on MTCLKD pin input
	0	0	1	x	x	x	Internal clock: counts on P1φ/2
	0	1	0	x	x	x	Internal clock: counts on P1φ/8
	0	1	1	x	x	x	Internal clock: counts on P1φ/32
	1	0	0	x	x	x	Internal clock: counts on P1φ/256
	1	0	1	x	x	x	Internal clock: counts on P1φ/1024
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	External clock: counts on MTIOC1A pin input

x: Don't care

Table 10.8 TPSC[2:0], TPSC2[2:0] (MTU1)

	TCR2[2:0]			TCR[2:0]			
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC2	TPSC1	TPSC0	Description
MTU1	0	0	0	0	0	0	Internal clock: counts on P1φ/1
	0	0	0	0	0	1	Internal clock: counts on P1φ/4
	0	0	0	0	1	0	Internal clock: counts on P1φ/16
	0	0	0	0	1	1	Internal clock: counts on P1φ/64
	0	0	0	1	0	0	External clock: counts on MTCLKA pin input
	0	0	0	1	0	1	External clock: counts on MTCLKB pin input
	0	0	0	1	1	0	Internal clock: counts on P1φ/256
	0	0	0	1	1	1	Overflow/underflow of MTU2.TCNT
	0	0	1	x	x	x	Internal clock: counts on P1φ/2
	0	1	0	x	x	x	Internal clock: counts on P1φ/8
	0	1	1	x	x	x	Internal clock: counts on P1φ/32
	1	0	0	x	x	x	Internal clock: counts on P1φ/1024
	1	0	1	x	x	x	Setting prohibited
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	Setting prohibited

x: Don't care

Note: This setting has no effect when MTU1 is in phase counting mode.

Table 10.9 TPSC[2:0], TPSC2[2:0] (MTU2)

	TCR2[2:0]			TCR[2:0]			
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC2	TPSC1	TPSC0	Description
MTU2	0	0	0	0	0	0	Internal clock: counts on P1φ/1
	0	0	0	0	0	1	Internal clock: counts on P1φ/4
	0	0	0	0	1	0	Internal clock: counts on P1φ/16
	0	0	0	0	1	1	Internal clock: counts on P1φ/64
	0	0	0	1	0	0	External clock: counts on MTCLKA pin input
	0	0	0	1	0	1	External clock: counts on MTCLKB pin input
	0	0	0	1	1	0	External clock: counts on MTCLKC pin input
	0	0	0	1	1	1	Internal clock: counts on P1φ/1024
	0	0	1	x	x	x	Internal clock: counts on P1φ/2
	0	1	0	x	x	x	Internal clock: counts on P1φ/8
	0	1	1	x	x	x	Internal clock: counts on P1φ/32
	1	0	0	x	x	x	Internal clock: counts on P1φ/256
	1	0	1	x	x	x	Setting prohibited
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	Setting prohibited

x: Don't care

Note: This setting has no effect when the MTU2 is in phase counting mode.

Table 10.10 TPSC[2:0], TPSC2[2:0] (MTU3, MTU4, MTU6, MTU7, and MTU8)

	TCR2[2:0]			TCR[2:0]			
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC2	TPSC1	TPSC0	Description
MTU3 MTU4 MTU6 MTU7 MTU8	0	0	0	0	0	0	Internal clock: counts on P1φ/1
	0	0	0	0	0	1	Internal clock: counts on P1φ/4
	0	0	0	0	1	0	Internal clock: counts on P1φ/16
	0	0	0	0	1	1	Internal clock: counts on P1φ/64
	0	0	0	1	0	0	Internal clock: counts on P1φ/256
	0	0	0	1	0	1	Internal clock: counts on P1φ/1024
	0	0	0	1	1	0	External clock: counts on MTCLKA pin input
	0	0	0	1	1	1	External clock: counts on MTCLKB pin input
	0	0	1	x	x	x	Internal clock: counts on P1φ/2
	0	1	0	x	x	x	Internal clock: counts on P1φ/8
	0	1	1	x	x	x	Internal clock: counts on P1φ/32
	1	0	0	x	x	x	Setting prohibited
	1	0	1	x	x	x	Setting prohibited
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	Setting prohibited

x: Don't care

Table 10.11 TPSC[1:0], TPSC2[2:0] (MTU5)

	TCR2[2:0]			TCR[1:0]		
	Bit 2	Bit 1	Bit 0	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC1	TPSC0	Description
MTU5	0	0	0	0	0	Internal clock: counts on P1φ/1
	0	0	0	0	1	Internal clock: counts on P1φ/4
	0	0	0	1	0	Internal clock: counts on P1φ/16
	0	0	0	1	1	Internal clock: counts on P1φ/64
	0	0	1	x	x	Internal clock: counts on P1φ/2
	0	1	0	x	x	Internal clock: counts on P1φ/8
	0	1	1	x	x	Internal clock: counts on P1φ/32
	1	0	0	x	x	Internal clock: counts on P1φ/256
	1	0	1	x	x	Internal clock: counts on P1φ/1024
	1	1	0	x	x	Setting prohibited
	1	1	1	x	x	Internal clock: counts on MTIOC1A pin input

x: Don't care

10.2.3 Timer Mode Register 1 (TMDR1)

- MTU0.TMDR1

Address(es): MTU0.TMDR1 H'E8041301

	b7	b6	b5	b4	b3	b2	b1	b0
	—	BFE	BFB	BFA	MD[3:0]			
Value after reset:	0	0	0	0	0	0	0	0

- MTU1.TMDR1, MTU2.TMDR1

Address(es): MTU1.TMDR1 H'E8041381, MTU2.TMDR1 H'E8041401

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	MD[3:0]			
Value after reset:	0	0	0	0	0	0	0	0

- MTU3.TMDR1, MTU4.TMDR1, MTU6.TMDR1, MTU7.TMDR1, MTU8.TMDR1

Address(es): MTU3.TMDR1 H'E8041202, MTU4.TMDR1 H'E8041203, MTU6.TMDR1 H'E8041A02, MTU7.TMDR1 H'E8041A03, MTU8.TMDR1 H'E8041601

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	BFB	BFA	MD[3:0]			
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	MD[3:0]	Mode Select	These bits specify the timer operating mode. See Table 10.12 for details.	R/W
b4	BFA	Buffer Operation A	0: TGRA and TGRC operate normally 1: TGRA and TGRC used together for buffer operation	R/W
b5	BFB	Buffer Operation B	0: TGRB and TGRD operate normally 1: TGRB and TGRD used together for buffer operation	R/W
b6	BFE	Buffer Operation E	0: MTU0.TGRE and MTU0.TGRF operate normally 1: MTU0.TGRE and MTU0.TGRF used together for buffer operation	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

The TMDR1 register specifies the operating mode of each channel. The MTU has a total of eight TMDR1 registers, one each for MTU0 to MTU4, MTU 6, MTU7, and MTU8. TMDR1 register values should be specified only while TCNT operation is stopped.

Table 10.12 Operating Mode Setting by MD[3:0] Bits (MTU0 to MTU4, MTU6 to MTU8)

Bit 3 MD3	Bit 2 MD2	Bit 1 MD1	Bit 0 MD0	Description	MTU0	MTU1	MTU2	MTU1 & MTU2 (LWA = 1)	MTU3	MTU4	MTU6	MTU7	MTU8
0	0	0	0	Normal mode	√	√	√		√	√	√	√	√
0	0	0	1	Setting prohibited									
0	0	1	0	PWM mode 1	√	√	√		√	√	√	√	
0	0	1	1	PWM mode 2	√	√	√						
0	1	0	0	Phase counting mode 1		√	√	√					
0	1	0	1	Phase counting mode 2		√	√	√					
0	1	1	0	Phase counting mode 3		√	√	√					
0	1	1	1	Phase counting mode 4		√	√	√					
1	0	0	0	Reset-synchronized PWM mode*1					√		√		
1	0	0	1	Phase counting mode 5		√	√	√					
1	0	1	x	Setting prohibited									
1	1	0	0	Setting prohibited									
1	1	0	1	Complementary PWM mode 1 (transfer at crest)*1					√		√		
1	1	1	0	Complementary PWM mode 2 (transfer at trough)*1					√		√		
1	1	1	1	Complementary PWM mode 3 (transfer at crest and trough)*1					√		√		

x: Don't care

Note: Only set the corresponding operating mode listed above for each channel.

Note 1. Reset-synchronized PWM mode and complementary PWM mode can only be set for MTU3 and MTU6. When MTU3 or MTU6 is set to reset-synchronized PWM mode or complementary PWM mode, the MTU4 or MTU7 settings become ineffective and automatically conform to the MTU3 or MTU6 setting, respectively. MTU4 and MTU7 should be set to the initial values (normal mode).

BFA Bit (Buffer Operation A)

This bit specifies whether to operate TGRA in the normal way or to use TGRA and TGRC together for buffer operation. When TGRC is used as a buffer register, TGRC input capture/output compare does not take place in modes other than complementary PWM mode, but compare match with TGRC occurs in complementary PWM mode.

If a compare match occurs on MTU4 in the Tb interval in complementary PWM mode, the TGIEC bit in timer interrupt enable register (MTU4.TIER) should be set to 0.

In reset-synchronized PWM mode or complementary PWM mode, buffer operation in MTU3 and MTU4 (or MTU6 and MTU7) depends on the settings in the BFA bit of MTU3.TMDR1 (MTU6.TMDR1). The BFA bit of MTU4.TMDR1 (MTU7.TMDR1) should be set to 0.

In MTU1 and MTU2, which have no TGRC, this bit is reserved. It is read as 0. The write value should be 0.

See Figure 10.50 for an illustration of the Tb interval in complementary PWM mode.

BFB Bit (Buffer Operation B)

This bit specifies whether to operate TGRB in the normal way or to use TGRB and TGRD together for buffer operation. When TGRD is used as a buffer register, TGRD input capture/output compare does not take place in modes other than complementary PWM mode, but compare match with TGRD occurs in complementary PWM mode.

If a compare match occurs on MTU4 in the Tb interval in complementary PWM mode, the TGIED bit in timer interrupt enable register (MTU4.TIER) should be set to 0.

In reset-synchronized PWM mode or complementary PWM mode, buffer operation in MTU3 and MTU4 (or MTU6 and MTU7) depends on the settings in the BFB bit of MTU3.TMDR1 (MTU6.TMDR1). The BFB bit of MTU4.TMDR1 (MTU7.TMDR1) should be set to 0.

In MTU1 and MTU2, which have no TGRD, this bit is reserved. It is read as 0. The write value should be 0.

See Figure 10.50 for an illustration of the Tb interval in complementary PWM mode.

BFE Bit (Buffer Operation E)

This bit specifies whether to operate MTU0.TGRE and MTU0.TGRF in the normal way or to use them together for buffer operation. Compare match with TGRF occurs even when TGRF is used as a buffer register.
In MTU0 to MTU4, MTU6, MTU7, and MTU8, this bit is reserved. It is read as 0. The write value should be 0.

10.2.4 Timer Mode Registers 2 (TMDR2A and TMDR2B)

Address(es): MTU.TMDR2A H'E8041270, MTU.TMDR2B H'E8041A70

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	DRS
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DRS	Double Buffer Select	0: Double buffer function is disabled 1: Double buffer function is enabled	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TMDR2A and TMDR2B specify the double buffer function in complementary PWM mode 3 (transfer at the crest and trough of the counter value). The MTU has two TMDR2 registers, one each for MTU3 (TMDR2A) and MTU6 (TMDR2B). TMDR2A and TMDR2B values should be specified only while TCNT operation is stopped.

DRS Bit (Double Buffer Select)

This bit enables or disables the double buffer function in complementary PWM mode.

10.2.5 Timer Mode Register 3 (TMDR3)

Address(es): MTU1.TMDR3 H'E8041391

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	PHCK SEL	LWA
Value after reset:	0	0	0	0	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	LWA	MTU1/MTU2 Combination Longword Access Control	0: 16-bit access is enabled. 1: 32-bit access is enabled.	R/W
b1	PHCKSEL	External Input Phase Clock Select	Selects the external clock pin for phase counting mode. 0: MTCLKA and MTCLKB are selected for the external phase clock. 1: MTCLKC and MTCLKD are selected for the external phase clock.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The TMDR3 register controls longword access to a 32-bit register or counter in a combination of MTU1 and MTU2. There is only one TMDR3 register in MTU1. The counter (TCNTLW), general register A (TGRALW), and general register B (TGRBLW) of MTU1 and MTU2 are accessed in the combinations listed in Table 10.13.

LWA Bit (MTU1/MTU2 Combination Longword Access Control)

This bit selects a 32-bit access in a combination of MTU1 and MTU2.

When the LWA bit is set to 0, MTU1 and MTU2 operate independently as 16-bit timers and the TCNTLW, TGRALW, and TGRBLW registers cannot be accessed.

When the LWA bit is set to 1, MTU1 and MTU2 are cascaded and operate as a 32-bit timer, which is controlled through the TCR, TCR2, TIOR, and TMDR1 registers in MTU1. The TCR, TCR2, TIOR, and TMDR1 register settings in MTU2 are ignored. The 16-bit registers (TCNT, TGRA, and TGRB) in MTU1 and MTU2 cannot be accessed. MTU2 input capture and compare match are disabled.

Note that MTU1 and MTU2 can be cascaded by setting the LWA bit to 1 only in phase-counting mode. Cascade connection cannot be used in normal, PWM1, or PWM2 mode. When setting the LWA bit to 1, be sure to select phase-counting mode.

In addition, initialize the TCNT, TGRA, and TGRB registers in MTU1 and MTU2 before setting the LWA bit to 1.

PHCKSEL Bit (External Input Phase Clock Select)

When the MTU1 and MTU2 registers are combined for 32-bit phase counting mode or MTU2 phase counting mode, this bit selects either the A- or B-phase signal from the external clock. See Table 10.66, Clock Input Pins in Phase Counting Mode for details.

Table 10.13 Setting and Combination of the TMDR3 Register

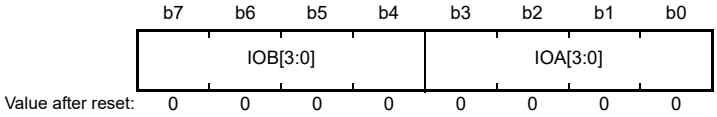
	TMDR3.LWA = 0		TMDR3.LWA = 1	
Register	Symbol	Access mode	Symbol	Access mode
Counter in MTU1*1	MTU1.TCNT	Word	MTU1.TCNT_1_LW	Longword
Counter in MTU2	MTU2.TCNT	Word		
General register A in MTU1	MTU1.TGRA	Word	MTU1.TGRA_1_LW	Longword
General register A in MTU2	MTU2.TGRA	Word		
General register B in MTU1	MTU1.TGRB	Word	MTU1.TGRB_1_LW	Longword
General register B in MTU2	MTU2.TGRB	Word		

Note 1. When the LWA bit is set to 1, setting the count clock for MTU1 as MTU2.TCNT overflow/underflow is not required.

10.2.6 Timer I/O Control Register (TIOR)

- MTU0.TIORH, MTU1.TIOR, MTU2.TIOR, MTU3.TIORH, MTU4.TIORH, MTU6.TIORH, MTU7.TIORH, MTU8.TIORH

Address(es): MTU0.TIORH H'E8041302, MTU1.TIOR H'E8041382, MTU2.TIOR H'E8041402, MTU3.TIORH H'E8041204, MTU4.TIORH H'E8041206, MTU6.TIORH H'E8041A04, MTU7.TIORH H'E8041A06, MTU8.TIORH H'E8041602



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	IOA[3:0]	I/O Control A*1	See the following tables. MTU0.TIORH: Table 10.28 MTU1.TIOR: Table 10.30 MTU2.TIOR: Table 10.31 MTU3.TIORH: Table 10.32 MTU4.TIORH: Table 10.34 MTU6.TIORH: Table 10.36 MTU7.TIORH: Table 10.38 MTU8.TIORH: Table 10.40	R/W
b7 to b4	IOB[3:0]	I/O Control B*1	See the following tables. MTU0.TIORH: Table 10.14 MTU1.TIOR: Table 10.16 MTU2.TIOR: Table 10.17 MTU3.TIORH: Table 10.18 MTU4.TIORH: Table 10.20 MTU6.TIORH: Table 10.22 MTU7.TIORH: Table 10.24 MTU8.TIORH: Table 10.26	R/W

Note 1. When the value of IOn[3:0] (n = A, B) is changed to the output-prohibited state (0000b or 0100b) during low, high, or toggle output at compare match, this register is in Hi-Z.

- MTU0.TIORL, MTU3.TIORL, MTU4.TIORL, MTU6.TIORL, MTU7.TIORL, MTU8.TIORL

Address(es): MTU0.TIORL H'E8041303, MTU3.TIORL H'E8041205, MTU4.TIORL H'E8041207, MTU6.TIORL H'E8041A05, MTU7.TIORL H'E8041A07, MTU8.TIORL H'E8041603

	b7	b6	b5	b4	b3	b2	b1	b0
	IOD[3:0]				IOC[3:0]			
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	IOC[3:0]	I/O Control C*1	See the following tables. MTU0.TIORL: Table 10.29 MTU3.TIORL: Table 10.33 MTU4.TIORL: Table 10.35 MTU6.TIORL: Table 10.37 MTU7.TIORL: Table 10.39 MTU8.TIORL: Table 10.41	R/W
b7 to b4	IOD[3:0]	I/O Control D*1	See the following tables. MTU0.TIORL: Table 10.15 MTU3.TIORL: Table 10.19 MTU4.TIORL: Table 10.21 MTU6.TIORL: Table 10.23 MTU7.TIORL: Table 10.25 MTU8.TIORL: Table 10.27	R/W

Note 1. When the value of IOn[3:0] (n = C, D) is changed to the output-prohibited state (0000b or 0100b) during low, high, or toggle output at compare match, this register is in Hi-Z.

- MTU5.TIORU, MTU5.TIORV, MTU5.TIORW

Address(es): MTU5.TIORU H'E8041C86, MTU5.TIORV H'E8041C96, MTU5.TIORW H'E8041CA6

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	IOC[4:0]				
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	IOC[4:0]	I/O Control C	See the following table. MTU5.TIORU, MTU5.TIORV, MTU5.TIORW: Table 10.42	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The TIOR register controls the TGR register. The MTU has a total of 17 TIOR registers, two each for MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8, one each for MTU1 and MTU2, and three (MTU5.TIORU/TIORV/TIORW) for MTU5. The TIOR register should be set when the TMDR register setting is normal mode, PWM mode, or phase counting mode.

Note that TIOR is affected by the TMDR1 setting.

The initial output specified by TIOR is valid when the counter is stopped (the CSTn bits in TSTRA and TSTRB are cleared to 0). Note also that, in PWM mode 2, the output at the point at which the counter is cleared to 0 is specified.

When TGRC or TGRD is designated for buffer operation, this setting is invalid and the register operates as a buffer register.

Table 10.14 TIORH (MTU0)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU0.TGRB Function	MTIOC0B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by TCNT (LWA = 0) or TCNTLW (LWA = 1) in MTU1*1

x: Don't care

Note 1. Input capture will not be generated in MTU0 if P1φ/1 is selected as the count clock for MTU1. Select a clock other than P1φ/1.

Table 10.15 TIORL (MTU0)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU0.TGRD Function	MTIOC0D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register*1	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by TCNT (LWA = 0) or TCNTLW (LWA = 1) in MTU1*2

x: Don't care

Note 1. When the MTU0.TMDR1.BFB is set to 1 and MTU0.TGRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Note 2. Input capture will not be generated in MTU0 if P1φ/1 is selected as the count clock for MTU1. Select a clock other than P1φ/1.

Table 10.16 TIOR (MTU1)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU1.TGRB/TGRBLW Function	MTIOC1B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	0	0		Input capture at occurrence of compare match or input capture in the MTU0.TGRC register
1	1	1	x		Input capture at occurrence of compare match in the MTU8.TGRC register

x: Don't care

Table 10.17 TIOR (MTU2)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU2.TGRB Function	MTIOC2B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 10.18 TIORH (MTU3)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU3.TGRB Function	MTIOC3B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 10.19 TIORL (MTU3)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU3.TGRD Function	MTIOC3D Pin Function
0	0	0	0	Output compare register *1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the BFB bit in MTU3.TMDR1 is set to 1 and MTU3.TGRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 10.20 TIORH (MTU4)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU4.TGRB Function	MTIOC4B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 10.21 TIORL (MTU4)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU4.TGRD Function	MTIOC4D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the BFB bit in MTU4.TMDR1 is set to 1 and MTU4.TGRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 10.22 TIORH (MTU6)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU6.TGRB Function	MTIOC6B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 10.23 TIORL (MTU6)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU6.TGRD Function	MTIOC6D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the BFB bit in MTU6.TMDR1 is set to 1 and MTU6.TGRD register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 10.24 TIORH (MTU7)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU7.TGRB Function	MTIOC7B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 10.25 TIORL (MTU7)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU7.TGRD Function	MTIOC7D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the BFB bit in MTU7.TMDR1 is set to 1 and MTU7.TGRD register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 10.26 TIORH (MTU8)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU8.TGRB Function	MTIOC8B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by MTU1.TCNT (LWA = 0) or MTU1.TCNTLW (LWA = 1)*1

x: Don't care

Note 1. Input capture will not be generated in MTU8 if P1φ/1 is selected as the count clock for MTU1. Select a clock other than P1φ/1.

Table 10.27 TIORL (MTU8)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU8.TGRD Function	MTIOC8D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the BFB bit of the TMDR1 register is set to 1 and the TGRD register is used as a buffer register in MTU8, this setting is invalid and input capture/output compare is not generated.

Table 10.28 TIORH (MTU0)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU0.TGRA Function	MTIOC0A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	0	0		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by TCNT (LWA = 0) or TCNTLW (LWA = 1) in MTU1*1
1	1	1	x		Input capture on generation of compare match with MTU8.TGRC

x: Don't care

Note 1. Input capture will not be generated in MTU0 if P1φ/1 is selected as the count clock for MTU1. Select a clock other than P1φ/1.

Table 10.29 TIORL (MTU0)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU0.TGRC Function	MTIOC0C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register*1	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by TCNT (LWA = 0) or TCNTLW (LWA = 1) in MTU1*2

x: Don't care

Note 1. When the BFA bit in MTU0.TMDR1 is set to 1 and MTU0.TGRC is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Note 2. Input capture will not be generated in MTU0 if P1φ/1 is selected as the count clock for MTU1. Select a clock other than P1φ/1.

Table 10.30 TIOR (MTU1)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU1.TGRA/TGRALW Function	MTIOC1A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Input capture at generation of MTU0.TGRA compare match/input capture.

x: Don't care

Table 10.31 TIOR (MTU2)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU2.TGRA Function	MTIOC2A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 10.32 TIORH (MTU3)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU3.TGRA Function	MTIOC3A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 10.33 TIORL (MTU3)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU3.TGRC Function	MTIOC3C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the BFA bit in MTU3.TMDR1 is set to 1 and MTU3.TGRC is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 10.34 TIORH (MTU4)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU4.TGRA Function	MTIOC4A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 10.35 TIORL (MTU4)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU4.TGRC Function	MTIOC4C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the BFA bit in MTU4.TMDR1 is set to 1 and MTU4.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 10.36 TIORH (MTU6)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU6.TGRA Function	MTIOC6A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 10.37 TIORL (MTU6)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU6.TGRC Function	MTIOC6C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the BFA bit in MTU6.TMDR1 is set to 1 and MTU6.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 10.38 TIORH (MTU7)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU7.TGRA Function	MTIOC7A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 10.39 TIORL (MTU7)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU7.TGRC Function	MTIOC7C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the BFA bit in MTU7.TMDR1 is set to 1 and MTU7.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 10.40 TIORH (MTU8)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU8.TGRA Function	MTIOC8A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 10.41 TIORL (MTU8)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU8.TGRC Function	MTIOC8C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the MTU8.TMDR1.BFA bit is set to 1 and the MTU8.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 10.42 TIORU, TIORV, and TIORW (MTU5)

Description						
Bit 4 IOC4	Bit 3 IOC3	Bit 2 IOC2	Bit 1 IOC1	Bit 0 IOC0	MTU5.TGRU, MTU5.TGRV, MTU5.TGRW Function	MTIC5U, MTIC5V, MTIC5W Pin Function
0	0	0	0	0	Output compare register	No function
0	0	0	0	1		Setting prohibited
0	0	0	1	x		Setting prohibited
0	0	1	x	x		Setting prohibited
0	1	x	x	x		Setting prohibited
1	0	0	0	0	Input capture register*1	Setting prohibited
1	0	0	0	1		Input capture at rising edge.
1	0	0	1	0		Input capture at falling edge.
1	0	0	1	1		Input capture at both edges.
1	0	1	x	x		Input capture on generation of compare match with MTU8.TGRC
1	1	0	0	0		Setting prohibited
1	1	0	0	1		Measurement of low pulse width of external input signal. Capture at trough in complementary PWM mode.
1	1	0	1	0		Measurement of low pulse width of external input signal. Capture at crest of complementary PWM mode.
1	1	0	1	1		Measurement of low pulse width of external input signal. Capture at crest and trough of complementary PWM mode.
1	1	1	0	0		Setting prohibited
1	1	1	0	1		Measurement of high pulse width of external input signal. Capture at trough in complementary PWM mode.
1	1	1	1	0		Measurement of high pulse width of external input signal. Capture at crest of complementary PWM mode.
1	1	1	1	1		Measurement of high pulse width of external input signal. Capture at crest and trough of complementary PWM mode.

x: Don't care

Note 1. Setting the IOC[4:0] bits to "19h", "1Ah", "1Bh", "1Dh", "1Eh", or "1Fh" is only allowed when the external pulse width measurement function is used or the dead time compensation function, in coordination with MTU6 and MTU7, is used. For details, refer to section 10.3.11, External Pulse Width Measurement, and section 10.3.12, Dead Time Compensation.

10.2.7 Timer Compare Match Clear Register (TCNTCMPCLR)

Address(es): MTU5.TCNTCMPCLR H'E8041CB6

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CMPCLR5U	CMPCLR5V	CMPCLR5W
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CMPCLR5W	TCNT Compare Clear 5W	0: Disables MTU5.TCNTW to be cleared to 0000h at MTU5.TCNTW and MTU5.TGRW compare match or input capture 1: Enables MTU5.TCNTW to be cleared to 0000h at MTU5.TCNTW and MTU5.TGRW compare match or input capture	R/W
b1	CMPCLR5V	TCNT Compare Clear 5V	0: Disables MTU5.TCNTV to be cleared to 0000h at MTU5.TCNTV and MTU5.TGRV compare match or input capture 1: Enables MTU5.TCNTV to be cleared to 0000h at MTU5.TCNTV and MTU5.TGRV compare match or input capture	R/W
b2	CMPCLR5U	TCNT Compare Clear 5U	0: Disables MTU5.TCNTU to be cleared to 0000h at MTU5.TCNTU and MTU5.TGRU compare match or input capture 1: Enables MTU5.TCNTU to be cleared to 0000h at MTU5.TCNTU and MTU5.TGRU compare match or input capture	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TCNTCMPCLR specifies requests to clear MTU5.TCNTU, MTU5.TCNTV, and MTU5.TCNTW. The MTU has one TCNTCMPCLR (on MTU5).

10.2.8 Timer Interrupt Enable Register (TIER)

- MTU1.TIER, MTU2.TIER

Address(es): MTU1.TIER H'E8041384, MTU2.TIER H'E8041404

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA
Value after reset:	0	0	0	0	0	0	0	0

- MTU0.TIER, MTU3.TIER, MTU6.TIER

Address(es): MTU0.TIER H'E8041304, MTU3.TIER H'E8041208, MTU6.TIER H'E8041A08

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Value after reset:	0	0	0	0	0	0	0	0

- MTU4.TIER, MTU7.TIER

Address(es): MTU4.TIER H'E8041209, MTU7.TIER H'E8041A09

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	TTGE2	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Value after reset:	0	0	0	0	0	0	0	0

- MTU8.TIER

Address(es): MTU8.TIER H'E8041604

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TGIEA	TGR Interrupt Enable A	0: Interrupt requests (TGIA) disabled 1: Interrupt requests (TGIA) enabled	R/W
b1	TGIEB	TGR Interrupt Enable B	0: Interrupt requests (TGIB) disabled 1: Interrupt requests (TGIB) enabled	R/W
b2	TGIEC	TGR Interrupt Enable C	0: Interrupt requests (TGIC) disabled 1: Interrupt requests (TGIC) enabled	R/W
b3	TGIED	TGR Interrupt Enable D	0: Interrupt requests (TGID) disabled 1: Interrupt requests (TGID) enabled	R/W
b4	TCIEV	Overflow Interrupt Enable	0: Interrupt requests (TCIV) disabled 1: Interrupt requests (TCIV) enabled	R/W
b5	TCIEU	Underflow Interrupt Enable	0: Interrupt requests (TCIU) disabled 1: Interrupt requests (TCIU) enabled	R/W
b6	TTGE2	A/D Converter Start Request Enable 2	0: A/D converter start request generation by MTUn.TCNT underflow (trough) disabled 1: A/D converter start request generation by MTUn.TCNT underflow (trough) enabled	R/W
b7	TTGE	A/D Converter Start Request Enable	0: A/D converter start request generation disabled 1: A/D converter start request generation enabled	R/W

n = 4, 7

The TIER register enables or disables interrupt requests from each channel. The MTU has a total of ten TIER registers, two for MTU0 and one each for MTU1 to MTU8.

TGIEA and TGIEB Bits (TGR Interrupt Enable A and B)

Each bit enables or disables interrupt requests (TGIn) (n = A, B).

TGIEC and TGIED Bits (TGR Interrupt Enable C and D)

Each bit enables or disables an interrupt request (TGIn) (n = C, D).
In MTU1 and MTU2, these bits are reserved. They are read as 0. The write value should be 0.

TCIEV Bit (Overflow Interrupt Enable)

This bit enables or disables interrupt requests (TCIV).

TCIEU Bit (Underflow Interrupt Enable)

This bit enables or disables interrupt requests (TCIU).
In MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8, this bit is reserved. It is read as 0. The write value should be 0.

TTGE2 Bit (A/D Converter Start Request Enable 2)

This bit enables or disables generation of A/D converter start requests by MTUn.TCNT underflow (trough) in complementary PWM mode (n = 4, 7).
In MTU0 to MTU3, MTU6, and MTU8, this bit is reserved. It is read as 0. The write value should be 0.

TTGE Bit (A/D Converter Start Request Enable)

This bit enables or disables generation of A/D converter start requests by TGRA input capture/compare match.
MTU8 is a reserved bit. It is read as 0. The write value should be 0.

- MTU0.TIER2

Address(es): MTU0.TIER2 H'E8041324

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE2	—	—	—	—	—	TGIEF	TGIEE
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TGIEE	TGR Interrupt Enable E	0: Interrupt requests (TGIE) disabled 1: Interrupt requests (TGIE) enabled	R/W
b1	TGIEF	TGR Interrupt Enable F	0: Interrupt requests (TGIF) disabled 1: Interrupt requests (TGIF) enabled	R/W
b6 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	TTGE2	A/D Converter Start Request Enable 2	0: A/D converter start request generation by compare match between MTU0.TCNT and MTU0.TGRE disabled 1: A/D converter start request generation by compare match between MTU0.TCNT and MTU0.TGRE enabled	R/W

TGIEE and TGIEF Bits (TGR Interrupt Enable E and F)

Each bit enables or disables interrupt requests by compare match between MTU0.TCNT and MTU0.TGRn (n = E, F).

TTGE2 Bit (A/D Converter Start Request Enable 2)

This bit enables or disables AD converter start requests by compare match between TCNT and TGRE in MTU0.

- MTU5.TIER

Address(es): MTU5.TIER H'E8041CB2

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TGIE5U	TGIE5V	TGIE5W
Value after reset:	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	TGIE5W	TGR Interrupt Enable 5W	0: Interrupt requests TGIW5 disabled 1: Interrupt requests TGIW5 enabled	R/W
b1	TGIE5V	TGR Interrupt Enable 5V	0: Interrupt requests TGIV5 disabled 1: Interrupt requests TGIV5 enabled	R/W
b2	TGIE5U	TGR Interrupt Enable 5U	0: Interrupt requests TGIU5 disabled 1: Interrupt requests TGIU5 enabled	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TGIE5n Bits (TGR Interrupt Enable 5n)

Each bit enables or disables interrupt requests (TGIn5) (n = U, V, W).

10.2.9 Timer Status Register (TSR)

- MTU1.TSR, MTU2.TSR

Address(es): MTU1.TSR H'E8041385, MTU2.TSR H'E8041405

	b7	b6	b5	b4	b3	b2	b1	b0
	TCFD	—	—	—	—	—	—	—
Value after reset:	1	1	0	0	0	0	0	0

- MTU3.TSR, MTU4.TSR, MTU6.TSR, MTU7.TSR

Address(es): MTU3.TSR H'E804122C, MTU4.TSR H'E804122D, MTU6.TSR H'E8041A2C, MTU7.TSR H'E8041A2D

	b7	b6	b5	b4	b3	b2	b1	b0
	TCFD	—	—	—	—	—	—	—
Value after reset:	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	The read value is undefined.The write value should be 1.	R/W
b6	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b7	TCFD	Count Direction Flag	0: TCNT counts down 1: TCNT counts up	R

TSR indicates the states of each of the channels. The MTU has a total of six TSR registers, one each for MTU1 to MTU4, MTU6, and MTU7.

TCFD Flag (Count Direction Flag)

Status flag that indicates the direction in which TCNT is counting in MTU1 to MTU4, MTU6, and MTU7.

10.2.10 Timer Buffer Operation Transfer Mode Register (TBTM)

- MTU0.TBTM

Address(es): MTU0.TBTM H'E8041326

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TTSE	TTSB	TTSA
Value after reset:	0	0	0	0	0	0	0	0

- MTU3.TBTM, MTU4.TBTM, MTU6.TBTM, MTU7.TBTM

Address(es): MTU3.TBTM H'E8041238, MTU4.TBTM H'E8041239, MTU6.TBTM H'E8041A38, MTU7.TBTM H'E8041A39

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	TTSB	TTSA
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TTSA	Timing Select A	0: When compare match A occurs in each channel, data is transferred from TGRC to TGRA 1: When TCNT is cleared in each channel, data is transferred from TGRC to TGRA	R/W
b1	TTSB	Timing Select B	0: When compare match B occurs in each channel, data is transferred from TGRD to TGRB 1: When TCNT is cleared in each channel, data is transferred from TGRD to TGRB	R/W
b2	TTSE	Timing Select E	0: When compare match E occurs in MTU0, data is transferred from MTU0.TGRF to MTU0.TGRE 1: When MTU0.TCNT is cleared in MTU0, data is transferred from MTU0.TGRF to MTU0.TGRE	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TBTM specifies the timing for transferring data from the buffer register to the timer general register in PWM mode. The MTU has a total of five TBTM registers, one each for MTU0, MTU3, MTU4, MTU6, and MTU7.

TTSA Bit (Timing Select A)

This bit specifies the timing for transferring data from TGRC to TGRA in each channel when they are used together for buffer operation. When a channel is not set to PWM mode, do not set the TTSA bit in the channel to 1.

TTSB Bit (Timing Select B)

This bit specifies the timing for transferring data from TGRD to TGRB in each channel when they are used together for buffer operation. When a channel is not set to PWM mode, do not set the TTSB bit in the channel to 1.

TTSE Bit (Timing Select E)

This bit specifies the timing for transferring data from MTU0.TGRF to MTU0.TGRE when they are used together for buffer operation.

In MTU3, MTU4, MTU6, and MTU7, this bit is reserved. It is read as 0 and the write value should be 0. When a channel is not set to PWM mode, do not set the TTSE bit in the channel to 1.

10.2.11 Timer Input Capture Control Register (TICCR)

Address(es): MTU1.TICCR H'E8041390

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	I2BE	I2AE	I1BE	I1AE
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	I1AE	Input Capture Enable	0: Does not include the MTIOC1A pin in the MTU2.TGRA input capture conditions 1: Includes the MTIOC1A pin in the MTU2.TGRA input capture conditions	R/W
b1	I1BE	Input Capture Enable	0: Does not include the TMTIOC1B pin in the MTU2.TGRB input capture conditions 1: Includes the MTIOC1B pin in the MTU2.TGRB input capture conditions	R/W
b2	I2AE	Input Capture Enable	0: Does not include the MTIOC2A pin in the MTU1.TGRA input capture conditions 1: Includes the MTIOC2A pin in the MTU1.TGRA input capture conditions	R/W
b3	I2BE	Input Capture Enable	0: Does not include the MTIOC2B pin in the MTU1.TGRB input capture conditions 1: Includes the MTIOC2B pin in the MTU1.TGRB input capture conditions	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TICCR specifies input capture conditions when MTU1.TCNT and MTU2.TCNT are cascaded. The MTU has one TICCR for MTU1.

10.2.12 Timer Synchronous Clear Register (TSYCR)

Address(es): MTU6.TSYCR H'E8041A50

	b7	b6	b5	b4	b3	b2	b1	b0
	CE0A	CE0B	CE0C	CE0D	CE1A	CE1B	CE2A	CE2B
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CE2B	Clear Enable 2B	0: Disables counter clearing by the MTU2.TGIB2 interrupt generation timing. 1: Enables counter clearing by the MTU2.TGIB2 interrupt generation timing.	R/W
b1	CE2A	Clear Enable 2A	0: Disables counter clearing by the MTU2.TGIA2 interrupt generation timing*1. 1: Enables counter clearing by the MTU2.TGIA2 interrupt generation timing*1.	R/W
b2	CE1B	Clear Enable 1B	0: Disables counter clearing by the MTU1.TGIB1 interrupt generation timing*1. 1: Enables counter clearing by the MTU1.TGIB1 interrupt generation timing*1.	R/W
b3	CE1A	Clear Enable 1A	0: Disables counter clearing by the MTU1.TGIA1 interrupt generation timing*1. 1: Enables counter clearing by the MTU1.TGIA1 interrupt generation timing*1.	R/W
b4	CE0D	Clear Enable 0D	0: Disables counter clearing by the MTU0.TGID0 interrupt generation timing*1. 1: Enables counter clearing by the MTU0.TGID0 interrupt generation timing*1.	R/W
b5	CE0C	Clear Enable 0C	0: Disables counter clearing by the MTU0.TGIC0 interrupt generation timing*1. 1: Enables counter clearing by the MTU0.TGIC0 interrupt generation timing*1.	R/W
b6	CE0B	Clear Enable 0B	0: Disables counter clearing by the MTU0.TGIB0 interrupt generation timing*1. 1: Enables counter clearing by the MTU0.TGIB0 interrupt generation timing*1.	R/W
b7	CE0A	Clear Enable 0A	0: Disables counter clearing by the MTU0.TGIA0 interrupt generation timing*1. 1: Enables counter clearing by the MTU0.TGIA0 interrupt generation timing*1.	R/W

Note 1. This does not depend on the TIERn.TGIEm bit setting. (n = 0, 1, 2; m = A, B, C, D)

TSYCR specifies synchronous clear conditions for MTU6.TCNT and MTU7.TCNT. The MTU has one TSYCR for MTU1.

CEnm Bits (Clear Enable nm; n = 0, 1, 2; m = A, B, C, D)

These bits enable or disable counter clearing by the MTUn.TGImn interrupt generation timing.

10.2.13 Timer Counter (TCNT)

- MTU0.TCNT to MTU7.TCNT

Address(es): MTU0.TCNT H'E8041306, MTU1.TCNT H'E8041386, MTU2.TCNT H'E8041406, MTU3.TCNT H'E8041210, MTU4.TCNT H'E8041212, MTU5.TCNTU H'E8041C80, MTU5.TCNTV H'E8041C90, MTU5.TCNTW H'E8041CA0, MTU6.TCNT H'E8041A10, MTU7.TCNT H'E8041A12

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Note: TCNT must not be accessed in eight bits; it should be accessed in 16 bits.

- MTU8.TCNT

Address(es): MTU8.TCNT H'E8041608

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Note: TCNT must not be accessed in 8 or 16 bits; it should be accessed in 32 bits.

MTU0.TCNT to MTU7.TCNT are 16-bit readable/writable counters and MTU8.TCNT is a 32-bit readable/writable counter. The MTU has a total of 11 TCNT counters, one each for MTU0 to MTU4, MTU6, MTU7, and MTU8 and three (TCNTU, TCNTV, and TCNTW) for MTU5. The TCNT counters in MTU0 to MTU4, MTU6, and MTU7 are initialized to 0000h by a reset, and the TCNT counter in MTU8 is initialized to 00000000h by a reset. The TCNTU, TCNTV, and TCNTW counters in MTU5 are initialized to 0000h by a reset. In MTU0 to MTU4, MTU6, and MTU7, the TCNT counters must not be accessed in 8-bit units; they should be accessed in 16-bit units. The MTU8.TCNT counter must not be accessed in 8- or 16-bit units; it should be accessed in 32-bit units.
The MTU1.TCNT and MTU2.TCNT counters are read as 0000h when TMDR3.LWA is 1. See section 10.2.5, Timer Mode Register 3 (TMDR3) for details.

10.2.14 Timer Longword Counter (TCNTLW)

Address(es): MTU1.TCNTLW H'E80413A0

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Note: TCNTLW must not be accessed in 8 or 16 bits; it should be accessed in 32 bits.

The TCNTLW counter is a 32-bit readable/writable counter. Only one counter of this type is provided, and is formed by combining MTU1.TCNT and MTU2.TCNT. Such operation is only effective when TMDR3.LWA is 1. The TCNTLW counter is initialized to 0000 0000h by a reset. This counter is read as 0000 0000h when TMDR3.LWA is 0. See section 10.2.5, Timer Mode Register 3 (TMDR3) for details. This register can only be used in 32-bit phase counting mode.

10.2.15 Timer General Register (TGR)

- MTU0.TGR to MTU7.TGR

Address(es): MTU0.TGRA H'E8041308, MTU0.TGRB H'E804130A, MTU0.TGRC H'E804130C, MTU0.TGRD H'E804130E, MTU0.TGRE H'E8041320, MTU0.TGRF H'E8041322,
MTU1.TGRA H'E8041388, MTU1.TGRB H'E804138A,
MTU2.TGRA H'E8041408, MTU2.TGRB H'E804140A,
MTU3.TGRA H'E8041218, MTU3.TGRB H'E804121A, MTU3.TGRC H'E8041224, MTU3.TGRD H'E8041226, MTU3.TGRE H'E8041272
MTU4.TGRA H'E804121C, MTU4.TGRB H'E804121E, MTU4.TGRC H'E8041228, MTU4.TGRD H'E804122A, MTU4.TGRE H'E8041274, MTU4.TGRF H'E8041276,
MTU5.TGRU H'E8041C82, MTU5.TGRV H'E8041C92, MTU5.TGRW H'E8041CA2,
MTU6.TGRA H'E8041A18, MTU6.TGRB H'E8041A1A, MTU6.TGRC H'E8041A24, MTU6.TGRD H'E8041A26, MTU6.TGRE H'E8041A72,
MTU7.TGRA H'E8041A1C, MTU7.TGRB H'E8041A1E, MTU7.TGRC H'E8041A28, MTU7.TGRD H'E8041A2A, MTU7.TGRE H'E8041A74, MTU7.TGRF H'E8041A76

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: TGR must not be accessed in eight bits; it should be accessed in 16 bits. The initial value of TGR is FFFFh.

- MTU8.TGR

Address(es): MTU8.TGRA H'E804160C, MTU8.TGRB H'E8041610, MTU8.TGRC H'E8041614, MTU8.TGRD H'E8041618

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: TGR must not be accessed in 8 or 16 bits; it should be accessed in 32 bits.

The MTU0.TGR to MTU7.TGR registers are 16-bit readable/writable registers; the MTU8.TGR register is a 32-bit readable/writable register. The MTU has a total of 39 TGR registers, six for MTU0, two each for MTU1 and MTU2, five each for MTU3 and MTU6, six each for MTU4 and MTU7, three for MTU5 and four for MTU8.
The TGRA, TGRB, TGRC, and TGRD registers function as either output compare or input capture registers. The TGRC and TGRD registers for MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8 can also be designated for operation as buffer registers. TGR buffer register combinations are TGRA and TGRC, and TGRB and TGRD.
MTU0.TGRE and MTU0.TGRF function as compare registers. When the MTU0.TCNT count matches the MTU0.TGRE value, an A/D converter start request can be issued. The TGRF register can also be designated for operation as a buffer register. TGR buffer register combination is TGRE and TGRF. MTU5.TGRU, MTU5.TGRV, and MTU5.TGRW function as compare match, input capture, or external pulse width measurement registers.
The MTU1.TGRA, MTU2.TGRA, MTU1.TGRB, and MTU2.TGRB registers are read as 0000h when TMDR3.LWA is 1. See section 10.2.5, Timer Mode Register 3 (TMDR3) for details.

10.2.16 Timer Longword General Registers (TGRALW and TGRBLW)

Address(es): MTU1.TGRALW H'E80413A4, MTU1.TGRBLW H'E80413A8

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: TGRALW and TGRBLW must not be accessed in 8 or 16 bits; they should be accessed in 32 bits.

The TGRALW (TGRBLW) register is a 32-bit readable/writable register. Two general registers of this type are provided, and are formed by combining the TGRA (TGRB) registers in MTU1 and MTU2. Such operation is only effective when TMDR3.LWA is 1.

The TGRALW (TGRBLW) register is initialized to FFFF FFFFh by a reset, but it is read as 0000 0000h when TMDR3.LWA is 0. See section 10.2.5, Timer Mode Register 3 (TMDR3) for details.

The TGRALW (TGRBLW) register functions as an output compare or input capture register when TMDR3.LWA is 1. This register can only be used in 32-bit phase counting mode.

10.2.17 Timer Start Registers (TSTRA, TSTRB, and TSTR)

- MTU.TSTRA (MTU0, MTU1, MTU2, MTU3, MTU4, MTU8)

Address(es): MTU.TSTRA H'E8041280

	b7	b6	b5	b4	b3	b2	b1	b0
	CST4	CST3	—	—	CST8	CST2	CST1	CST0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CST0	Counter Start 0	0: MTU0.TCNT counting is stopped 1: MTU0.TCNT performs count operation	R/W
b1	CST1	Counter Start 1	0: MTU1.TCNT counting is stopped 1: MTU1.TCNT performs count operation	R/W
b2	CST2	Counter Start 2	0: MTU2.TCNT counting is stopped 1: MTU2.TCNT performs count operation	R/W
b3	CST8	Counter start 8	0: MTU8.TCNT counting is stopped 1: MTU8.TCNT performs count operation.	R/W
b5, b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	CST3	Counter Start 3	0: MTU3.TCNT counting is stopped 1: MTU3.TCNT performs count operation	R/W
b7	CST4	Counter Start 4	0: MTU4.TCNT counting is stopped 1: MTU4.TCNT performs count operation	R/W

Note: When 1 is written to a bit in TCSYSTR, the corresponding bit in TSTRA is also set to 1 automatically.

The TSTRA register starts or stops TCNT operation in MTU0 to MTU4 and MTU8.
TSTRA starts or stops TCNT operation in MTU0 to MTU4.
TSTRB starts or stops TCNT operation in MTU6 and MTU7.
TSTR starts or stops TCNT operation in MTU5.
Before setting the operating mode in TMDR1 or setting the TCNT count clock in TCR, be sure to stop the TCNT counting.

CSTn Bits (Counter Start n) (n = 0, 1, 2, 3, 4, 8)

Each bit starts or stops TCNT in the corresponding channel.
If 0 is written to the CSTn bit during operation with the MTIOC pin designated for output, the counter stops. In this case, the initial output value specified by the TOCR1A or TOCR2A register is output from the MTIOC pin in complementary PWM mode or reset-synchronized PWM mode.
In the other modes, the level of the output compare signal from the MTIOC pin is retained. If TIOR is written to while the CSTn bit is 0, the level of output from the pin will be updated to the specified initial output value.

- MTU.TSTRB(MTU6, MTU7)

Address(es): MTU.TSTRB H'E8041A80

	b7	b6	b5	b4	b3	b2	b1	b0
	CST7	CST6	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	CST6	Counter Start 6	0: MTU6.TCNT counting is stopped 1: MTU6.TCNT performs count operation	R/W
b7	CST7	Counter Start 7	0: MTU7.TCNT counting is stopped 1: MTU7.TCNT performs count operation	R/W

Note: When 1 is written to a bit in TCSYSTR, the corresponding bit in TSTRB is also set to 1 automatically.

CSTn Bits (Counter Start n) (n = 6, 7)

Each bit starts or stops TCNT in the corresponding channel.

If 0 is written to the CSTn bit during operation with the MTIOC pin designated for output, the counter stops. In this case, the initial output value specified by the TOCR1B or TOCR2B register is output from the MTIOC pin in complementary PWM mode or reset-synchronized PWM mode.

In the other modes, the level of the output compare signal from the MTIOC pin is retained. If TIOR is written to while the CSTn bit is 0, the level of output from the pin will be updated to the specified initial output value.

- MTU5.TSTR(MTU5)

Address(es): MTU5.TSTR H'E8041CB4

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CSTU5	CSTV5	CSTW5
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CSTW5	Counter Start W5	0: MTU5.TCNTW counting is stopped 1: MTU5.TCNTW performs count operation	R/W
b1	CSTV5	Counter Start V5	0: MTU5.TCNTV counting is stopped 1: MTU5.TCNTV performs count operation	R/W
b2	CSTU5	Counter Start U5	0: MTU5.TCNTU counting is stopped 1: MTU5.TCNTU performs count operation	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

10.2.18 Timer Synchronous Registers (TSYRA and TSYRB)

- MTU.TSYRA(MTU0, MTU1, MTU2, MTU3, MTU4)

Address(es): MTU.TSYRA H'E8041281

	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC4	SYNC3	—	—	—	SYNC2	SYNC1	SYNC0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SYNC0	Timer Synchronous Operation 0	0: MTU0.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU0.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)	R/W
b1	SYNC1	Timer Synchronous Operation 1	0: MTU1.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU1.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)	R/W
b2	SYNC2	Timer Synchronous Operation 2	0: MTU2.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU2.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)	R/W
b5 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	SYNC3	Timer Synchronous Operation 3	0: MTU3.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU3.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)	R/W
b7	SYNC4	Timer Synchronous Operation 4	0: MTU4.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU4.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)	R/W

TSYRA selects independent operation or synchronous operation of TCNT in MTU0 to MTU4.
TSYRB selects independent operation or synchronous operation of TCNT in MTU6 and MTU7.
A channel performs synchronous operation when the corresponding bit in TSYR is set to 1.

SYNCn Bits (Timer Synchronous Operation n) (n = 0, 1, 2, 3, 4)

Each bit selects whether operation is independent of or synchronized with other channels.
When synchronous operation is selected, the TCNT synchronous setting of multiple channels and synchronous clearing by counter clearing on another channel are possible.
To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of the TCR.CCLR[2:0] bits.

- MTU.TSYRB(MTU6, MTU7)

Address(es): MTU.TSYRB H'E8041A81

	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC7	SYNC6	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	SYNC6	Timer Synchronous Operation 6	0: MTU6.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU6.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)	R/W
b7	SYNC7	Timer Synchronous Operation 7	0: MTU7.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU7.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)	R/W

SYNCn Bits (Timer Synchronous Operation n) (n = 6, 7)

Each bit selects whether operation is independent of or synchronized with other channels.

When synchronous operation is selected, the TCNT synchronous setting of multiple channels and synchronous clearing by counter clearing on another channel are possible.

To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of bits TCR.CCLR[2:0].

10.2.19 Timer Counter Synchronous Start Register (TCSYSTR)

Address(es): MTU.TCSYSTR H'E8041282

	b7	b6	b5	b4	b3	b2	b1	b0
	SCH0	SCH1	SCH2	SCH3	SCH4	—	SCH6	SCH7
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SCH7	Synchronous Start 7	0: Does not specify synchronous start for MTU7.TCNT 1: Specifies synchronous start for MTU7.TCNT	R/(W)*1
b1	SCH6	Synchronous Start 6	0: Does not specify synchronous start for MTU6.TCNT 1: Specifies synchronous start for MTU6.TCNT	R/(W)*1
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R
b3	SCH4	Synchronous Start 4	0: Does not specify synchronous start for MTU4.TCNT 1: Specifies synchronous start for MTU4.TCNT	R/(W)*1
b4	SCH3	Synchronous Start 3	0: Does not specify synchronous start for MTU3.TCNT 1: Specifies synchronous start for MTU3.TCNT	R/(W)*1
b5	SCH2	Synchronous Start 2	0: Does not specify synchronous start for MTU2.TCNT 1: Specifies synchronous start for MTU2.TCNT	R/(W)*1
b6	SCH1	Synchronous Start 1	0: Does not specify synchronous start for MTU1.TCNT 1: Specifies synchronous start for MTU1.TCNT	R/(W)*1
b7	SCH0	Synchronous Start 0	0: Does not specify synchronous start for MTU0.TCNT 1: Specifies synchronous start for MTU0.TCNT	R/(W)*1

Note 1. Only 1 can be written to this bit. It is automatically cleared to 0 when counting begins.

TCSYSTR specifies synchronous start of the counters.

SCH7 Bit (Synchronous Start 7)

This bit controls synchronous start of TCNT in MTU7.

[Clearing condition]

- When the CST7 bit in TSTRB is set to 1 while SCH7 = 1

SCH6 Bit (Synchronous Start 6)

This bit controls synchronous start of TCNT in MTU6.

[Clearing condition]

- When the CST6 bit in TSTRB is set to 1 while SCH6 = 1

SCH4 Bit (Synchronous Start 4)

This bit controls synchronous start of TCNT in MTU4.

[Clearing condition]

- When the CST4 bit in TSTRA is set to 1 while SCH4 = 1

SCH3 Bit (Synchronous Start 3)

This bit controls synchronous start of TCNT in MTU3.

[Clearing condition]

- When the CST3 bit in TSTRA is set to 1 while SCH3 = 1

SCH2 Bit (Synchronous Start 2)

This bit controls synchronous start of TCNT in MTU2.
[Clearing condition]

- When the CST2 bit in TSTRA is set to 1 while SCH2 = 1

SCH1 Bit (Synchronous Start 1)

This bit controls synchronous start of TCNT in MTU1.
[Clearing condition]

- When the CST1 bit in TSTRA is set to 1 while SCH1 = 1

SCH0 Bit (Synchronous Start 0)

This bit controls synchronous start of TCNT in MTU0.
[Clearing condition]

- When the CST0 bit in TSTRA is set to 1 while SCH0 = 1

10.2.20 Timer Read/Write Enable Registers (TRWERA and TRWERB)

Address(es): MTU.TRWERA H'E8041284, MTU.TRWERB H'E8041A84

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	RWE
Value after reset:	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	RWE	Read/Write Enable	0: Read/write access to the registers is disabled 1: Read/write access to the registers is enabled	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TRWERA enables or disables access to the registers and counters that have write-protection capability against accidental modification in MTU3 and MTU4.
TRWERB enables or disables access to the registers and counters that have write-protection capability against accidental modification in MTU6 and MTU7.

RWE Bit (Read/Write Enable)

This bit enables or disables access to the registers that have write-protection capability against accidental modification.
[Clearing condition]
When 0 is written to the RWE bit after reading RWE = 1

- Registers and Counters having Write-Protection Capability against Accidental Modification (TRWERA)
  24 registers: MTUn.TCR, MTUn.TCR2, MTUn.TMDR1, MTUn.TIORH, MTUn.TIORL, MTUn.TIER, MTUn.TGRA, MTUn.TGRB, TOERA, TOCR1A, TOCR2A, TGCRA, TCDRA, TDDRA, and MTUn.TCNT (n = 3, 4)
- Registers and Counters having Write-Protection Capability against Accidental Modification (TRWERB)
  23 registers: MTUn.TCR, MTUn.TCR2, MTUn.TMDR1, MTUn.TIORH, MTUn.TIORL, MTUn.TIER, MTUn.TGRA, MTUn.TGRB, TOERB, TOCR1B, TOCR2B, TCDRB, TDDRB, and MTUn.TCNT (n = 6, 7)

10.2.21 Timer Output Master Enable Registers (TOERA and TOERB)

- MTU.TOERA

Address(es): MTU.TOERA H'E804120A

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
Value after reset:	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OE3B	Master Enable MTIOC3B	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b1	OE4A	Master Enable MTIOC4A	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b2	OE4B	Master Enable MTIOC4B	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b3	OE3D	Master Enable MTIOC3D	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b4	OE4C	Master Enable MTIOC4C	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b5	OE4D	Master Enable MTIOC4D	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b7, b6	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

Note 1. To output the inactive level from each pin when the MTU output is set to disabled, first set the data direction register (PDR) and port output data register (PODR) of I/O ports to output the inactive level from general I/O ports, and then set the port mode register (PMR) to use general I/O ports. For details, refer to section 51, GPIO.

TOERA enables or disables output settings for output pins MTIOC4D, MTIOC4C, MTIOC3D, MTIOC4B, MTIOC4A, and MTIOC3B.

These pins do not output values correctly if the bits of the TOERA register are not appropriately set. Write a desired value to the TOERA register before setting up the TIOR registers in MTU3 and MTU4.

Set the TOERA register after clearing the CST3 and CST4 bits in the TSTRA register to 0 (see Figure 10.44 and Figure 10.48).

- MTU.TOERB

Address(es): MTU.TOERB H'E8041A0A

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OE7D	OE7C	OE6D	OE7B	OE7A	OE6B
Value after reset:	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OE6B	Master Enable MTIOC6B	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b1	OE7A	Master Enable MTIOC7A	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b2	OE7B	Master Enable MTIOC7B	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b3	OE6D	Master Enable MTIOC6D	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b4	OE7C	Master Enable MTIOC7C	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b5	OE7D	Master Enable MTIOC7D	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b7, b6	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

Note 1. To output the inactive level from each pin when the MTU output is set to disabled, first set the data direction register (PDR) and port output data register (PODR) of I/O ports to output the inactive level from general I/O ports, and then set the port mode register (PMR) to use general I/O ports. For details, refer to section 51, GPIO.

TOERB enables or disables output settings for output pins MTIOC7D, MTIOC7C, MTIOC6D, MTIOC7B, MTIOC7A, and MTIOC6B.

These pins do not output values correctly if the bits of the TOERB register are not appropriately set. Write a desired value to the TOERB register before setting up the TIOR registers in MTU6 and MTU7.

Set the TOERB register after clearing the CST6 and CST7 bits in the TSTRB register to 0 (see Figure 10.44 and Figure 10.48).

10.2.22 Timer Output Control Registers 1 (TOCR1A and TOCR1B)

Address(es): MTU.TOCR1A H'E804120E, MTU.TOCR1B H'E8041A0E

	b7	b6	b5	b4	b3	b2	b1	b0
	—	PSYE	—	—	TOCL	TOCS	OLSN	OLSP
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OLSP	Output Level Select P*1*3	See Table 10.43.	R/W
b1	OLSN	Output Level Select N*1*3	See Table 10.44.	R/W
b2	TOCS	TOC Select	0: TOCR1j setting is selected (j = A, B) 1: TOCR2j setting is selected	R/W
b3	TOCL	TOC Register Write Protection*2*4	0: Write access to the TOCS, OLSN, and OLSP bits is enabled 1: Write access to the TOCS, OLSN, and OLSP bits is disabled	R/W
b5, b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	PSYE	PWM Synchronous Output Enable	0: Toggle output is disabled 1: Toggle output is enabled	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Note 1. Clearing the TOCR1j.TOCS bit to 0 makes this bit setting valid.
Note 2. Setting the TOCR1j.TOCL bit to 1 prevents accidental modification when the CPU goes out of control.
Note 3. If dead-time is not generated, the negative-phase output is the exact inverse of the positive-phase output. In this case, only the OLSP bit is valid.
Note 4. This bit can be set to 1 only once after a reset. After 1 is written, 0 cannot be written to the bit.

TOCR1A and TOCR1B enable or disable PWM-synchronized toggle output in complementary PWM mode and reset-synchronized PWM mode, and control inversion of PWM output level.

OLSP Bit (Output Level Select P)

This bit selects the positive-phase output level in reset-synchronized PWM mode and complementary PWM mode. The initial output is selected while the counter is stopped.

OLSN Bit (Output Level Select N)

This bit selects the negative-phase output level in reset-synchronized PWM mode and complementary PWM mode. The initial output is selected while the counter is stopped.

TOCS Bit (TOC Select)

This bit selects either the TOCR1j or TOCR2j (j = A, B) setting to be used for the output level in complementary PWM mode and reset-synchronized PWM mode.

TOCL Bit (TOC Register Write Protection)

This bit enables or disables write access to the TOCS, OLSN, and OLSP bits in TOCR1j (j = A, B).

PSYE Bit (PWM Synchronous Output Enable)

This bit enables or disables toggle output synchronized with the PWM cycle.

Table 10.43 Output Level Select Function

Bit 0	Function			
			Compare Match Output	
OLSP	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Table 10.44 Output Level Select Function

Bit 1	Function			
			Compare Match Output	
OLSN	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Figure 10.3 shows an example of output in complementary PWM mode (one phase) when OLSN = 1 and OLSP = 1.

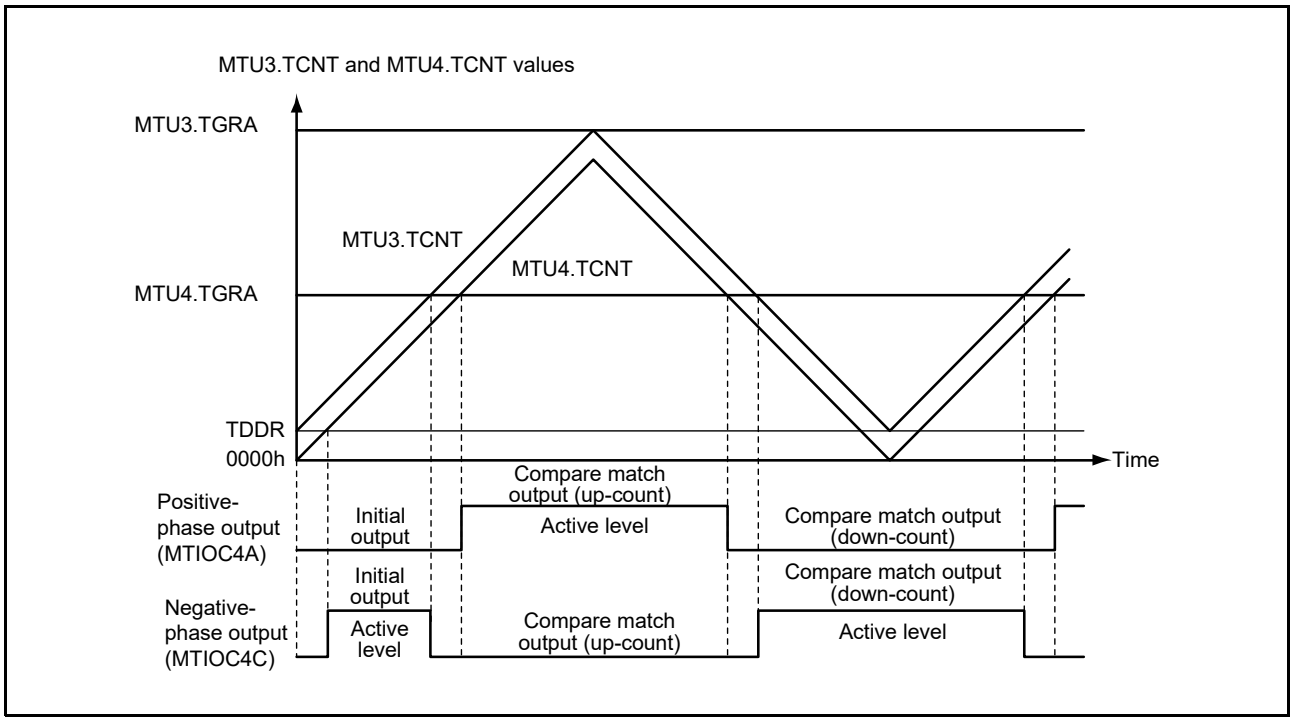


Figure 10.3 Example of Output in Complementary PWM Mode

10.2.23 Timer Output Control Registers 2 (TOCR2A and TOCR2B)

Address(es): MTU.TOCR2A H'E804120F, MTU.TOCR2B H'E8041A0F

	b7	b6	b5	b4	b3	b2	b1	b0
	BF[1:0]		OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OLS1P	Output Level Select 1P*1*2	This bit selects the output level on MTIOC3B or MTIOC6B in reset-synchronized PWM mode and complementary PWM mode. See Table 10.45.	R/W
b1	OLS1N	Output Level Select 1N*1*2	This bit selects the output level on MTIOC3D or MTIOC6D in reset-synchronized PWM mode and complementary PWM mode. See Table 10.46.	R/W
b2	OLS2P	Output Level Select 2P*1*2	This bit selects the output level on MTIOC4A or MTIOC7Ain reset-synchronized PWM mode and complementary PWM mode. See Table 10.47.	R/W
b3	OLS2N	Output Level Select 2N*1*2	This bit selects the output level on MTIOC4C or MTIOC7C in reset-synchronized PWM mode and complementary PWM mode. See Table 10.48.	R/W
b4	OLS3P	Output Level Select 3P*1*2	This bit selects the output level on MTIOC4B or MTIOC7B in reset-synchronized PWM mode and complementary PWM mode. See Table 10.49.	R/W
b5	OLS3N	Output Level Select 3N*1*2	This bit selects the output level on MTIOC4D or MTIOC7D in reset-synchronized PWM mode and complementary PWM mode. See Table 10.50.	R/W
b7, b6	BF[1:0]	TOLBR Buffer Transfer Timing Select	These bits select the timing for transferring data from TOLBRj to TOCR2j. See Table 10.51 for details.	R/W

j = A, B

Note 1. Setting the TOCR1j.TOCS bit to 1 makes this bit setting valid.

Note 2. If dead-time is not generated, the negative-phase output is the exact inverse of the positive-phase output. In this case, only the OLSiP bits are valid (i = 1 to 3).

TOCR2A and TOCR2B control inversion of PWM output level in complementary PWM mode and reset-synchronized PWM mode.

The initial output is selected while the counter is stopped.

Table 10.45 MTIOCmB Output Level Select Function

Bit 0	Function			
			Compare Match Output	
OLS1P	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

m = 3, 6

Table 10.46 MTIOCmD Output Level Select Function

Bit 1	Function			
			Compare Match Output	
OLS1N	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

m = 3, 6

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Table 10.47 MTIOCmA Output Level Select Function

Bit 2	Function			
			Compare Match Output	
OLS2P	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

m = 4, 7

Table 10.48 MTIOCmC Output Level Select Function

Bit 3	Function			
			Compare Match Output	
OLS2N	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

m = 4, 7

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Table 10.49 MTIOCmB Output Level Select Function

Bit 4	Function			
			Compare Match Output	
OLS3P	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

m = 4, 7

Table 10.50 MTIOCmD Output Level Select Function

Bit 5	Function			
			Compare Match Output	
OLS3N	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

m = 4, 7

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Table 10.51 Setting of TOCR2j.BF[1:0] Bits

Bit 7	Bit 6	Description	
BF1	BF0	Complementary PWM Mode	Reset-Synchronized PWM Mode
0	0	Does not transfer data from the buffer register (TOLBRj) to TOCR2j.	Does not transfer data from the buffer register (TOLBRj) to TOCR2j.
	1	Transfers data from the buffer register (TOLBRj) to TOCR2j at the crest of the MTUn.TCNT count.	Transfers data from the buffer register (TOLBRj) to TOCR2j when MTUm.TCNT or MTUn.TCNT is cleared.
1	0	Transfers data from the buffer register (TOLBRj) to TOCR2j at the trough of the MTUn.TCNT count.	Setting prohibited
	1	Transfers data from the buffer register (TOLBRj) to TOCR2j at the crest and trough of the MTUn.TCNT count.	Setting prohibited

n = 4, 7; m = 3, 6; j = A, B

10.2.24 Timer Output Level Buffer Registers (TOLBRA and TOLBRB)

Address(es): MTU.TOLBRA H'E8041236, MTU.TOLBRB H'E8041A36

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OLS1P	Output Level Select 1P	Specify the buffer value to be transferred to the OLS1P bit in TOCR2j.	R/W
b1	OLS1N	Output Level Select 1N	Specify the buffer value to be transferred to the OLS1N bit in TOCR2j.	R/W
b2	OLS2P	Output Level Select 2P	Specify the buffer value to be transferred to the OLS2P bit in TOCR2j.	R/W
b3	OLS2N	Output Level Select 2N	Specify the buffer value to be transferred to the OLS2N bit in TOCR2j.	R/W
b4	OLS3P	Output Level Select 3P	Specify the buffer value to be transferred to the OLS3P bit in TOCR2j.	R/W
b5	OLS3N	Output Level Select 3N	Specify the buffer value to be transferred to the OLS3N bit in TOCR2j.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

j = A, B

TOLBRA and TOLBRB are buffer registers for TOCR2A and TOCR2B and specify the PWM output level in complementary PWM mode and reset-synchronized PWM mode.

Figure 10.4 shows an example of the PWM output level setting procedure in buffer operation.

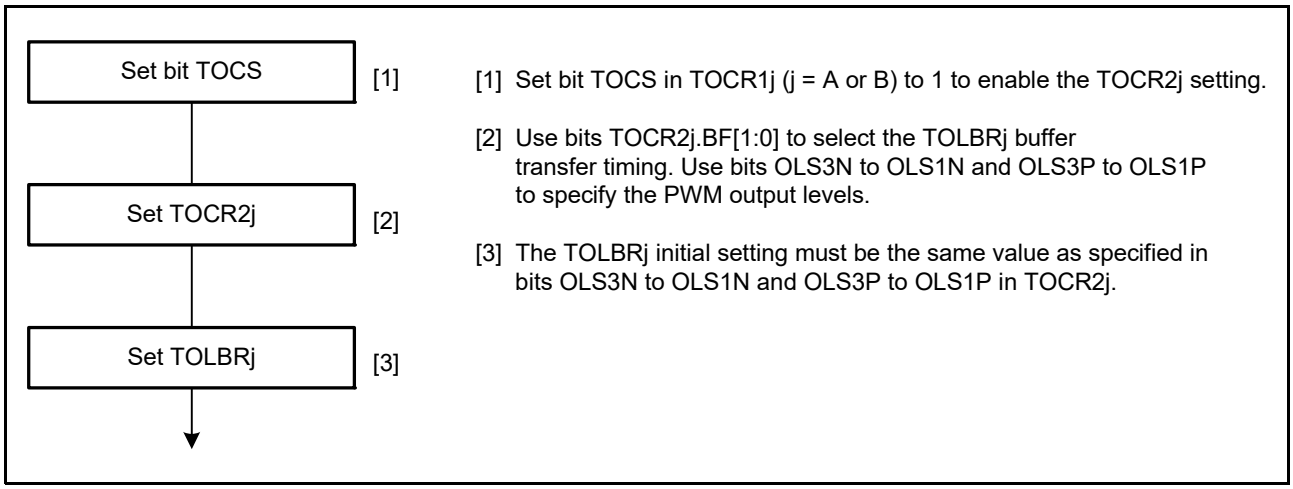


Figure 10.4 Example of PWM Output Level Setting Procedure in Buffer Operation

10.2.25 Timer Gate Control Register A (TGCRA)

Address(es): MTU.TGCRA H'E804120D

	b7	b6	b5	b4	b3	b2	b1	b0
	—	BDC	N	P	FB	WF	VF	UF
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	UF	Output Phase Switch	These bits turn on or off the positive-phase/negative-phase output. The setting of these bits is valid only when the FB bit is set to 1. In this case, the setting of b0 to b2 is used instead of the external input. See Table 10.52.	R/W
b1	VF			R/W
b2	WF			R/W
b3	FB	External Feedback Signal Enable	0: Output is switched by external input (input sources are TGRA, TGRB, and TGRC input capture signals in MTU0) 1: Output is switched by software (TGCRA's UF, VF, and WF settings)	R/W
b4	P	Positive-Phase Output (P) Control	0: Level output 1: Reset-synchronized PWM or complementary PWM output	R/W
b5	N	Negative-Phase Output (N) Control	0: Level output 1: Reset-synchronized PWM or complementary PWM output	R/W
b6	BDC	Brushless DC Motor	0: Ordinary output 1: Functions of this register are effective.	R/W
b7	—	Reserved	This bit is read as 1. The write value should be 1.	R/W

TGCRA controls the output waveform necessary for brushless DC motor control in reset-synchronized PWM mode and complementary PWM mode. TGCRA register settings are ineffective for anything other than complementary PWM mode and reset-synchronized PWM mode.

UF, VF, and WF Bits (Output Phase Switch)

The setting of these bits is valid only when the FB bit is set to 1. In this case, the setting of b0 to b2 is used instead of the external input. See Table 10.52 for details.

FB Bit (External Feedback Signal Enable)

This bit selects whether the positive-/negative-phase output is switched automatically with the TGRA, TGRB, and TGRC input capture signals in MTU0 or by writing 0 or 1 to bits 2 to 0 in TGCRA.
When the FB bit in the TGCRA register is 0, the output signals from MTU3 and MTU4 are switched automatically with the TGRA, TGRB, and TGRC input capture signals in MTU0.

P Bit (Positive-Phase Output (P) Control)

This bit selects either the level output or the reset-synchronized PWM/complementary PWM output for the positive-phase output pins (MTIOC3B, MTIOC4A, and MTIOC4B pins).

N Bit (Negative-Phase Output (N) Control)

This bit selects the level output or the reset-synchronized PWM/complementary PWM output for the negative-phase output pins (MTIOC3D, MTIOC4C, and MTIOC4D pins).

BDC Bit (Brushless DC Motor)

This bit selects whether to make the functions of TGCRA effective or ineffective.

Table 10.52 Output Level Select Function

Bit 2	Bit 1	Bit 0	Function					
			MTIOC3B	MTIOC4A	MTIOC4B	MTIOC3D	MTIOC4C	MTIOC4D
WF	VF	UF	U Phase	V Phase	W Phase	U Phase	V Phase	W Phase
0	0	0	OFF	OFF	OFF	OFF	OFF	OFF
		1	ON	OFF	OFF	OFF	OFF	ON
	1	0	OFF	ON	OFF	ON	OFF	OFF
		1	OFF	ON	OFF	OFF	OFF	ON
1	0	0	OFF	OFF	ON	OFF	ON	OFF
		1	ON	OFF	OFF	OFF	ON	OFF
	1	0	OFF	OFF	ON	ON	OFF	OFF
		1	OFF	OFF	OFF	OFF	OFF	OFF

10.2.26 Timer Subcounters (TCNTSA and TCNTSB)

Address(es): MTU.TCNTSA H'E8041220, MTU.TCNTSB H'E8041A20

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Note: TCNTSA and TCNTSB must not be accessed in eight bits; it should be accessed in 16 bits.

TCNTSA and TCNTSB are 16-bit read-only counters that are used only in complementary PWM mode.
The initial value of TCNTSA and TCNTSB after a reset is 0000h.

10.2.27 Timer Cycle Data Registers (TCDRA and TCDRB)

Address(es): MTU.TCDRA H'E8041214, MTU.TCDRB H'E8041A14

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: TCDRA and TCDRB must not be accessed in eight bits; it should be accessed in 16 bits.

TCDRA and TCDRB are 16-bit readable/writable registers used only in complementary PWM mode. Set half the PWM carrier cycle as the TCDRA and TCDRB values. The TCDRA and TCDRB registers are constantly compared with the TCNTSA and TCNTSB counters in complementary PWM mode, respectively. When a match occurs, the TCNTSA and TCNTSB counters switch the count direction (down-count to up-count). The initial value of TCDRA and TCDRB after a reset is FFFFh.

10.2.28 Timer Cycle Buffer Registers (TCBRA and TCBRB)

Address(es): MTU.TCBRA H'E8041222, MTU.TCBRB H'E8041A22

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: TCBRA and TCBRB must not be accessed in eight bits; it should be accessed in 16 bits.

TCBRA and TCBRB are 16-bit readable/writable registers, used only in complementary PWM mode, that function as buffer registers for TCDRA and TCDRB. The TCBRA and TCBRB values are transferred to TCDRA and TCDRB with the transfer timing set in TMDR1. The initial value of TCBRA and TCBRB after a reset is FFFFh.

10.2.29 Timer Dead Time Data Registers (TDDRA and TDDRB)

Address(es): MTU.TDDRA H'E8041216, MTU.TDDRB H'E8041A16

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: TDDRA and TDDRB must not be accessed in eight bits; it should be accessed in 16 bits.

TDDRA and TDDRB are 16-bit readable/writable registers, used only in complementary PWM mode, that specify the MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) counter offset value. In complementary PWM mode, when the MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) counters are cleared and then restarted, the TDDRA (TDDRB) value is loaded into the MTU3.TCNT (MTU6.TCNT) counter and the count operation starts. The initial value of TDDRA and TDDRB after a reset is FFFFh.

10.2.30 Timer Dead Time Enable Registers (TDERA and TDERB)

Address(es): MTU.TDERA H'E8041234, MTU.TDERB H'E8041A34

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TDER
Value after reset:	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	TDER	Dead Time Enable	0: No dead time is generated 1: Dead time is generated*1	R/(W)
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. TDDRA and TDDRB must be set to 1 or a larger value.

TDERA and TDERB control dead time generation in complementary PWM mode. The MTU has one TDER each for MTU3 and MTU6. TDERA and TDERB should be modified only while TCNT stops.

TDER Bit (Dead Time Enable)

This bit specifies whether to generate dead time.

[Clearing condition]

- When 0 is written to TDER after reading TDER = 1

10.2.31 Timer Buffer Transfer Set Registers (TBTERA and TBTERB)

Address(es): MTU.TBTERA H'E8041232, MTU.TBTERB H'E8041A32

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	BTE[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	BTE[1:0]	Buffer Transfer Disable and Interrupt Skipping Link Setting	These bits enable or disable transfer from the buffer registers*1 used in complementary PWM mode to the temporary registers, and specify whether to link the transfer with interrupt skipping function 1. For details, see Table 10.53.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Applicable buffer registers (TBTERA):
MTU3.TGRC, MTU3.TGRD, MTU4.TGRC, MTU4.TGRD, and TCBRA
Applicable buffer registers (TBTERB):
MTU6.TGRC, MTU6.TGRD, MTU7.TGRC, MTU7.TGRD, and TCBRB

TBTERA and TBTERB enable or disable transfer from the buffer registers used in complementary PWM mode to the temporary registers, and specify whether to link the transfer with interrupt skipping 1 operation.

Table 10.53 Setting of BTE[1:0] Bits in TBTERA and TBTERB

Bit 1	Bit 0	
BTE1	BTE0	Description
0	0	Enables transfer from the buffer registers to the temporary registers*1 and does not link the transfer with interrupt skipping function 1.
	1	Disables transfer from the buffer registers to the temporary registers.
1	0	Links transfer from the buffer registers to the temporary registers with interrupt skipping function 1.*2
	1	Setting prohibited

Note 1. Data is transferred according to the MD3 to MD0 bit setting in TMDR1. For details, refer to section 10.3.8, Complementary PWM Mode.

Note 2. When interrupt skipping is disabled (the T3AEN and T4VEN (T6AEN and T7VEN) bits are cleared to 0 in the timer interrupt skipping set register (TITCR1A (TITCR1B)) or the skipping count set bits (T3ACOR and T4VCOR (T6ACOR and T7VCOR)) in TITCR1A (TITCR1B) are cleared to 0), be sure to disable link of buffer transfer with interrupt skipping (clear the BTE1 bit in the timer buffer transfer set register (TBTERA (TBTERB)) to 0). If link with interrupt skipping is enabled while interrupt skipping is disabled, buffer transfer will not be performed.

10.2.32 Timer Waveform Control Registers (TWCRA and TWCRB)

Address(es): MTU.TWCRA H'E8041260, MTU.TWCRB H'E8041A60

	b7	b6	b5	b4	b3	b2	b1	b0
	CCE	—	—	—	—	—	SCC	WRE
Value after reset:	0*2	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	WRE	Waveform Retain Enable	0: Initial values specified in TOCR1A and TOCR2A (TOCR1B and TOCR2B) are output 1: Initial output is inhibited	R/(W) *3
b1	SCC*1*3	Synchronous Clearing Control	(Only valid in register TWCRB) 0: Clearing of MTU6.TCNT and MTU7.TCNT in response to synchronous clearing for MTU0, MTU1, MTU2–MTU6, MTU7 is enabled. 1: Clearing of MTU6.TCNT and MTU7.TCNT in response to synchronous clearing for MTU0, MTU1, MTU2–MTU6, MTU7 is disabled.	R/(W)
b6 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	CCE*2	Compare Match Clear Enable	0: Counters are not cleared at MTU3.TGRA (MTU6.TGRA) compare match 1: Counters are cleared at MTU3.TGRA (MTU6.TGRA) compare match	R/(W)

Note 1. This bit is only valid in register TWCRB and is a reserved bit in register TWCRA.
Note 2. Do not set to 1 when complementary PWM mode 1 is not selected.
Note 3. Do not set to 1 when complementary PWM mode is not selected.

TWCRA and TWCRB control the output waveform when synchronous counter clearing occurs in MTU3.TNCT and MTU4.TNCT (MTU6.TNCT and MTU7.TNCT) in complementary PWM mode and specifies whether to clear the counters at MTU3.TGRA (MTU6.TGRA) compare match.
The CCE bit and WRE bit in TWCRA and TWCRB should be modified only while TCNT stops.

WRE Bit (Waveform Retain Enable)

This bit selects the waveform output when synchronous counter clearing occurs in complementary PWM mode.
The initial output is inhibited with this function only when synchronous clearing occurs within the Tb interval at the trough in complementary PWM mode. When synchronous clearing occurs outside this interval, the initial values specified in TOCR1A and TOCR2A (TOCR1B and TOCR2B) are output regardless of the WRE bit setting. The initial values specified in TOCR1A and TOCR2A (TOCR1B and TOCR2B) are also output when synchronous clearing occurs in the Tb interval at the trough immediately after MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT) start operation.
For the Tb interval at the trough in complementary PWM mode, see Figure 10.50.
[Setting condition]

- When 1 is written to the WRE bit after reading WRE = 0

SCC Bit (Synchronous Clearing Control)

The setting of this bit selects whether MTU6.TCNT and MTU7.TCNT are or are not cleared when counter-synchronous clearing is generated for MTU0, MTU1, MTU2–MTU6, MTU7 in complementary PWM mode.
Make the complementary PWM mode settings for MTU6 and MTU7 when this function is in use. When writing a new value to the SCC bit while the counter is operating, do not change the values of the CCE and WRE bits.
Synchronous clearing from the MTU module only becomes disabled due to the setting of the SCC bit when synchronous clearing is generated outside the Tb interval in the trough. If synchronous clearing is generated within the Tb interval in the trough including immediately after the value at which MTU6.TCNT and MTU7.TCNT start, MTU6.TCNT and MTU7.TCNT are cleared.
Regarding the Tb interval in the trough in complementary PWM mode, see Figure 10.50.

[Setting condition]

- Writing of 1 to the SCC bit after reading it as 0

The corresponding bit in register TWCRA is reserved and is read as 0. When writing to TWCRA, write 0 to this bit.

CCE Bit (Compare Match Clear Enable)

This bit specifies whether to clear counters at MTU3.TGRA (MTU6.TGRA) compare match in complementary PWM mode.

[Setting condition]

- When 1 is written to CCE after reading CCE = 0

10.2.33 Noise Filter Control Register n (NFCRn) (n = 0 to 4, 6, 7, 8, C)

- MTU0.NFCR0, MTU1.NFCR1, MTU2.NFCR2, MTU3.NFCR3, MTU4.NFCR4, MTU6.NFCR6, MTU7.NFCR7, MTU8.NFCR8

Address(es): MTU0.NFCR0 H'E8041290, MTU1.NFCR1 H'E8041291, MTU2.NFCR2 H'E8041292, MTU3.NFCR3 H'E8041293, MTU4.NFCR4 H'E8041294, MTU6.NFCR6 H'E8041A93, MTU7.NFCR7 H'E8041A94, MTU8.NFCR8 H'E8041298

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	NFCS[1:0]		NFDEN	NFCEN	NFBEN	NFAEN
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NFAEN	Noise Filter A Enable	0: The noise filter for the MTIOCnA pin is disabled. 1: The noise filter for the MTIOCnA pin is enabled.	R/W
b1	NFBEN	Noise Filter B Enable	0: The noise filter for the MTIOCnB pin is disabled. 1: The noise filter for the MTIOCnB pin is enabled.	R/W
b2	NFCEN	Noise Filter C Enable	0: The noise filter for the MTIOCnC pin is disabled. 1: The noise filter for the MTIOCnC pin is enabled.	R/W *1
b3	NFDEN	Noise Filter D Enable	0: The noise filter for the MTIOCnD pin is disabled. 1: The noise filter for the MTIOCnD pin is enabled.	R/W *1
b5, b4	NFCS[1:0]	Noise Filter Clock Select	b5 b4 0 0: P1φ/1 0 1: P1φ/8 1 0: P1φ/32 1 1: Clock source for counting	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. These bits are reserved in the NFCR1 and NFCR2 registers. These bits are read as 0 and writing to them has no effect.

The NFCRn register (n = 0 to 4, 6, 7, or 8) specifies the noise filter function of the input capture input pin of the corresponding channel.

NFAEN Bit (Noise Filter A Enable)

This bit disables or enables the noise filter for input from the MTIOCnA pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFBEN Bit (Noise Filter B Enable)

This bit disables or enables the noise filter for input from the MTIOCnB pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFCEN Bit (Noise Filter C Enable)

This bit disables or enables the noise filter for input from the MTIOCnC pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFDEN Bit (Noise Filter D Enable)

This bit disables or enables the noise filter for input from the MTIOCnD pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits set the sampling interval for the noise filters. When setting the NFCS[1:0] bits, wait for two cycles of the selected sampling interval before setting the input-capture function. When the NFCS[1:0] bits are set to 11b, i.e. selecting the external clock as the source to drive counting, wait for two cycles of the external clock before setting the input capture function.

- MTU0.NFCRC

Address(es): MTU0.NFCRC H'E8041299

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	NFCS[1:0]		NFDEN	NFCEN	NFBEN	NFAEN
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NFAEN	Noise Filter A Enable	0: The noise filter for the MTCLKA pin is disabled. 1: The noise filter for the MTCLKA pin is enabled.	R/W
b1	NFBEN	Noise Filter B Enable	0: The noise filter for the MTCLKB pin is disabled. 1: The noise filter for the MTCLKB pin is enabled.	R/W
b2	NFCEN	Noise Filter C Enable	0: The noise filter for the MTCLKC pin is disabled. 1: The noise filter for the MTCLKC pin is enabled.	R/W
b3	NFDEN	Noise Filter D Enable	0: The noise filter for the MTCLKD pin is disabled. 1: The noise filter for the MTCLKD pin is enabled.	R/W
b5, b4	NFCS[1:0]	Noise Filter Clock Select	b5 b4 0 0: P1φ/1 0 1: P1φ/2 1 0: P1φ/8 1 1: P1φ/32	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The NFCRC register is used in common with all channels and specifies the noise filter function of the external clock pins.

NFAEN Bit (Noise Filter A Enable)

This bit disables or enables the noise filter for input from the MTCLKA pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFBEN Bit (Noise Filter B Enable)

This bit disables or enables the noise filter for input from the MTCLKB pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFCEN Bit (Noise Filter C Enable)

This bit disables or enables the noise filter for input from the MTCLKC pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFDEN Bit (Noise Filter D Enable)

This bit disables or enables the noise filter for input from the MTCLKD pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits set the sampling interval for the noise filters. After setting the NFCS[1:0] bits, wait for two cycles of the selected sampling interval to set the input capture function.

10.2.34 Noise Filter Control Register 5 (NFCR5)

Address(es): MTU5.NFCR5 H'E8041A95

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	NFCS[1:0]		—	NFWEN	NFVEN	NFUEN
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NFUEN	Noise Filter U Enable	0: The noise filter for the MTIC5U pin is disabled. 1: The noise filter for the MTIC5U pin is enabled.	R/W
b1	NFVEN	Noise Filter V Enable	0: The noise filter for the MTIC5V pin is disabled. 1: The noise filter for the MTIC5V pin is enabled.	R/W
b2	NFWEN	Noise Filter W Enable	0: The noise filter for the MTIC5W pin is disabled. 1: The noise filter for the MTIC5W pin is enabled.	R/W
b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b5, b4	NFCS[1:0]	Noise Filter Clock Select	b5 b4 0 0: P1φ/1 0 1: P1φ/8 1 0: P1φ/32 1 1: Clock source for counting	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NFUEN Bit (Noise Filter U Enable)

This bit disables or enables the noise filter for input from the MTIC5U pin. Since unexpected edges may be internally generated when the value of this bit is changed, select the output compare function for the relevant pin in the timer I/O control register.

NFVEN Bit (Noise Filter V Enable)

This bit disables or enables the noise filter for input from the MTIC5V pin. Since unexpected edges may be internally generated when the value of this bit is changed, select the output compare function for the relevant pin in the timer I/O control register.

NFWEN Bit (Noise Filter W Enable)

This bit disables or enables the noise filter for input from the MTIC5W pin. Since unexpected edges may be internally generated when the value of this bit is changed, select the output compare function for the relevant pin in the timer I/O control register.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits set the sampling interval for the noise filters. When setting the NFCS[1:0] bits, wait for two cycles of the selected sampling interval before setting the input-capture function.

10.2.35 Timer A/D Converter Start Request Control Register (TADCR)

- MTU4.TADCR

Address(es): MTU4.TADCR H'E8041240

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BF[1:0]		—	—	—	—	—	—	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ITB4VE	TCIV4 Interrupt Skipping Link Enable*1*2*3	0: A/D converter start request signal TRG4BN and TCIV4 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG4BN and TCIV4 interrupt skipping 1 are linked	R/W
b1	ITB3AE	TGIA3 Interrupt Skipping Link Enable*1*2*3	0: A/D converter start request signal TRG4BN and TGIA3 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG4BN and TGIA3 interrupt skipping 1 are linked	R/W
b2	ITA4VE	TCIV4 Interrupt Skipping Link Enable*1*2*3	0: A/D converter start request signal TRG4AN and TCIV4 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG4AN and TCIV4 interrupt skipping 1 are linked	R/W
b3	ITA3AE	TGIA3 Interrupt Skipping Link Enable*1*2*3	0: A/D converter start request signal TRG4AN and TGIA3 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG4AN and TGIA3 interrupt skipping 1 are linked	R/W
b4	DT4BE	Down-Count TRG4BN Enable*3	0: A/D converter start requests (TRG4BN) disabled during MTU4.TCNT down-count operation 1: A/D converter start requests (TRG4BN) enabled during MTU4.TCNT down-count operation	R/W
b5	UT4BE	Up-Count TRG4BN Enable	0: A/D converter start requests (TRG4BN) disabled during MTU4.TCNT up-count operation 1: A/D converter start requests (TRG4BN) enabled during MTU4.TCNT up-count operation	R/W
b6	DT4AE	Down-Count TRG4AN Enable*3	0: A/D converter start requests (TRG4AN) disabled during MTU4.TCNT down-count operation 1: A/D converter start requests (TRG4AN) enabled during MTU4.TCNT down-count operation	R/W
b7	UT4AE	Up-Count TRG4AN Enable	0: A/D converter start requests (TRG4AN) disabled during MTU4.TCNT up-count operation 1: A/D converter up requests (TRG4AN) enabled during MTU4.TCNT down-count operation	R/W
b13 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15, b14	BF[1:0]	MTU4.TADCOBRA/TADCOBRB Transfer Timing Select	See Table 10.54 for details. These bits specify the transfer timing from MTU4.TADCOBRA and MTU4.TADCOBRB to MTU4.TADCORA and MTU4.TADCORB.	R/W

Note: The TADCR register in MTU4 must not be accessed in eight bits; it should be accessed in 16 bits.
Note 1. Clear the bit to 0 when interrupt skipping is disabled (the T3AEN or T4VEN bit in TITCR1A is cleared to 0 or the T3ACOR or T4VCOR bits in TITCR1A are cleared to 0).
Note 2. If link with interrupt skipping is enabled while interrupt skipping is disabled, A/D converter start requests will not be issued.
Note 3. Clear the bit to 0 when complementary PWM mode is not selected.

TADCR enables or disables A/D converter start requests and specifies whether to link A/D converter start requests with interrupt skipping function. The MTU has one TADCR each for MTU4 and MTU7.

Table 10.54 Setting of Transfer Timing by TADCR.BF[1:0] Bits (MTU4)

Bit 15	Bit 14	Description			
BF[1]	BF[0]	Complementary PWM Mode	Reset-Synchronized PWM Mode	PWM Mode 1	Normal Mode
0	0	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4).
0	1	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) at the crest of the TCNT count in MTU4.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) when a compare match occurs between TCNT and TGRA in MTU3.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) when a compare match occurs between TCNT and TGRA in MTU4.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) when a compare match occurs between TCNT and TGRA in MTU4.
1	0	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) at the trough of the TCNT count in MTU4.	Setting prohibited	Setting prohibited	Setting prohibited
1	1	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) at the crest and trough of the TCNT count in MTU4.	Setting prohibited	Setting prohibited	Setting prohibited

- MTU7.TADCR

Address(es): MTU7, TADCR H'E8041A40

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BF[1:0]		—	—	—	—	—	—	UT7AE	DT7AE	UT7BE	DT7BE	ITA6AE	ITA7VE	ITB6AE	ITB7VE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ITB7VE	TCIV7 Interrupt Skipping Link Enable*1*2*3	0: A/D converter start request signal TRG7BN and TCIV7 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG7BN and TCIV7 interrupt skipping 1 are linked	R/W
b1	ITB6AE	TGIA6 Interrupt Skipping Link Enable*1*2*3	0: A/D converter start request signal TRG7BN and TGIA6 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG7BN and TGIA6 interrupt skipping 1 are linked	R/W
b2	ITA7VE	TCIV7 Interrupt Skipping Link Enable*1*2*3	0: A/D converter start request signal TRG7AN and TCIV7 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG7AN and TCIV7 interrupt skipping 1 are linked	R/W
b3	ITA6AE	TGIA6 Interrupt Skipping Link Enable*1*2*3	0: A/D converter start request signal TRG7AN and TGIA6 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG7AN and TGIA6 interrupt skipping 1 are linked	R/W
b4	DT7BE	Down-Count TRG7BN Enable*3	0: A/D converter start requests (TRG7BN) disabled during MTU7.TCNT down-count operation 1: A/D converter start requests (TRG7BN) enabled during MTU7.TCNT down-count operation	R/W
b5	UT7BE	Up-Count TRG7BN Enable	0: A/D converter start requests (TRG7BN) disabled during MTU7.TCNT up-count operation 1: A/D converter start requests (TRG7BN) enabled during MTU7.TCNT up-count operation	R/W
b6	DT7AE	Down-Count TRG7AN Enable*3	0: A/D converter start requests (TRG7AN) disabled during MTU7.TCNT down-count operation 1: A/D converter start requests (TRG7AN) enabled during MTU7.TCNT down-count operation	R/W
b7	UT7AE	Up-Count TRG7AN Enable	0: A/D converter start requests (TRG7AN) disabled during MTU7.TCNT up-count operation 1: A/D converter up requests (TRG7AN) enabled during MTU7.TCNT down-count operation	R/W
b13 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15, b14	BF[1:0]	MTU7.TADCOBRA/TADCOBRB Transfer Timing Select	See Table 10.55 for details. These bits specify the transfer timing from MTU7.TADCOBRA and MTU7.TADCOBRB to MTU7.TADCORA and MTU7.TADCORB.	R/W

Note: The TADCR register in MTU7 must not be accessed in eight bits; it should be accessed in 16 bits.
Note 1. Clear the bit to 0 when interrupt skipping is disabled (the T6AEN or T7VEN bit in TITCR1B is cleared to 0 or the T6ACOR or T7VCOR bits in TITCR1B are cleared to 0).
Note 2. If link with interrupt skipping is enabled while interrupt skipping is disabled, A/D converter start requests will not be issued.
Note 3. Clear the bit to 0 when complementary PWM mode is not selected.

Table 10.55 Setting of Transfer Timing by TADCR.BF[1:0] Bits (MTU7)

Bit 15	Bit 14	Description			
BF[1]	BF[0]	Complementary PWM Mode	Reset-Synchronized PWM Mode	PWM Mode 1	Normal Mode
0	0	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7).
0	1	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) at the crest of the TCNT count in MTU7.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) when a compare match occurs between TCNT and TGRA in MTU6.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) when a compare match occurs between TCNT and TGRA in MTU7.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) when a compare match occurs between TCNT and TGRA in MTU7.
1	0	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) at the trough of the TCNT count in MTU7.	Setting prohibited	Setting prohibited	Setting prohibited
1	1	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) at the crest and trough of the TCNT count in MTU7.	Setting prohibited	Setting prohibited	Setting prohibited

10.2.36 Timer A/D Converter Start Request Cycle Set Registers (TADCORA and TADCORB)

Address(es): MTU4.TADCORA H'E8041244, MTU4.TADCORB H'E8041246, MTU7.TADCORA H'E8041A44, MTU7.TADCORB H'E8041A46

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: TADCORA and TADCORB must not be accessed in eight bits; it should be accessed in 16 bits.

Note 1. When the A/D converter start request delaying function linked with skipping function 1 (for details, see section 10.3.9 (5) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 1) is used, the value of this register should be 0002h to TCDRA setting - 2 in MTU4 and 0002h to TCDRB setting - 2 in MTU7.

Note 2. When interrupt skipping function 2 is used and the difference between the TADCORA value and the TADCORB value is small, the skipping count may not be counted correctly and the A/D converter start request may not be generated with the expected timing in some cases. The TADCORA and TADCORB values should satisfy the following conditions.

(1) When skipping function 2 is specified with the skipping count set to 0
- The difference between the TADCORA and TADCORB values should be equal to or greater than 4.
- The TADCORA compare interval should be equal to or greater than 4 P1φ cycles (the TADCORA update value should be the previous value + 4 or greater, or previous value - 4 or smaller).
- The TADCORB compare interval should be equal to or greater than 4 P1φ cycles (the TADCORB update value should be the previous value + 4 or greater, or previous value - 4 or smaller).

(2) When skipping function 2 is specified with the skipping count set to 1 or greater
- The difference between the TADCORA and TADCORB values should be equal to or greater than 2.
- The TADCORB compare interval should be equal to or greater than 2 P1φ cycles (the TADCORB update value should be the previous value + 2 or greater, or previous value - 2 or smaller).

TADCORA and TADCORB are 16-bit readable/writable registers that issue a corresponding A/D converter start request when the MTUn.TCNT (n = 4, 7) count reaches the value in TADCORA or TADCORB.
MTUn.TADCORA and TADCORB are initialized to FFFFh by a reset.

10.2.37 Timer A/D Converter Start Request Cycle Set Buffer Registers (TADCOBRA and TADCOBRB)

Address(es): MTU4.TADCORA H'E8041248, MTU4.TADCORB H'E804124A, MTU7.TADCORA H'E8041A48, MTU7.TADCORB H'E8041A4A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: TADCOBRA and TADCOBRB must not be accessed in eight bits; it should be accessed in 16 bits.

TADCOBRA and TADCOBRB are 16-bit readable/writable registers whose values are transferred to TADCORA and TADCORB, respectively, when the crest or trough of the MTUn.TCNT count is reached.
TADCOBRA and TADCOBRB are initialized to FFFFh by a reset.

10.2.38 Timer Interrupt Skipping Mode Registers (TITMRA and TITMRB)

Address(es): MTU.TITMRA H'E804123A, MTU.TITMRB H'E8041A3A

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TITM
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TITM	Interrupt Skipping Function Select	Selects one of the two types of interrupt skipping functions. 0: Selects interrupt skipping function 1*1 1: Selects interrupt skipping function 2*2	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. TITCR1A or TITCR1B is used to enable interrupt skipping function 1.
Note 2. TITCR2A or TITCR2B is used to enable interrupt skipping function 2.

TITMRA and TITMRB are used to select either of two skipping functions for the TITMRA and TITMRB registers.

10.2.39 Timer Interrupt Skipping Set Registers 1 (TITCR1A and TITCR1B)

- MTU.TITCR1A

Address(es): MTU.TITMRA H'E8041230

	b7	b6	b5	b4	b3	b2	b1	b0
	T3AEN	T3ACOR[2:0]			T4VEN	T4VCOR[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	T4VCOR[2:0]	TCIV4 Interrupt Skipping Count Setting	These bits specify the TCIV4 interrupt skipping count within the range from 0 to 7.*1 For details, see Table 10.56.	R/W
b3	T4VEN	T4VEN	0: TCIV4 interrupt skipping disabled 1: TCIV4 interrupt skipping enabled	R/W
b6 to b4	T3ACOR[2:0]	TGIA3 Interrupt Skipping Count Setting	These bits specify the TGIA3 interrupt skipping count within the range from 0 to 7. For details, see Table 10.57.	R/W
b7	T3AEN	T3AEN	0: TGIA3 interrupt skipping disabled 1: TGIA3 interrupt skipping enabled	R/W

Note 1. When 0 is specified for the interrupt skipping count, no interrupt skipping will be performed.
Before changing the interrupt skipping count, be sure to clear the TITCR1A.T3AEN and TITCR1A.T4VEN bits to 0 to clear the skipping counter (TITCNT1A).

- MTU.TITCR1B

Address(es): MTU.TITCR1B H'E8041A30

	b7	b6	b5	b4	b3	b2	b1	b0
	T6AEN	T6ACOR[2:0]			T7VEN	T7VCOR[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	T7VCOR[2:0]	TCIV7 Interrupt Skipping Count Setting	These bits specify the TCIV7 interrupt skipping count within the range from 0 to 7.*1 For details, see Table 10.58.	R/W
b3	T7VEN	T7VEN	0: TCIV7 interrupt skipping disabled 1: TCIV7 interrupt skipping enabled	R/W
b6 to b4	T6ACOR[2:0]	TGIA6 Interrupt Skipping Count Setting	These bits specify the TGIA6 interrupt skipping count within the range from 0 to 7.*1 For details, see Table 10.59.	R/W
b7	T6AEN	T6AEN	0: TGIA6 interrupt skipping disabled 1: TGIA6 interrupt skipping enabled	R/W

Note 1. When 0 is specified for the interrupt skipping count, no interrupt skipping will be performed.
Before changing the interrupt skipping count, be sure to clear the TITCR1B.T6AEN and TITCR1B.T7VEN bits to 0 to clear the skipping counter (TITCNT1B).

TITCR1A and TITCR1B enable or disable interrupt skipping and specify the interrupt skipping count.
The setting in the TITCR1A (TITCR1B) register is valid only while the TITM bit in TITMRA (TITMRB) is set to 0; when the TITM bit in TITMRA (TITMRB) is set to 1, the setting in TITCR1A (TITCR1B) is cleared.

Table 10.56 Setting of Interrupt Skipping Count by T4VCOR[2:0] Bits

Bit 2 T4VCOR2	Bit 1 T4VCOR1	Bit 0 T4VCOR0	Description
0	0	0	Does not skip TCIV4 interrupts.
		1	Sets the TCIV4 interrupt skipping count to 1.
	1	0	Sets the TCIV4 interrupt skipping count to 2.
		1	Sets the TCIV4 interrupt skipping count to 3.
1	0	0	Sets the TCIV4 interrupt skipping count to 4.
		1	Sets the TCIV4 interrupt skipping count to 5.
	1	0	Sets the TCIV4 interrupt skipping count to 6.
		1	Sets the TCIV4 interrupt skipping count to 7.

Table 10.57 Setting of Interrupt Skipping Count by T3ACOR[2:0] Bits

Bit 6 T3ACOR2	Bit 5 T3ACOR1	Bit 4 T3ACOR0	Description
0	0	0	Does not skip TGIA3 interrupts.
		1	Sets the TGIA3 interrupt skipping count to 1.
	1	0	Sets the TGIA3 interrupt skipping count to 2.
		1	Sets the TGIA3 interrupt skipping count to 3.
1	0	0	Sets the TGIA3 interrupt skipping count to 4.
		1	Sets the TGIA3 interrupt skipping count to 5.
	1	0	Sets the TGIA3 interrupt skipping count to 6.
		1	Sets the TGIA3 interrupt skipping count to 7.

Table 10.58 Setting of Interrupt Skipping Count by T7VCOR[2:0] Bits

Bit 2 T7VCOR2	Bit 1 T7VCOR1	Bit 0 T7VCOR0	Description
0	0	0	Does not skip TCIV7 interrupts.
		1	Sets the TCIV7 interrupt skipping count to 1.
	1	0	Sets the TCIV7 interrupt skipping count to 2.
		1	Sets the TCIV7 interrupt skipping count to 3.
1	0	0	Sets the TCIV7 interrupt skipping count to 4.
		1	Sets the TCIV7 interrupt skipping count to 5.
	1	0	Sets the TCIV7 interrupt skipping count to 6.
		1	Sets the TCIV7 interrupt skipping count to 7.

Table 10.59 Setting of Interrupt Skipping Count by T6ACOR[2:0] Bits

Bit 6 T6ACOR2	Bit 5 T6ACOR1	Bit 4 T6ACOR0	Description
0	0	0	Does not skip TGIA6 interrupts.
		1	Sets the TGIA6 interrupt skipping count to 1.
	1	0	Sets the TGIA6 interrupt skipping count to 2.
		1	Sets the TGIA6 interrupt skipping count to 3.
1	0	0	Sets the TGIA6 interrupt skipping count to 4.
		1	Sets the TGIA6 interrupt skipping count to 5.
	1	0	Sets the TGIA6 interrupt skipping count to 6.
		1	Sets the TGIA6 interrupt skipping count to 7.

10.2.40 Timer Interrupt Skipping Counters 1 (TITCNT1A and TITCNT1B)

- MTU.TITCNT1A

Address(es): MTU.TITCNT1A H'E8041231

	b7	b6	b5	b4	b3	b2	b1	b0
	—	T3ACNT[2:0]			—	T4VCNT[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	T4VCNT[2:0]	TCIV4 Interrupt Counter	While the T4VEN bit in TITCR1A is set to 1, the count in these bits is incremented every time a TCIV4 interrupt occurs.	R
b3	—	Reserved	This bit is read as 0.	R
b6 to b4	T3ACNT[2:0]	TGIA3 Interrupt Counter	While the T3AEN bit in TITCR1A is set to 1, the count in these bits is incremented every time a TGIA3 interrupt occurs.	R
b7	—	Reserved	This bit is read as 0.	R

Note 1. To clear the TITCNT1A, clear the TITCR1A.T3AEN and TITCR1A.T4VEN bits to 0.

TITCNT1A and TITCNT1B are 8-bit readable/writable counters. TITCNTA and TITCNTB retain their values even after stopping the count operation of MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT).

T4VCNT[2:0] Bits (TCIV4 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRA is 1
- When the T4VEN bit in TITCR1A is cleared to 0
- When the T4VCOR[2:0] bits in TITCR1A are cleared to 000b
- When the T4VCNT[2:0] bits in TITCNT1A match the T4VCOR[2:0] bits in TITCR1A

T3ACNT[2:0] Bits (TGIA3 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRA is 1
- When the T3AEN bit in TITCR1A is cleared to 0
- When the T3ACOR[2:0] bits in TITCR1A are cleared to 000b
- When the T3ACNT[2:0] bits in TITCNT1A match the T3ACOR[2:0] bits in TITCR1A

- MTU.TITCNT1B

Address(es): MTU.TITCNT1B H'E8041A31

	b7	b6	b5	b4	b3	b2	b1	b0
	—	T6ACNT[2:0]			—	T7VCNT[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	T7VCNT[2:0]	TCIV7 Interrupt Counter	While the T7VEN bit in TITCR1B is set to 1, the count in these bits is incremented every time a TCIV7 interrupt occurs.	R
b3	—	Reserved	This bit is read as 0.	R
b6 to b4	T6ACNT[2:0]	TGIA6 Interrupt Counter	While the T6AEN bit in TITCR1B is set to 1, the count in these bits is incremented every time a TGIA6 interrupt occurs.	R
b7	—	Reserved	This bit is read as 0.	R

Note 1. To clear the TITCNT1B, clear the TITCR1B.T6AEN and TITCR1B.T7VEN bits to 0.

T7VCNT[2:0] Bits (TCIV7 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRB is 1
- When the T7VEN bit in TITCR1B is cleared to 0
- When the T7VCOR[2:0] bits in TITCR1B are cleared to 000b
- When the T7VCNT[2:0] bits in TITCNT1B match the T7VCOR[2:0] bits in TITCR1B

T6ACNT[2:0] Bits (TGIA6 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRB is 1
- When the T6AEN bit in TITCR1B is cleared to 0
- When the T6ACOR[2:0] bits in TITCR1B are cleared to 000b
- When the T6ACNT[2:0] bits in TITCNT1B match the T6ACOR[2:0] bits in TITCR1B

10.2.41 Timer Interrupt Skipping Set Registers 2 (TITCR2A and TITCR2B)

- MTU.TITCR2A

Address(es): MTU.TITCR2A H'E804123B

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG4COR[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TRG4COR[2:0]	TRG4AN/TRG4BN Interrupt Skipping Count Setting	These bits specify the TRG4AN/TRG4BN interrupt skipping count within the range from 0 to 7. For details, see Table 10.60.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TITCR2A and TITCR2B specify the interrupt skipping count for TRG4AN and TRG4BN (TRG7AN and TRG7BN). This setting is valid only while TITMRA or TITMRB is set to 1.

Table 10.60 Setting of Interrupt Skipping Count by TRG4COR[2:0] Bits

Bit 2	Bit 1	Bit 0	
TRG4COR2	TRG4COR1	TRG4COR0	Description
0	0	0	Does not skip TRG4AN and TRG4BN interrupts.
		1	Sets the TRG4AN and TRG4BN interrupt skipping count to 1.
	1	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 2.
		1	Sets the TRG4AN and TRG4BN interrupt skipping count to 3.
1	0	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 4.
		1	Sets the TRG4AN and TRG4BN interrupt skipping count to 5.
	1	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 6.
		1	Sets the TRG4AN and TRG4BN interrupt skipping count to 7.

- MTU.TITCR2B

Address(es): MTU.TITCR2B H'E8041A3B

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG7COR[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TRG7COR[2:0]	TRG7AN/TRG7BN Interrupt Skipping Count Setting	These bits specify the TRG7AN/TRG7BN interrupt skipping count within the range from 0 to 7. For details, see Table 10.61.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Table 10.61 Setting of Interrupt Skipping Count by TRG7COR[2:0] Bits

Bit 2	Bit 1	Bit 0	
TRG7COR2	TRG7COR1	TRG7COR0	Description
0	0	0	Does not skip TRG7AN and TRG7BN interrupts.
		1	Sets the TRG7AN and TRG7BN interrupt skipping count to 1.
	1	0	Sets the TRG7AN and TRG7BN interrupt skipping count to 2.
		1	Sets the TRG7AN and TRG7BN interrupt skipping count to 3.
1	0	0	Sets the TRG7AN and TRG7BN interrupt skipping count to 4.
		1	Sets the TRG7AN and TRG7BN interrupt skipping count to 5.
	1	0	Sets the TRG7AN and TRG7BN interrupt skipping count to 6.
		1	Sets the TRG7AN and TRG7BN interrupt skipping count to 7.

10.2.42 Timer Interrupt Skipping Counters 2 (TITCNT2A and TITCNT2B)

- MTU.TITCNT2A

Address(es): MTU.TITCNT2A H'E804123C

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG4CNT[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TRG4CNT[2:0]	TRG4AN/TRG4BN Interrupt Counter	These bits start counting from the value set in TRG4COR[2:0] and the count decrements every time TRG4AN or TRG4BN is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts become valid.	R
b7 to b3	—	Reserved	These bits are read as 0.	R

TITCNT2A and TITCNT2B start counting from the values set in the TRG4COR[2:0] and TRG7COR[2:0] bits and the count decrements every time TRG4AN or TRG4BN (TITCNT2A) is generated or TRG7AN or TRG7BN (TITCNT2B) is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts or the TRG7AN and TRG7BN interrupts become valid.

TRG4CNT[2:0] Bits (TRG4AN/TRG4BN Interrupt Counter)

These bits start counting from the value set in the TRG4COR[2:0] bits and the count decrements every time a TRG4AN or TRG4BN interrupt is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts become valid.

[Clearing conditions]

- When the TITM bit in TITMRA is 0
- When the TRG4COR[2:0] bits in TITCR2A are cleared to 000b
- When the count of TRG4AN and TRG4BN occurrence matches the TRG4COR[2:0] value in TITCR2A

- MTU.TITCNT2B

Address(es): MTU.TITCNT2B H'E8041A3C

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG7CNT[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TRG7CNT[2:0]	TRG7AN/TRG7BN Interrupt Counter	These bits start counting from the value set in TRG7COR[2:0] and the count decrements every time TRG7AN or TRG7BN is generated. When the count reaches 0 and is reloaded, the TRG7AN and TRG7BN interrupts become valid.	R
b7 to b3	—	Reserved	These bits are read as 0.	R

TRG7CNT[2:0] Bits (TRG7AN/TRG7BN Interrupt Counter)

These bits start counting from the value set in the TRG7COR[2:0] bits and the count decrements every time a TRG7AN or TRG7BN interrupt is generated. When the count reaches 0 and is reloaded, the TRG7AN and TRG7BN interrupts become valid.

[Clearing conditions]

- When the TITM bit in TITMRB is 0
- When the TRG7COR[2:0] bits in TITCR2B are cleared to 000b
- When the count of TRG7AN and TRG7BN occurrence matches the TRG7COR[2:0] value in TITCR2B

10.2.43 Bus Master Interface

The timer counter (MTU8.TCNT), general registers (MTU8.TGRn) for MTU8, and MTU1.TCNTLW, MTU1.TGRALW, and MTU1.TGRBLW registers when TMDR3.LWA = 1 are 32-bit registers. A 32-bit data bus to the bus master enables 32-bit read/write access. 8- and 16-bit read/write are not allowed. Access these registers in 32-bit units.

Excluding MTU8, the timer counters (MTU0.TCNT to MTU7.TCNT), general registers (MTU0.TGRn to MTU7.TGRn), timer subcounters (TCNTSA and TCNTSB), timer cycle buffer registers (TCBRA and TCBRB), timer dead time data registers (TDDRA and TDDRB), timer cycle data registers (TCDRA and TCDRB), timer A/D converter start request control registers (MTU4.TADCR and MTU7.TADCR), timer A/D converter start request cycle set registers (MTU4.TADCORA, MTU4.TADCORB, MTU7.TADCORA, and MTU7.TADCORB), and timer A/D converter start request cycle set buffer registers (MTU4.TADCOBRA, MTU4.TADCOBRB, MTU7.TADCOBRA, and MTU7.TADCOBRB) are 16-bit registers. A 16-bit data bus to the bus master enables 16-bit read/write access. 8-bit read/write is not allowed. Access the registers in 16-bit units.

All registers other than the above registers are 8-bit registers. Read from/write to these registers in 8-bit units.

10.3 Operation

10.3.1 Basic Functions

Each channel has TCNT and TGR. TCNT in each channel performs up-counting and is also capable of free-running count, periodic count, and external event count operations.
Each TGR can be used as an input capture register or an output compare register.

(1) Counter Operation

When one of bits CST0 to CST4 and CST8 in TSTRA, bits CST6 and CST7 in TSTRB, and bits CSTU5, CSTV5, and CSTW5 in MTU5.TSTR is set to 1, TCNT for the corresponding channel begins counting. TCNT can perform various count operations such as free-running count or periodic count operations.

(a) Example of Count Operation Setting Procedure

Figure 10.5 shows an example of the count operation setting procedure.

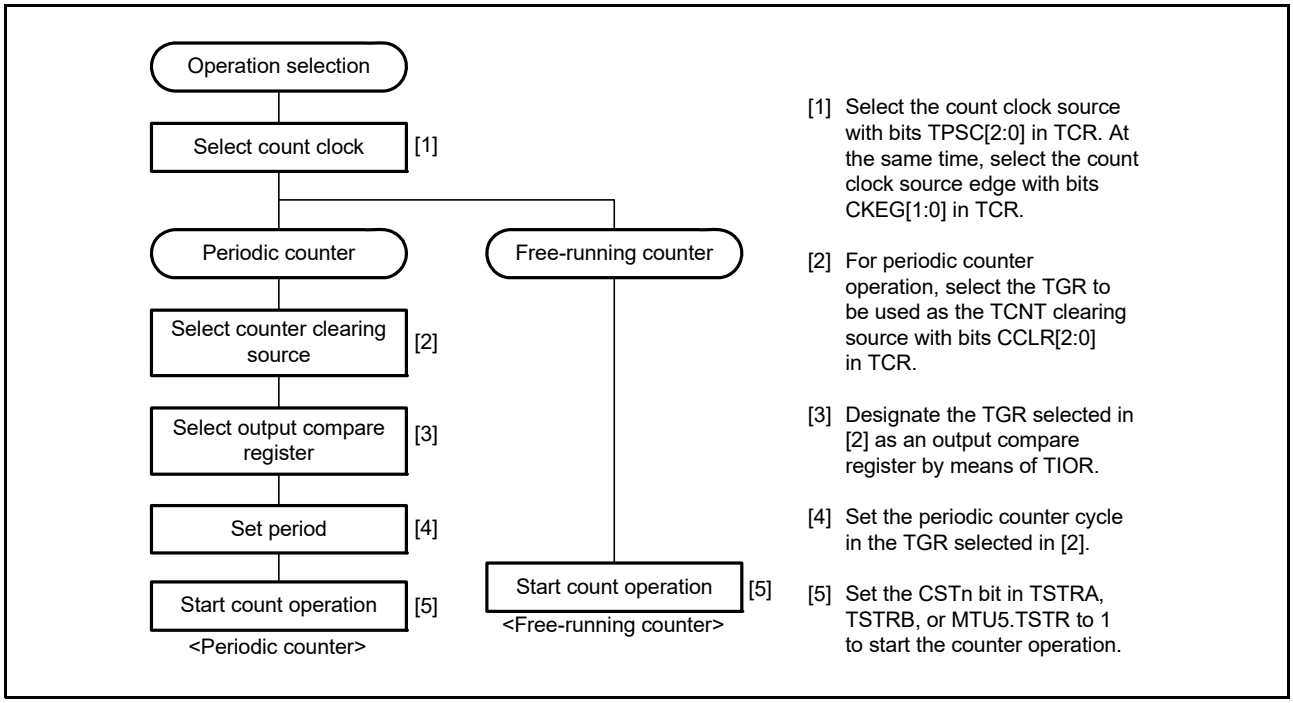


Figure 10.5 Example of Count Operation Setting Procedure

(b) Free-Running Count Operation and Periodic Count Operation

Immediately after a reset, all TCNT counters are designated as free-running counters. When the CSTn bit in TSTRA, TSTRB, or TSTR in MTU5 is set to 1, the corresponding TCNT counter starts up-counting as a free-running counter. When TCNT overflows (from FFFFh to 0000h), if the corresponding TIER.TCIEV bit is 1, an interrupt request is issued to the CPU. After an overflow, TCNT starts counting up again from 0000h.
Figure 10.6 illustrates free-running counter operation.

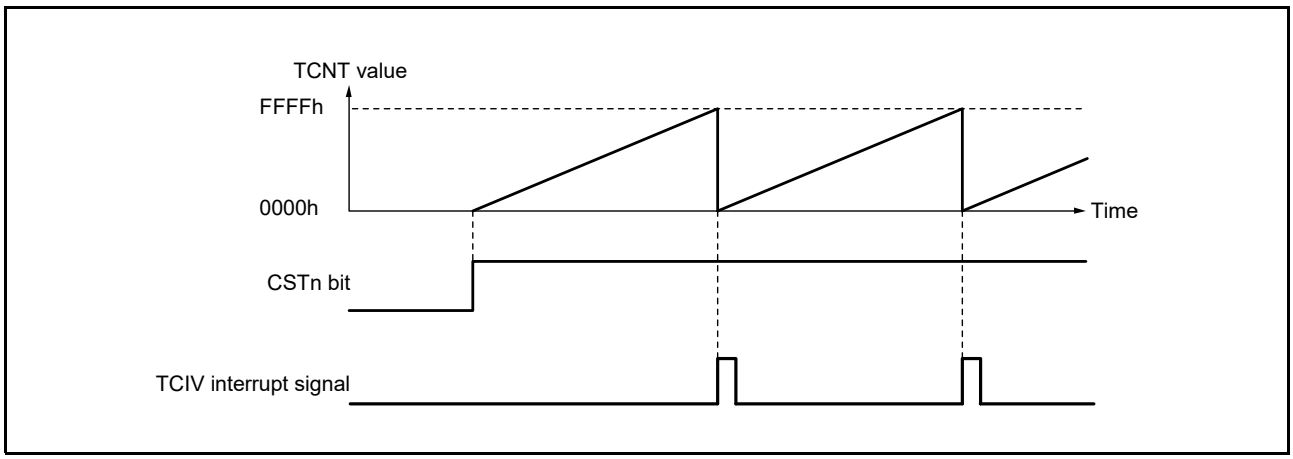


Figure 10.6 Free-Running Counter Operation

When compare match is selected as the TCNT clearing source, TCNT for the relevant channel performs periodic count operation. TGR for setting the cycle is designated as an output compare register, and counter clearing by compare match is selected by means of bits CCLR[2:0] in TCR. After the settings have been made, TCNT starts up-counting as a periodic counter when the corresponding CSTn bit in TSTRA, TSTRB, or TSTR in MTU5 is set to 1. When the count matches the value in TGR, TCNT is cleared to 0000h.
If the value of the corresponding TIER.TGIE bit is 1 at this point, an interrupt request is issued to the CPU. After a compare match, TCNT starts counting up again from 0000h.
Figure 10.7 illustrates periodic counter operation.

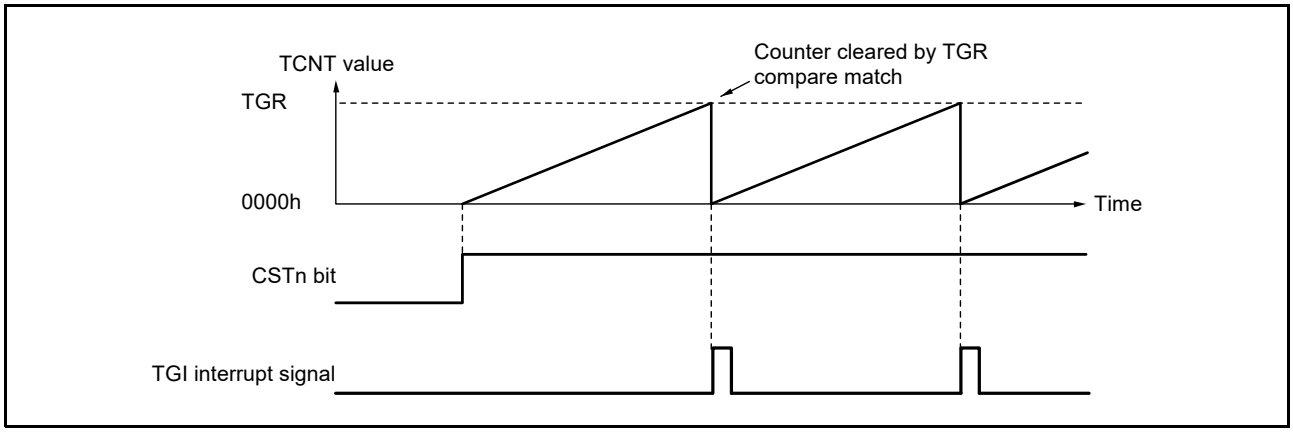


Figure 10.7 Periodic Counter Operation

(2) Waveform Output by Compare Match

Upon compare match, low, high, or toggle output from the corresponding pin can be performed.
The compare match output operation is not available in MTU5.

(a) Example of Procedure for Setting Waveform Output by Compare Match

Figure 10.8 shows an example of the procedure for setting waveform output by compare match

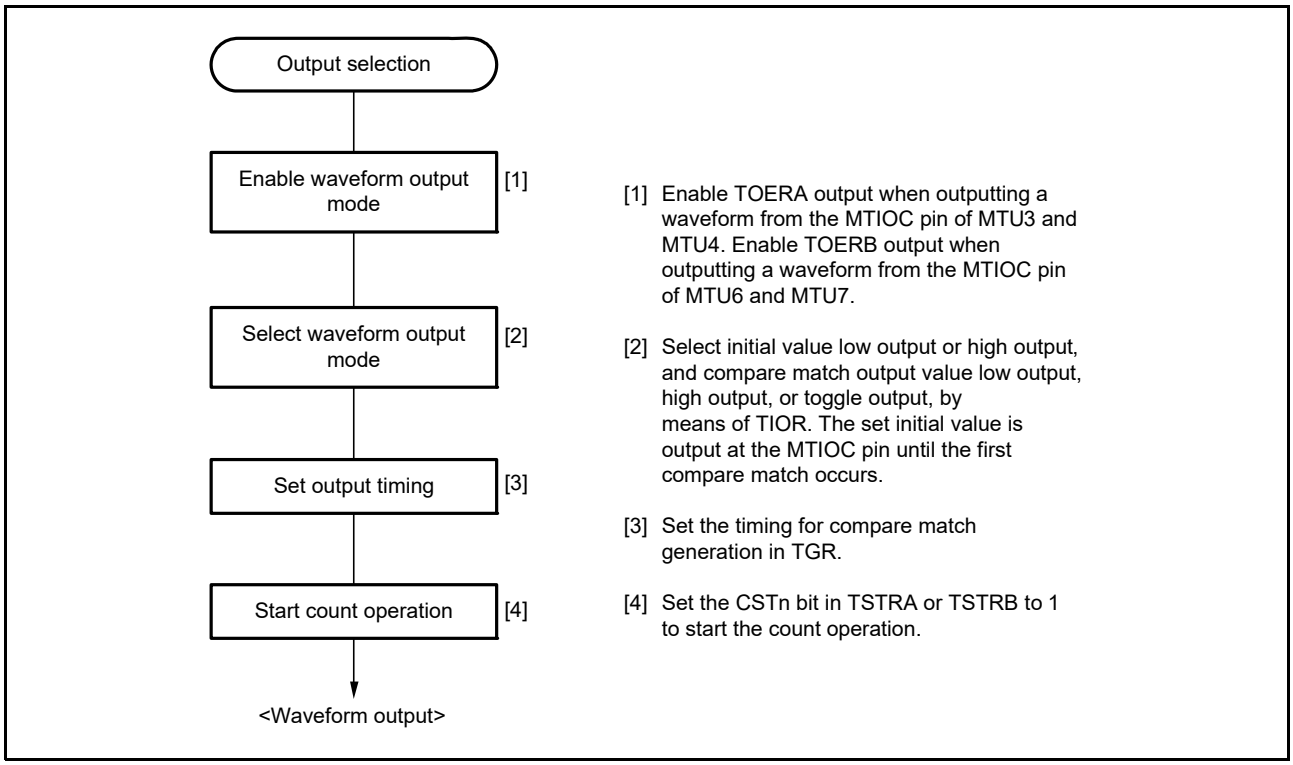


Figure 10.8 Example of Procedure for Setting Waveform Output by Compare Match

(b) Examples of Waveform Output Operation

Figure 10.9 shows an example of low output and high output.
In this example, TCNT has been designated as a free-running counter, and settings have been made so that high is output by compare match A and low is output by compare match B. When the pin level is the same as the specified level, the pin level does not change.

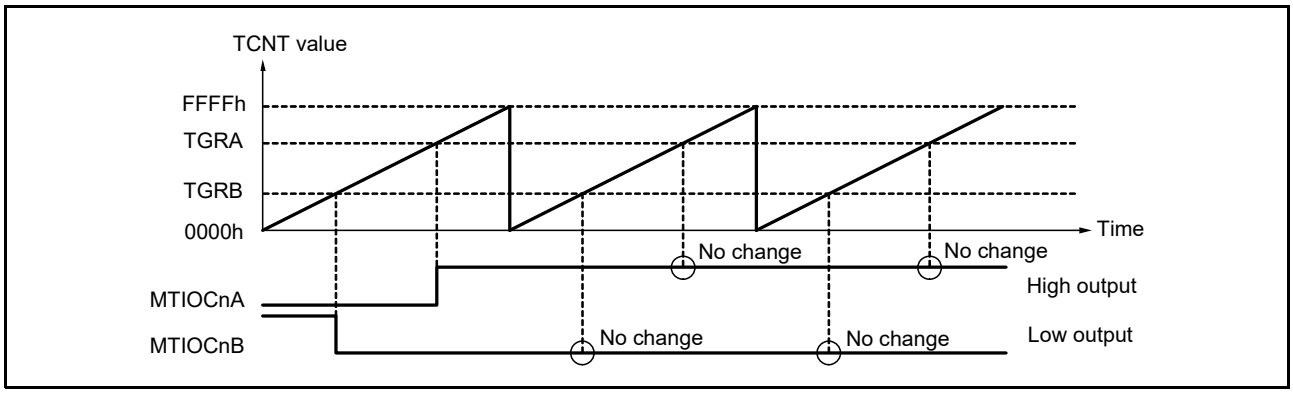


Figure 10.9 Example of low output and high output Operation (n = 0 to 4, 6, 7, 8)

Figure 10.10 shows an example of toggle output.
In this example, TCNT has been designated as a periodic counter (with counter clearing on compare match B), and settings have been made so that the output is toggled by both compare match A and compare match B.

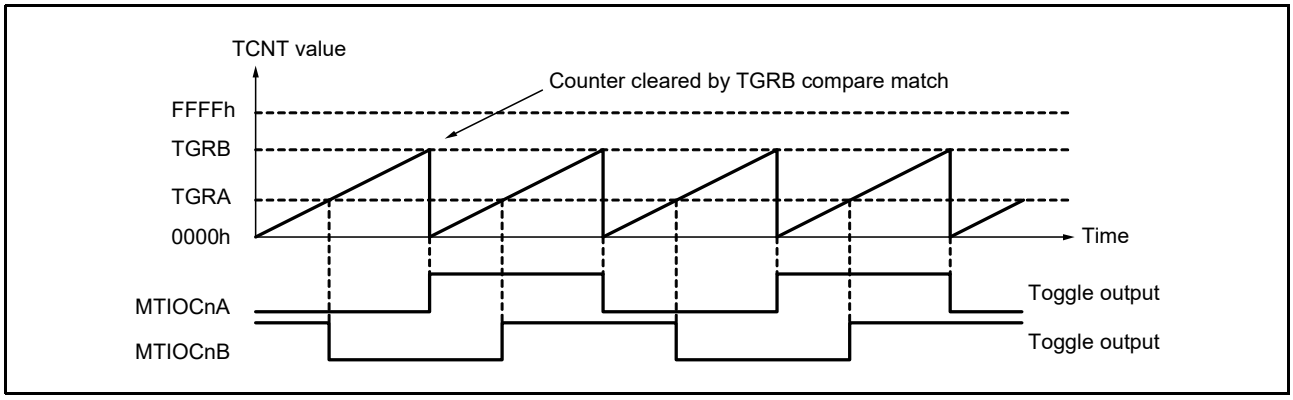


Figure 10.10 Example of Toggle Output Operation (n = 0 to 4, 6, 7, 8)

(3) Input Capture Function

The TCNT value can be transferred to TGR on detection of the MTIOCnm, MTIC5U, MTIC5V, or MTIC5W pin (n = 0 to 4, 6, 7, 8; m = A to D), input edge.
The rising edge, falling edge, or both edges can be selected as the detection edge. For MTU0 and MTU1, another channel's count clock or compare match signal can also be specified as the input capture source.

Note: When another channel's count clock is used as the input capture input for MTU0 and MTU1, P1φ/1 should not be selected as the count clock used for input capture input. Input capture will not be generated if P1φ/1 is selected.

(a) Example of Input Capture Operation Setting Procedure

Figure 10.11 shows an example of the input capture operation setting procedure.

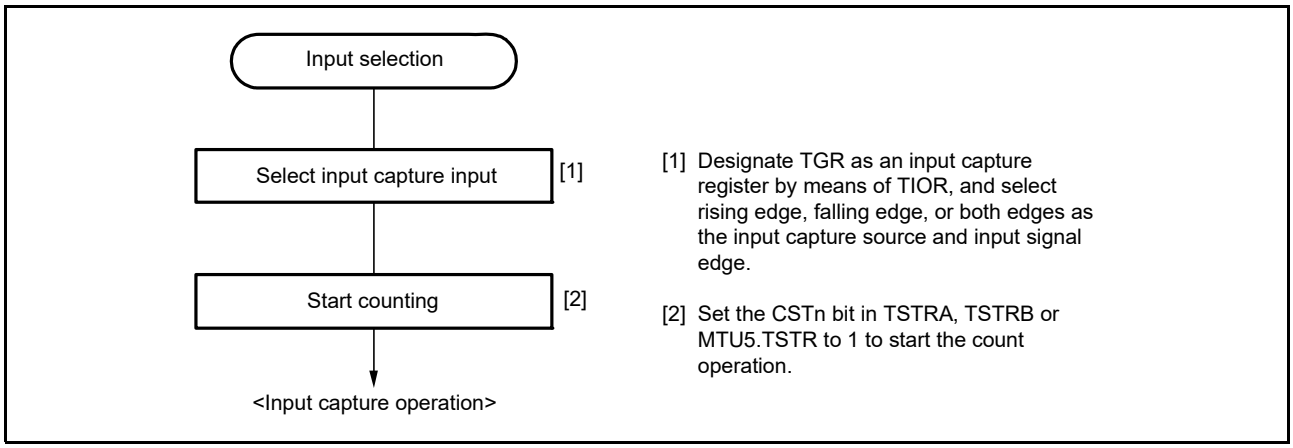


Figure 10.11 Example of Input Capture Operation Setting Procedure

(b) Example of Input Capture Operation

Figure 10.12 shows an example of input capture operation.
In this example, both rising and falling edges have been selected as the MTIOCnA pin input capture input edge, the falling edge has been selected as the MTIOCnB pin input capture input edge, and counter clearing by TGRB input capture has been designated for TCNT. (n = 0 to 4, 6, 7, 8)

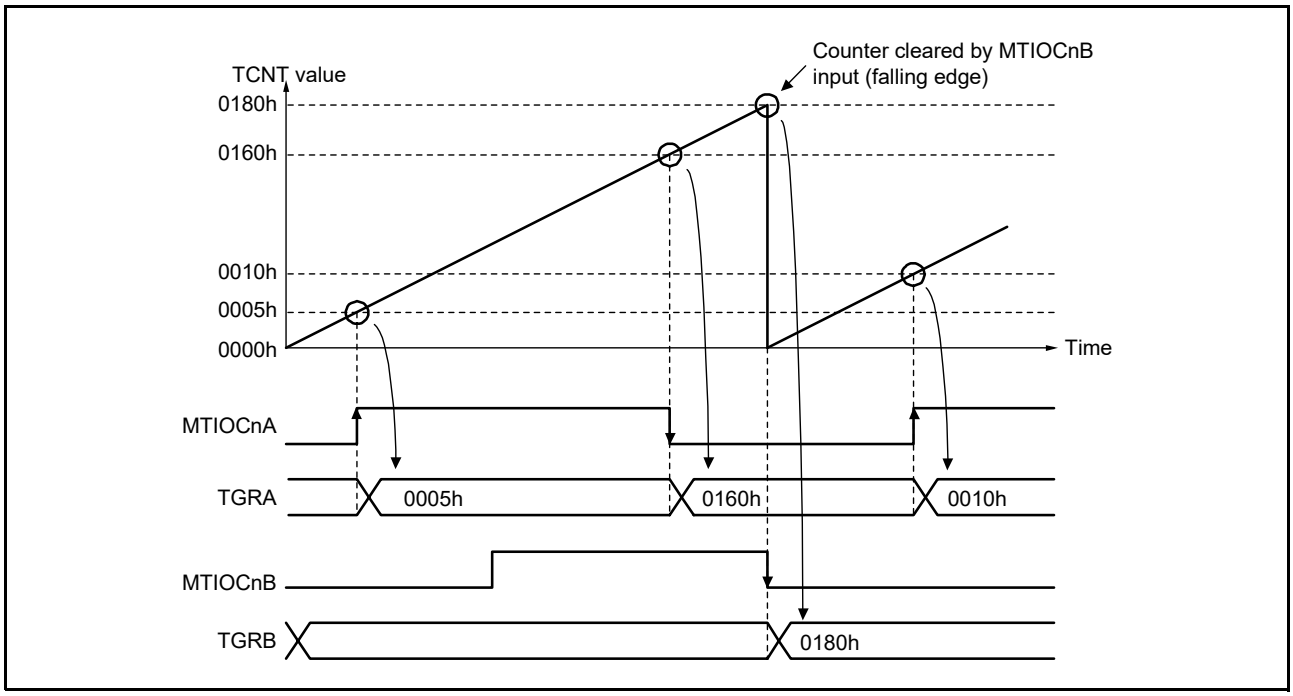


Figure 10.12 Example of Input Capture Operation (n = 0 to 4, 6, 7, 8)

10.3.2 Synchronous Operation

In synchronous operation, the values in multiple TCNT counters can be modified simultaneously (synchronous setting). In addition, multiple TCNT counters can be cleared simultaneously (synchronous clearing) by making the appropriate setting in TCR.
Synchronous operation can increase the number of TGR registers assigned to the same time base.
MTU0 to MTU4, MTU6, and MTU7 can all be designated for synchronous operation. MTU5 and MTU8 cannot be used for synchronous operation.

(1) Example of Synchronous Operation Setting Procedure

Figure 10.13 shows an example of the synchronous operation setting procedure.

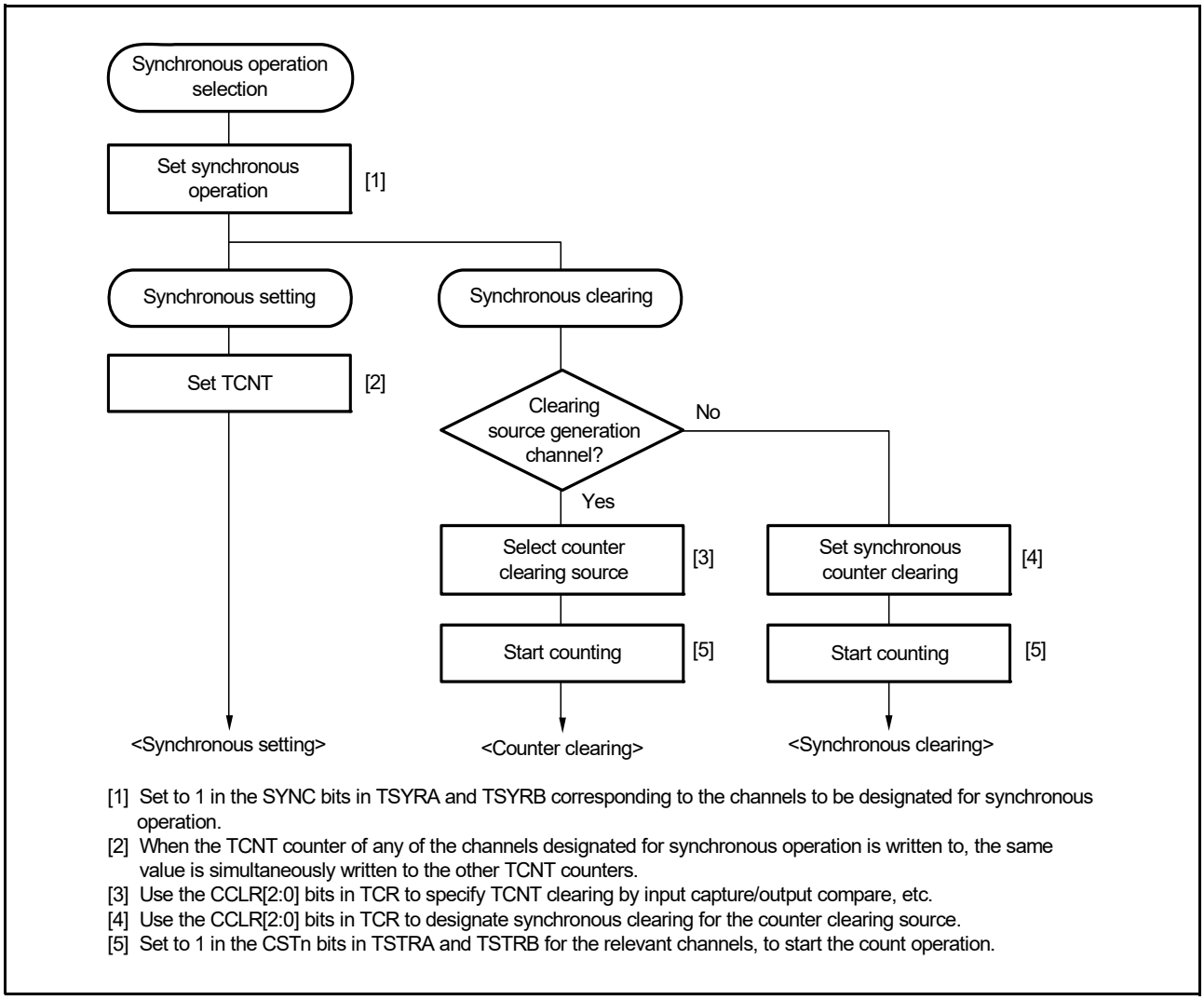


Figure 10.13 Example of Synchronous Operation Setting Procedure

(2) Example of Synchronous Operation

Figure 10.14 shows an example of synchronous operation.
In this example, synchronous operation and PWM mode 1 have been designated for MTU0 to MTU 2, MTU0.TGRB compare match has been set as the counter clearing source in MTU0, and synchronous clearing has been set for the counter clearing source in MTU1 and MTU2.
Three-phase PWM waveforms are output from pins MTIOC0A, MTIOC1A, and MTIOC2A. At this time, synchronous setting and synchronous clearing by MTU0.TGRB compare match are performed for the TCNT counters in MTU0 to MTU2, and the data set in MTU0.TGRB is used as the PWM cycle.
For details of PWM modes, see section 10.3.5, PWM Modes .

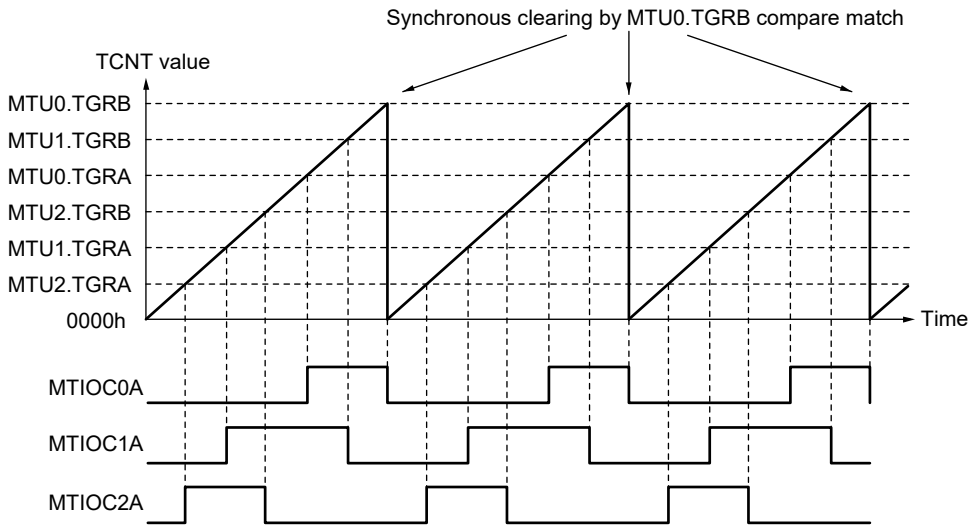


Figure 10.14 Example of Synchronous Operation

10.3.3 Buffer Operation

Buffer operation, provided for MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8, enables TGRC and TGRD to be used as buffer registers. In MTU0, TGRF can also be used as a buffer register.
Buffer operation differs depending on whether TGR has been designated as an input capture register or as a compare match register.

Note: MTU0.TGRE cannot be designated as an input capture register and can only operate as a compare match register.

Table 10.62 shows the register combinations used in buffer operation.

Table 10.62 Register Combinations in Buffer Operation

Channel	Timer General Register	Buffer Register
MTU0	TGRA	TGRC
	TGRB	TGRD
	TGRE	TGRF
MTU3	TGRA	TGRC
	TGRB	TGRD
MTU4	TGRA	TGRC
	TGRB	TGRD
MTU6	TGRA	TGRC
	TGRB	TGRD
MTU7	TGRA	TGRC
	TGRB	TGRD
MTU8	TGRA	TGRC
	TGRB	TGRD

- When TGR is an output compare register

  When a compare match occurs, the value in the buffer register for the corresponding channel is transferred to the timer general register.

This operation is illustrated in Figure 10.15.

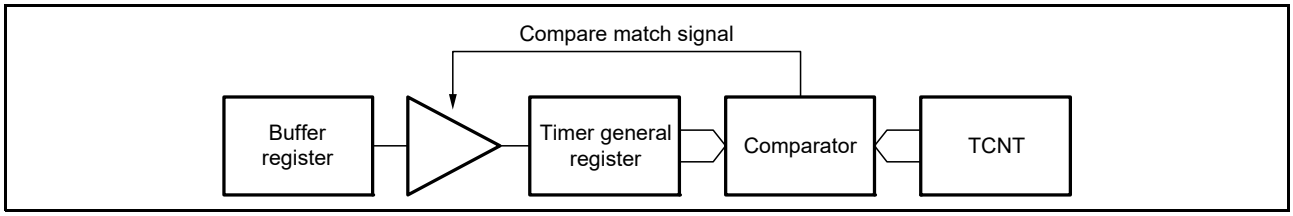


Figure 10.15 Compare Match Buffer Operation

- When TGR is an input capture register

  When an input capture occurs, the value in TCNT is transferred to TGR and the value previously held in TGR is transferred to the buffer register.

This operation is illustrated in Figure 10.16.

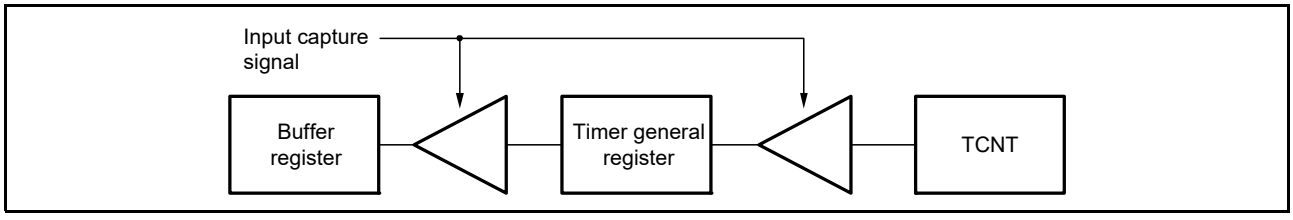


Figure 10.16 Input Capture Buffer Operation

(1) Example of Buffer Operation Setting Procedure

Figure 10.17 shows an example of the buffer operation setting procedure.

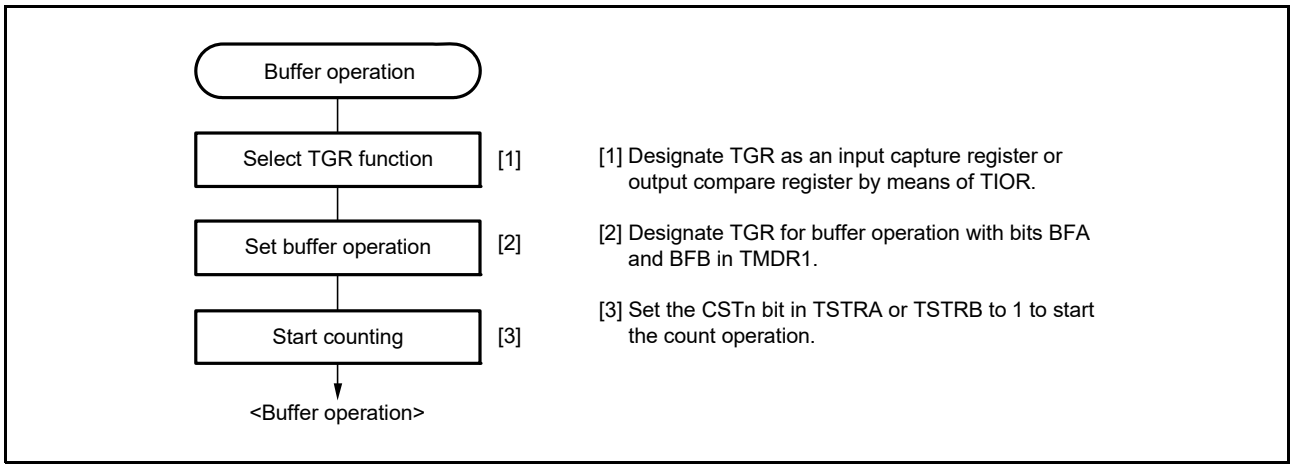


Figure 10.17 Example of Buffer Operation Setting Procedure

(2) Examples of Buffer Operation

(a) When TGR is an Output Compare Register

Figure 10.18 shows an operation example in which PWM mode 1 has been designated for MTU0, and buffer operation has been designated for TGRA and TGRC. The settings used in this example are TCNT clearing by compare match B, high output at compare match A, and low output at compare match B. In this example, the TTSA bit in TBTM is cleared to 0.

As buffer operation has been set, when compare match A occurs, the output changes and the value in buffer register TGRC is simultaneously transferred to timer general register TGRA. This operation is repeated each time compare match A occurs.

For details of PWM modes, see section 10.3.5, PWM Modes .

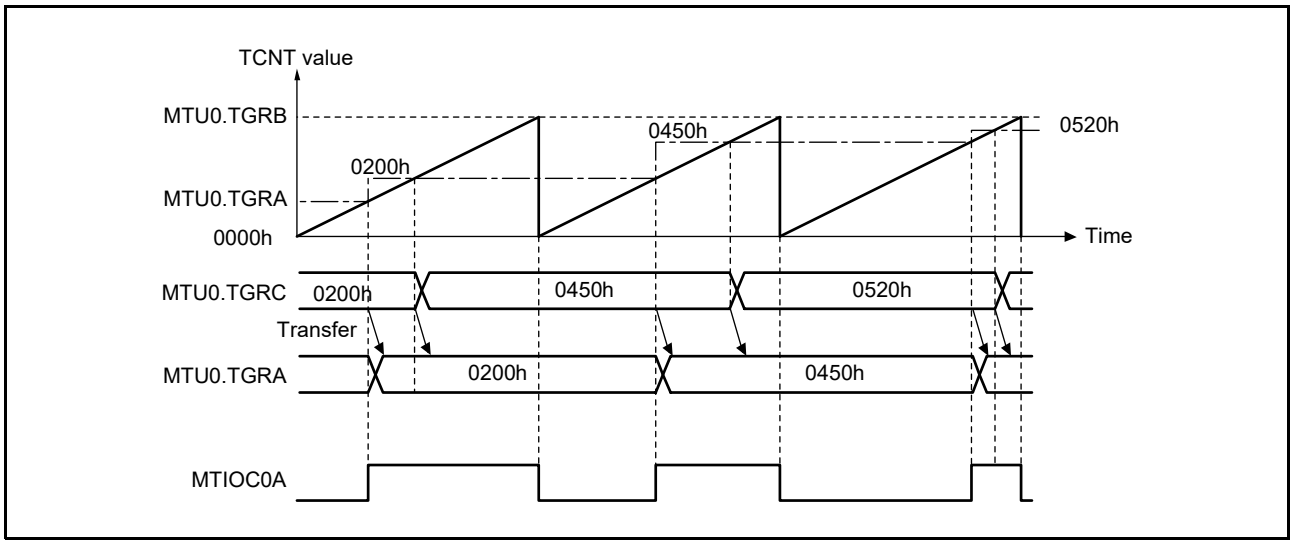


Figure 10.18 Example of Buffer Operation (1)

(b) When TGR is an Input Capture Register

Figure 10.19 shows an operation example in which TGRA has been designated as an input capture register, and buffer operation has been designated for TGRA and TGRC.
Counter clearing by TGRA input capture has been set for TCNT, and both rising and falling edges have been selected as the MTIOCnA pin input capture input edge. (n = 0 to 4, 6, 7, 8)
As buffer operation has been set, when the TCNT value is transferred to TGRA upon occurrence of input capture A, the value previously stored in TGRA is simultaneously transferred to TGRC.

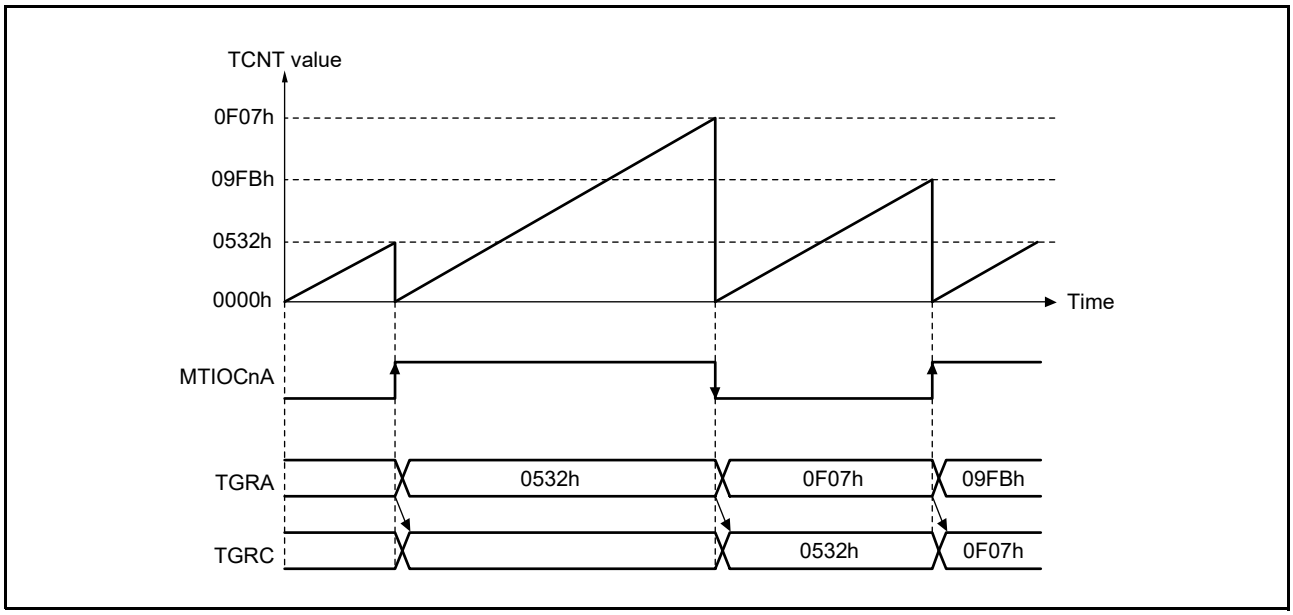


Figure 10.19 Example of Buffer Operation (2) (n = 0 to 4, 6, 7, 8)

(3) Selecting Timing for Transfer from Buffer Registers to Timer General Registers in Buffer Operation

The timing for transfer from buffer registers to timer general registers can be selected in PWM mode 1 or 2 for MTU0 or in PWM mode 1 for MTU3, MTU4, MTU6, and MTU7 by setting the buffer operation transfer mode registers (MTUn.TBTM (n = 0, 3, 4, 6, 7)). Either compare match (value after reset) or TCNT clearing can be selected for the transfer timing. TCNT clearing as transfer timing is one of the following cases.

- When TCNT overflows (FFFFh to 0000h)
- When 0000h is written to TCNT during counting
- When TCNT is cleared to 0000h under the condition specified in the CCLR[2:0] bits in TCR

Note: TBTM must be modified only while TCNT stops.

Figure 10.20 shows an operation example in which PWM mode 1 is designated for MTU0 and buffer operation is designated for MTU0.TGRA and MTU0.TGRC. The settings used in this example are MTU0.TCNT clearing by compare match B, high output at compare match A, and low output at compare match B. The TTSA bit in MTU0.TBTM is set to 1.

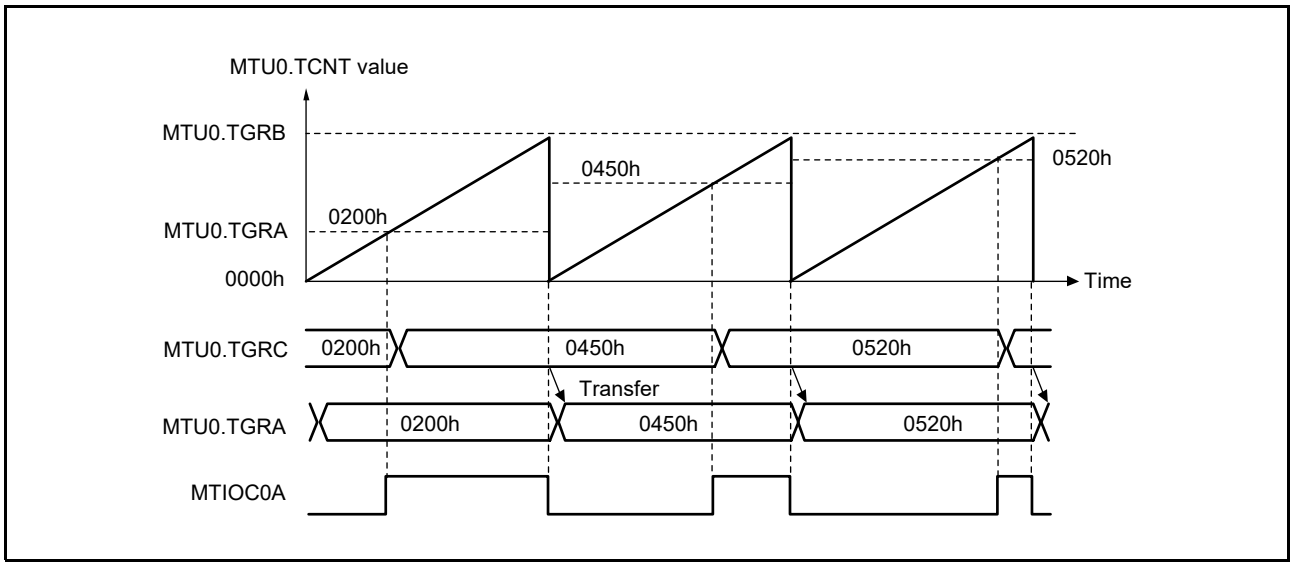


Figure 10.20 Example of Buffer Operation When MTU0.TCNT Clearing is Selected for MTU0.TGRC to MTU0.TGRA Transfer Timing

10.3.4 Cascaded Operation

In cascaded operation, two 16-bit counters in different channels are used together as a 32-bit counter.
There are two functions for connecting MTU1 and MTU2 to use as a 32-bit counter: cascade connection to be set when the MTU1.TMDR3.LWA bit is 0, and cascade connection 32-bit phase counting mode to be set when the MTU1.TMDR3.LWA bit is 1. For details on cascade connection 32-bit phase counting mode, refer to section 10.3.6.2, Cascade Connection 32-Bit Phase Counting Mode. This section describes the cascade connection function to be set when the MTU1.TMDR3.LWA bit is 0.
This function works when the LWA bit of TMDR3 in MTU1 is set to 0 and the TPSC[2:0] bits of TCR in MTU1 are set so that TCNT in MTU1 counts at an overflow/underflow of TCNT in MTU2.
Underflow occurs only when the MTU2 to which the lower 16 bits allocated is in phase counting mode.

Table 10.63 shows the register combinations used in cascaded operation.

Note: When phase counting mode is set for MTU1, the count clock setting is invalid and the counters operate independently in phase counting mode.

Table 10.63 Cascaded Combinations

Combination	Upper 16 Bits	Lower 16 Bits
MTU1 and MTU2	MTU1.TCNT	MTU2.TCNT

For simultaneous input capture of MTU1.TCNT and MTU2.TCNT during cascaded operation, additional input capture input pins can be specified by the input capture control register (TICCR). The input-capture condition is of edges in the signal produced by taking the logical OR of the input level on the main input pin and the input level on the added input pin. Accordingly, if either is at the high level, a change in the level of the other will not produce an edge for detection. For details, see (4) Cascaded Operation Example (c). For input capture in cascade connection, refer to section 10.6.21, Simultaneous Input Capture in MTU1.TCNT and MTU2.TCNT in Cascade Connection.

Table 10.64 shows the TICCR setting and input capture input pins.

Table 10.64 TICCR Setting and Input Capture Input Pins

Target Input Capture	TICCR Setting	Input Capture Input Pin
Input capture from MTU1.TCNT to MTU1.TGRA	I2AE bit = 0 (Initial value)	MTIOC1A
	I2AE bit = 1	MTIOC1A, MTIOC2A
Input capture from MTU1.TCNT to MTU1.TGRB	I2BE bit = 0 (Initial value)	MTIOC1B
	I2BE bit = 1	MTIOC1B, MTIOC2B
Input capture from MTU2.TCNT to MTU2.TGRA	I1AE bit = 0 (Initial value)	MTIOC2A
	I1AE bit = 1	MTIOC2A, MTIOC1A
Input capture from MTU2.TCNT to MTU2.TGRB	I1BE bit = 0 (Initial value)	MTIOC2B
	I1BE bit = 1	MTIOC2B, MTIOC1B

(1) Example of Cascaded Operation Setting Procedure

Figure 10.21 shows an example of the cascaded operation setting procedure.

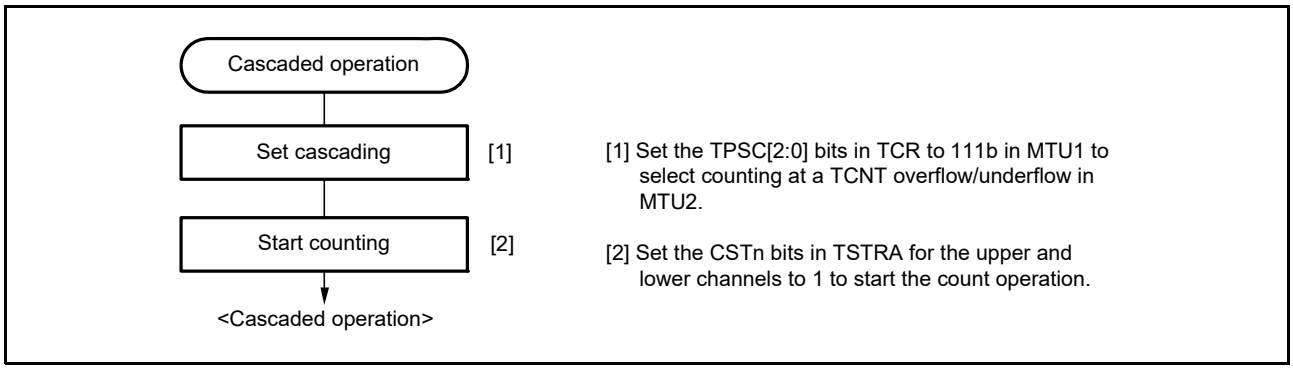


Figure 10.21 Cascaded Operation Setting Procedure

(2) Cascaded Operation Example (a)

Figure 10.22 shows the operation when MTU1.TCNT is set for counting at MTU2.TCNT overflow/underflow and MTU2 is set for phase counting mode 1 while MTU1.TCNT and MTU2.TCNT are cascaded.
MTU1.TCNT is incremented by MTU2.TCNT overflow and decremented by MTU2.TCNT underflow.

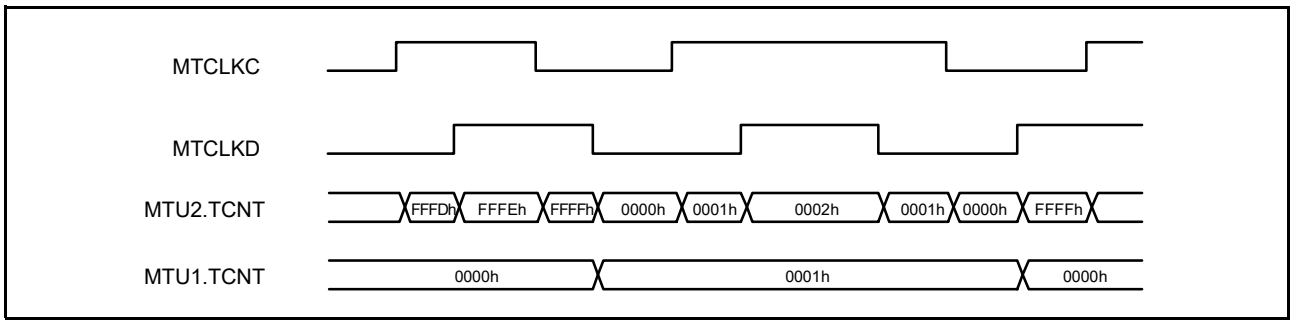


Figure 10.22 Cascaded Operation Example (a)

(3) Cascaded Operation Example (b)

Figure 10.23 illustrates the operation when MTU1.TCNT and MTU2.TCNT have been cascaded and the I2AE bit in TICCR has been set to 1 to include the MTIOC2A pin in the MTU1.TGRA input capture conditions. In this example, the MTU1.TIOR.IOA[3:0] bits have selected the MTIOC1A rising edge for the input capture timing while the MTU2.TIOR.IOA[3:0] bits have selected the MTIOC2A rising edge for the input capture timing.
Under these conditions, the rising edge of both MTIOC1A and MTIOC2A is used for the MTU1.TGRA input capture condition. For the MTU2.TGRA input capture condition, the MTIOC2A rising edge is used.

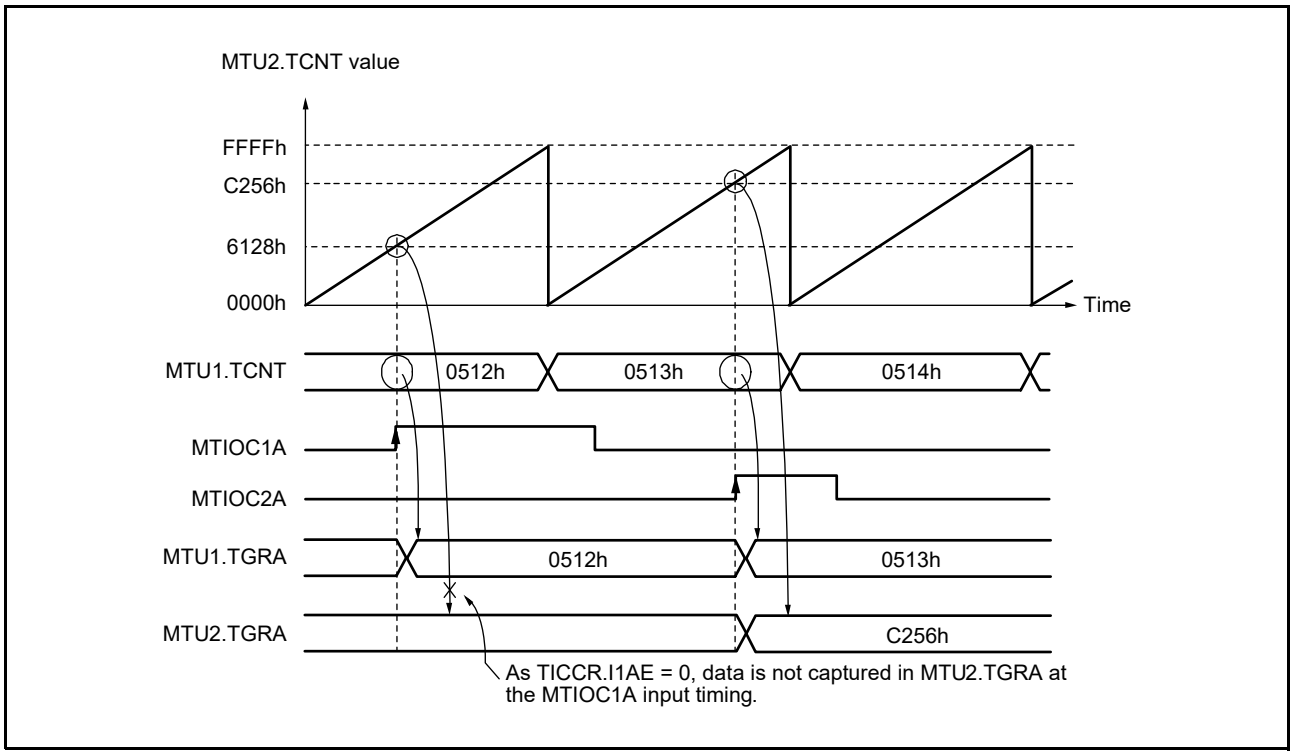
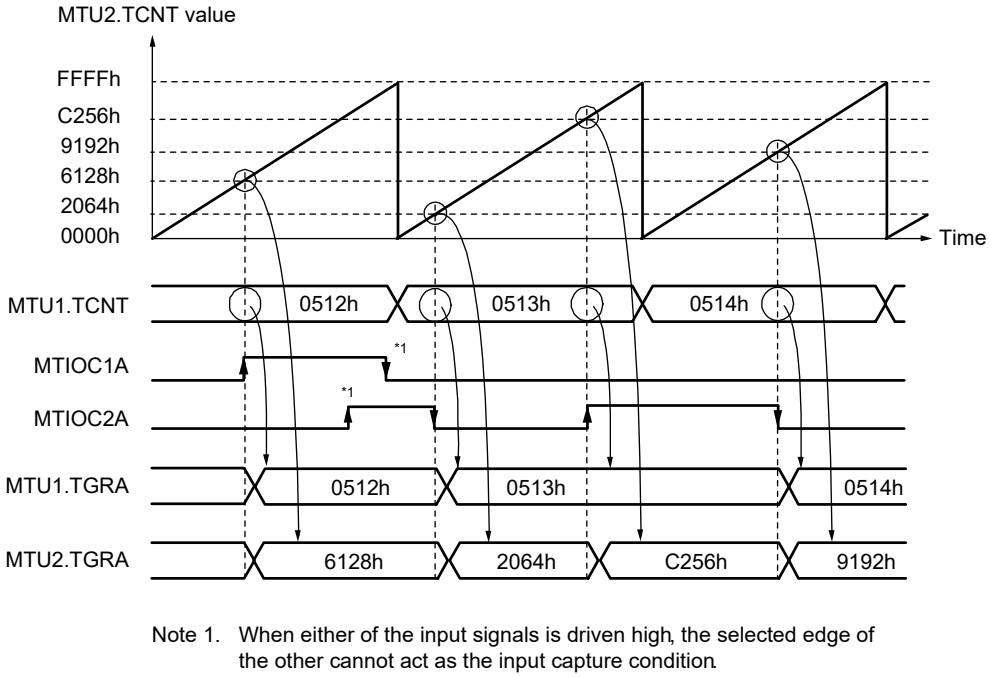


Figure 10.23 Cascaded Operation Example (b)

(4) Cascaded Operation Example (c)

Figure 10.24 illustrates the operation when MTU1.TCNT and MTU2.TCNT have been cascaded and the TICCR.I2AE and I1AE bits have been set to 1 to include the MTIOC2A and MTIOC1A pins in the MTU1.TGRA and MTU2.TGRA input capture conditions, respectively. In this example, the IOA[3:0] bits in both MTU1.TIOR and MTU2.TIOR have selected both the rising and falling edges for the input capture timing. Under these conditions, the ORed result of MTIOC1A and MTIOC2A input is used for the MTU1.TGRA and MTU2.TGRA input capture conditions.



Note 1. When either of the input signals is driven high, the selected edge of the other cannot act as the input capture condition.

Figure 10.24 Cascaded Operation Example (c)

(5) Cascaded Operation Example (d)

Figure 10.25 illustrates the operation when MTU1.TCNT and MTU2.TCNT have been cascaded and the TICCR.I2AE bit has been set to 1 to include the MTIOC2A pin in the MTU1.TGRA input capture conditions. In this example, the IOA[3:0] bits in MTU1.TIOR have selected occurrence of MTU0.TGRA compare match or input capture for the input capture timing while the IOA[3:0] bits in MTU2.TIOR have selected the MTIOC2A rising edge for the input capture timing.

Under these conditions, as MTU1.TIOR has selected occurrence of MTU0.TGRA compare match or input capture for the input capture timing, the MTIOC2A edge is not used for MTU1.TGRA input capture condition although the I2AE bit in TICCR has been set to 1.

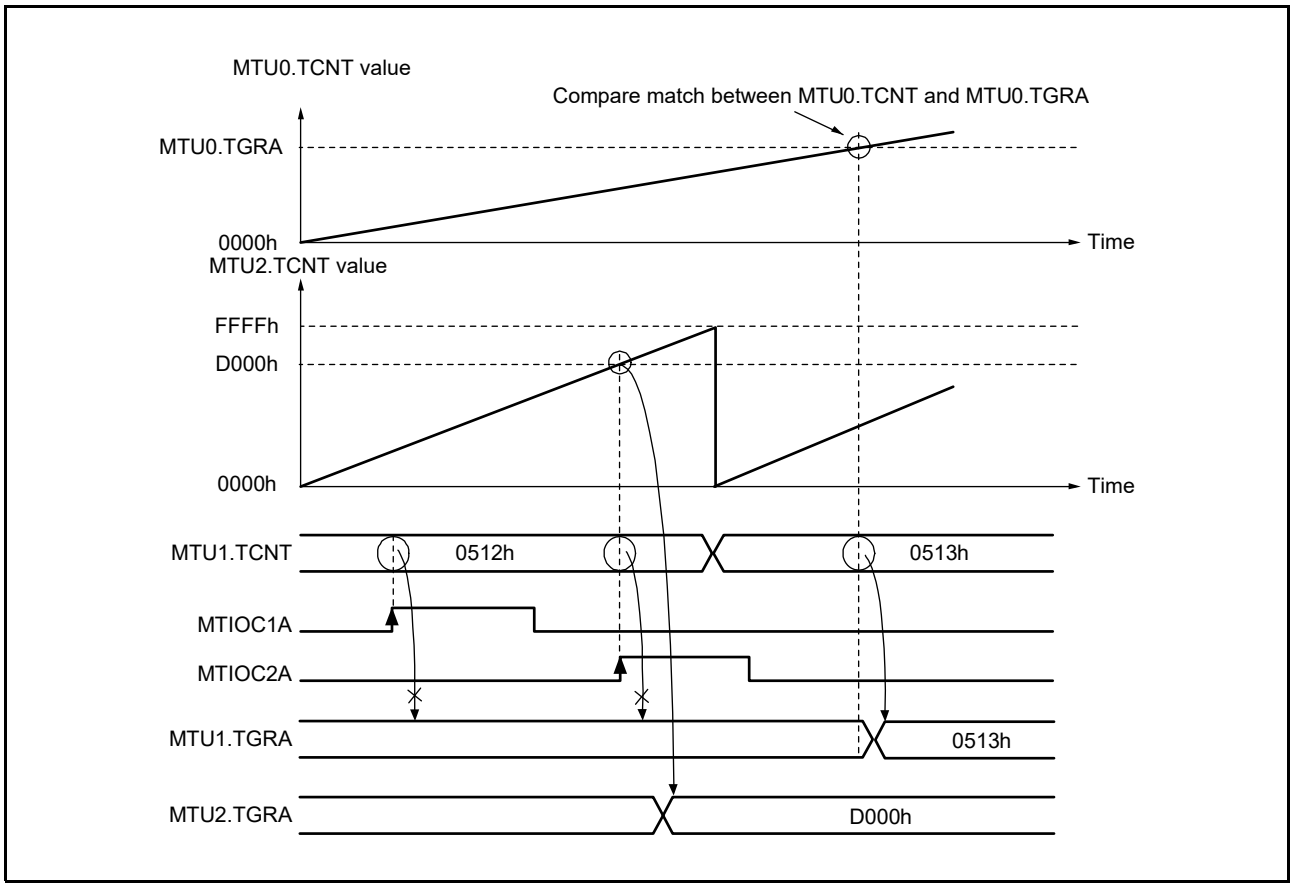


Figure 10.25 Cascaded Operation Example (d)

10.3.5 PWM Modes

PWM modes are provided to output PWM waveforms from the external pins. The output level can be selected as low, high, or toggle output in response to a compare match of each TGR.
PWM waveforms in the range of 0% to 100% duty cycle can be output according to the TGR settings.
By designating TGR compare match as the counter clearing source, the PWM cycle can be specified in that register.
Every channel except MTU5 and MTU8 can be set to PWM mode independently. Synchronous operation is also possible between the channels placed in PWM mode or between a channel in PWM mode and another channel in a different mode.
There are two PWM modes as described below.

(a) PWM Mode 1

PWM waveforms are output from the MTIOCnA and MTIOCnC pins by pairing TGRA with TGRB and TGRC with TGRD. The levels specified by the TIOR.IOA[3:0] and IOC[3:0] bits are output from the MTIOCnA and MTIOCnC pins at compare matches A and C, and the level specified by the TIOR.IOB[3:0] and IOD[3:0] bits are output at compare matches B and D (n = 0 to 4, 6, 7). The initial output value is set in TGRA or TGRC. If the values set in paired TGRs are identical, the output value does not change even when a compare match occurs.

In PWM mode 1, PWM waveforms in up to 12 phases can be output.

(b) PWM Mode 2

PWM waveform output is generated using one TGR as the cycle register and the others as duty registers. The level specified in TIOR is output at compare matches. Upon counter clearing by a cycle register compare match, the initial value set in TIOR is output from each pin. If the values set in the cycle and duty registers are identical, the output value does not change even when a compare match occurs.

In PWM mode 2, up to eight phases of PWM waveforms can be output when synchronous clearing is used as synchronous operation in the channels that cannot be placed in PWM mode 2.

The correspondence between PWM output pins and registers is shown in Table 10.65.

Table 10.65 PWM Output Registers and Output Pins

		Output Pins	
Channel	Register	PWM Mode 1	PWM Mode 2
MTU0	TGRA	MTIOC0A	MTIOC0A
	TGRB		MTIOC0B
	TGRC	MTIOC0C	MTIOC0C
	TGRD		MTIOC0D
MTU1	TGRA	MTIOC1A	MTIOC1A
	TGRB		MTIOC1B
MTU2	TGRA	MTIOC2A	MTIOC2A
	TGRB		MTIOC2B
MTU3	TGRA	MTIOC3A	Setting prohibited
	TGRB		
	TGRC	MTIOC3C	
	TGRD		
MTU4	TGRA	MTIOC4A	
	TGRB		
	TGRC	MTIOC4C	
	TGRD		
MTU6	TGRA	MTIOC6A	
	TGRB		
	TGRC	MTIOC6C	
	TGRD		
MTU7	TGRA	MTIOC7A	
	TGRB		
	TGRC	MTIOC7C	
	TGRD		

Note: In PWM mode 2, PWM waveform output is not possible for the TGR register in which the PWM cycle is set.

(1) Example of PWM Mode Setting Procedure

Figure 10.26 shows an example of the PWM mode setting procedure.

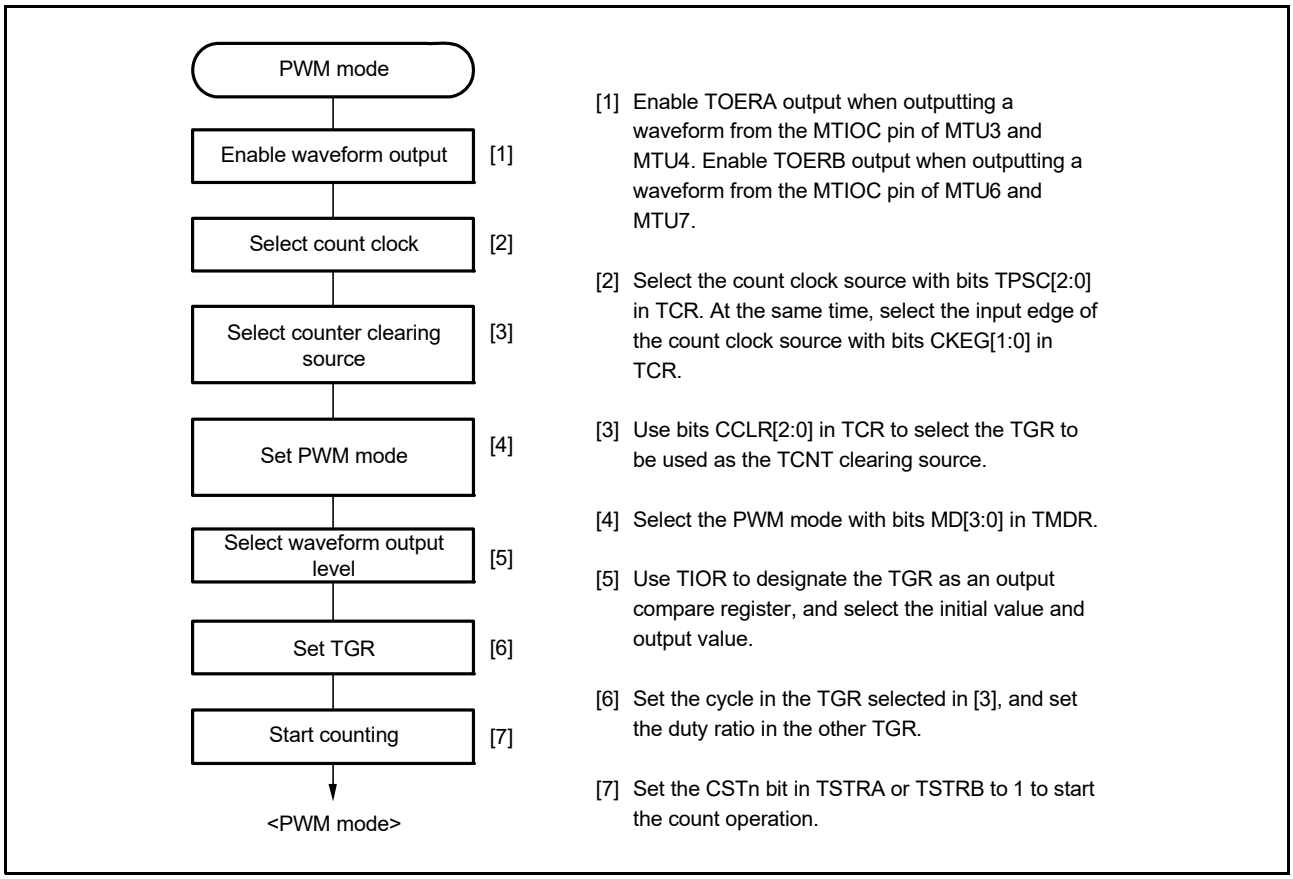


Figure 10.26 Example of PWM Mode Setting Procedure

(2) Examples of PWM Mode Operation

Figure 10.27 shows an example of operation in PWM mode 1.
In this example, TGRA compare match is set as the TCNT clearing source, 0 is set as the initial output value and output value for TGRA, and 1 is set as the output value for TGRB.
In this case, the value set in TGRA is used as the cycle, and the value set in TGRB is used as the duty ratio.

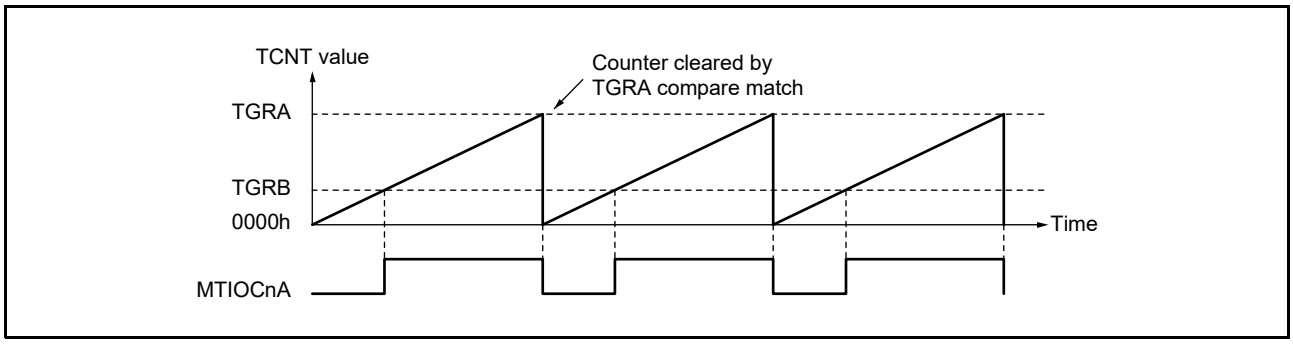


Figure 10.27 Example of PWM Mode 1 Operation (n = 0 to 4, 6, 7)

Figure 10.28 shows an example of operation in PWM mode 2.
In this example, synchronous operation is designated for MTU0 and MTU1, MTU1.TGRB compare match is set as the TCNT clearing source, and low is set as the initial output value and high as the output value for the other TGR registers (MTU0.TGRA to MTU0.TGRD and MTU1.TGRA), outputting 5-phase PWM waveforms.
In this case, the value set in MTU1.TGRB is used as the cycle, and the values set in the other TGRs are used as the duty ratio.

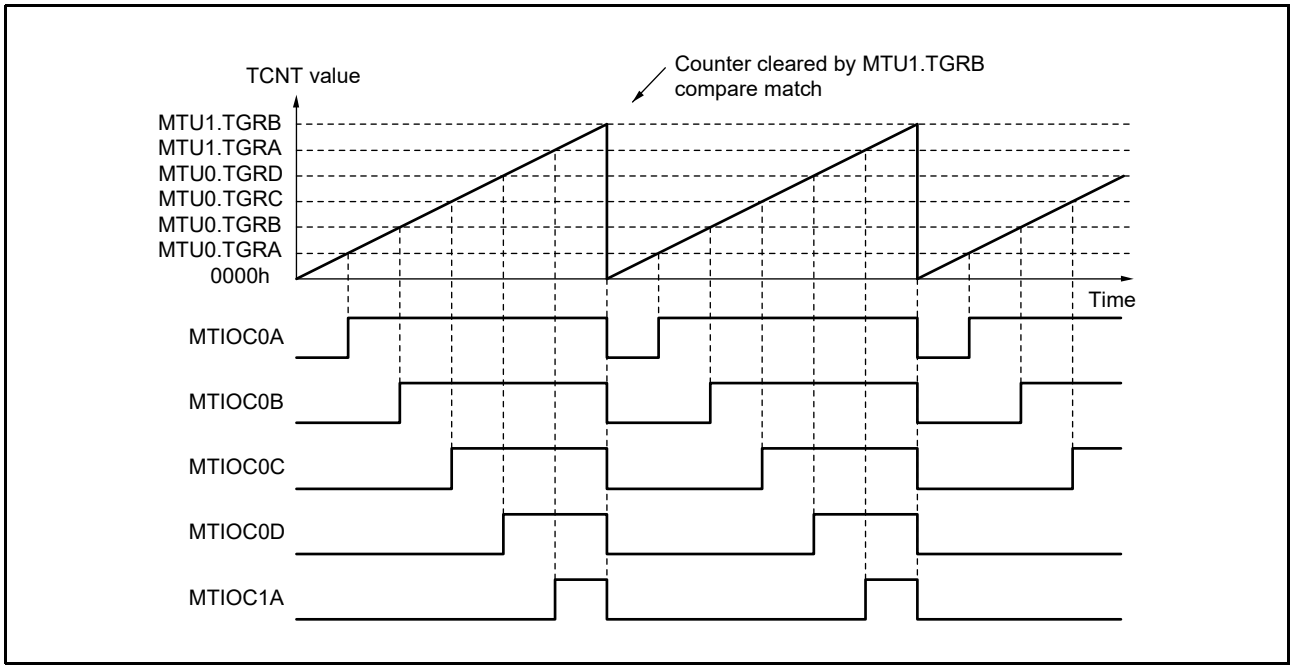


Figure 10.28 Example of PWM Mode 2 Operation

Figure 10.29 shows examples of PWM waveform output with 0% duty and 100% duty in PWM mode 1. In these examples, TGRA compare match is selected as the TCNT clearing source, the initial output value and the output value for TGRA are set to the low level, and the output value for TGRB is set to the high level.

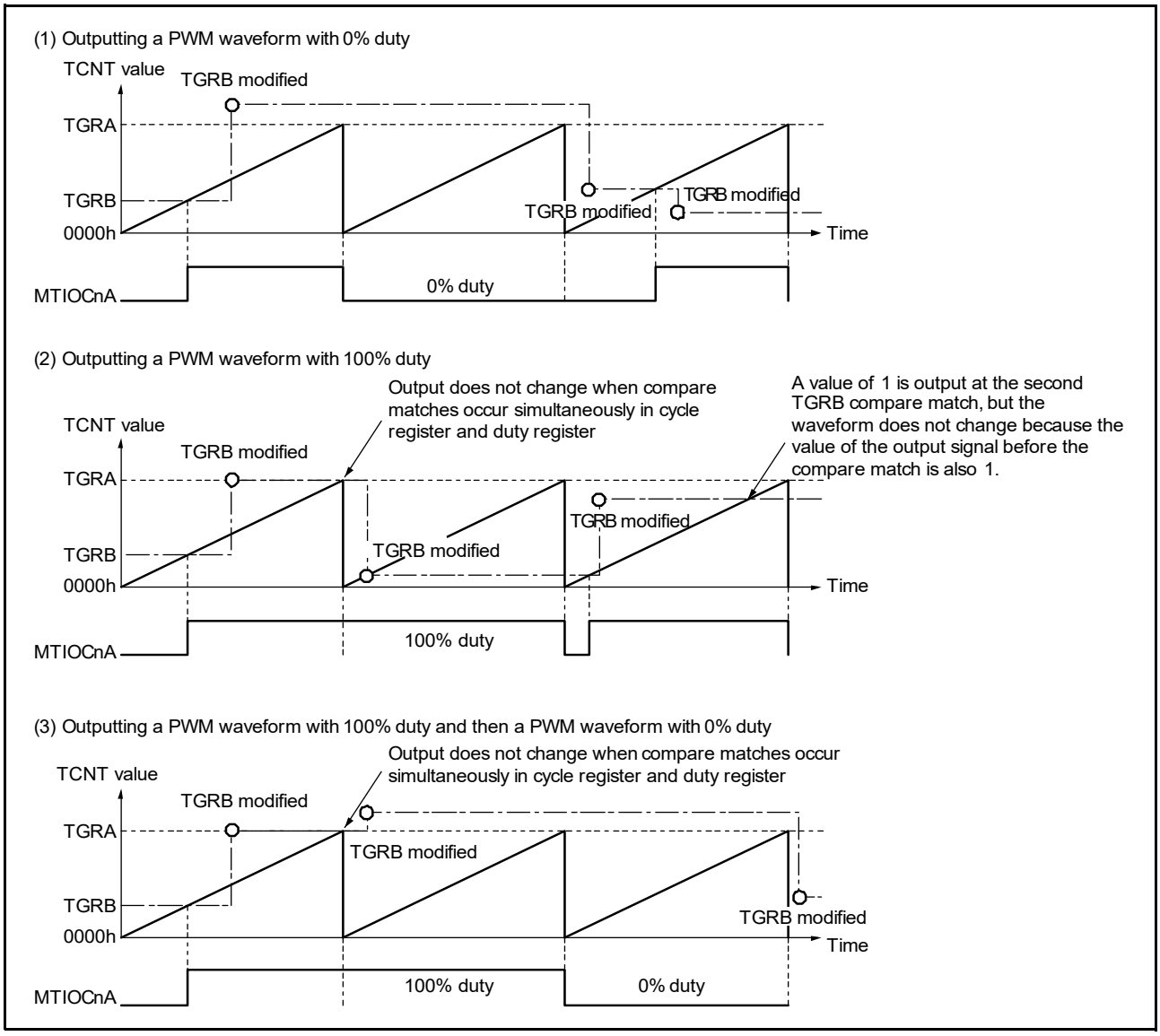


Figure 10.29 Examples of PWM Mode Operation (PWM Waveform Output with 0% Duty and 100% Duty)
(n = 0 to 4, 6, 7)

10.3.6 Phase Counting Mode

There are two phase counting modes: 16-bit phase counting mode in which MTU1 and MTU2 operate independently, and cascade connection 32-bit phase counting mode in which MTU1 and MTU2 are cascaded.

In phase counting mode, the phase difference between two external input clocks is detected and the corresponding TCNT is incremented or decremented.

Two external clock input pins for each phase counting mode are not affected by the settings of TCR.TPSC[2:0], TCR2.TPSC2[2:0], and CKEG[1:0]. The two external clock input pins used in 16-bit phase counting mode and cascade connection 32-bit phase counting mode of MTU2 can be selected by MTU1.TMDR3.PHCKSEL. In a phase counting mode other than 16-bit phase counting mode and cascade connection 32-bit phase counting mode of MTU2, MTCLKA and MTCLKB are selected for A-phase and B-phase, respectively. In phase counting mode, the external clock pins MTCLKA, MTCLKB, MTCLKC, and MTCLKD are used for two-phase encoder pulse input.

Table 10.66 lists the external clock input pins to be connected in each phase counting mode.

Table 10.66 Clock Input Pins in Phase Counting Mode

		External Clock Input Pins	
Phase Counting Mode	TMDR3.PHCKSEL bit	A-Phase	B-Phase
MTU1 16-bit phase counting mode	x (Don't care)	MTCLKA	MTCLKB
MTU2 16-bit phase counting mode	0	MTCLKA	MTCLKB
	1 (initial value)	MTCLKC	MTCLKD
Cascade connection 32-bit phase counting mode	0	MTCLKA	MTCLKB
	1 (initial value)	MTCLKC	MTCLKD

10.3.6.1 16-Bit Phase Counting Mode

When the MTU1.TMDR3.LWA is 0, 16-bit phase counting mode can be set individually for MTU1 and MTU2.
In 16-bit phase counting mode, the phase difference between two external input clocks is detected and the 16-bit counter TCNT of the corresponding channel is incremented or decremented.
When 16-bit phase counting mode is specified, an external clock is selected as the count clock and TCNT operates as an up-counter/down-counter regardless of the setting of TCR.TPSC[2:0], TCR2.TPSC2[2:0], and CKEG[1:0]. However, the functions of TCR.CCLR[1:0],TIOR, TIER, and TGR are enabled, and input capture/compare match and interrupt functions can be used.
These external input pins can be used for two-phase encoder pulse input.
If an overflow occurs during TCNT up-counting, a TCIV interrupt is generated when the corresponding TCIEV bit in the TIER register is set to 1.
If an underflow occurs during TCNT down-counting, a TCIU interrupt is generated when the corresponding TCIEU bit in the TIER register is set to 1.
The TCFD bit in TSR is the count direction flag. Read the TCFD flag to check whether TCNT is counting up or down.

(1) Example of 16-Bit Phase Counting Mode Setting Procedure

Figure 10.30 shows an example of the phase counting mode setting procedure.

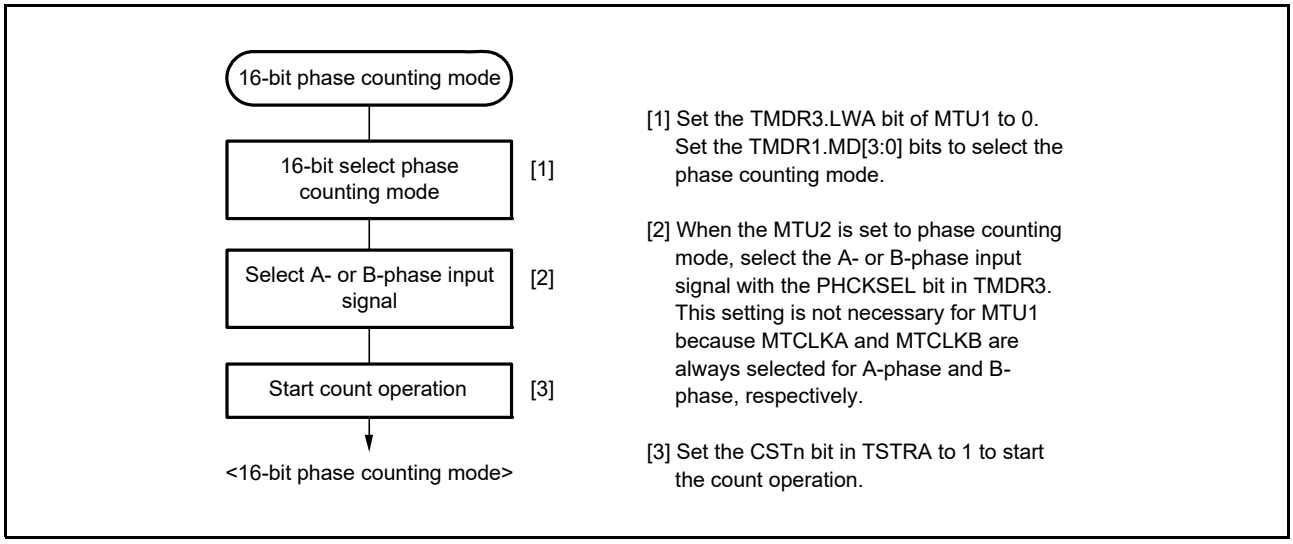


Figure 10.30 Example of 16-Bit Phase Counting Mode Setting Procedure

(2) Examples of 16-Bit Phase Counting Mode Operation

In phase counting mode, TCNT is incremented or decremented according to the phase difference between two external clocks. There are five modes according to the count conditions. Each mode operates under the condition PHCKSEL = 1, which means the phase clock for MTU1 is input from MTCLKA or MTCLKB and that for MTU2 is input from MTCLKC or MTCLKD.

(a) Phase Counting Mode 1

Figure 10.31 shows an example of operation in phase counting mode 1, and Table 10.67 summarizes the TCNT up-counting and down-counting conditions.

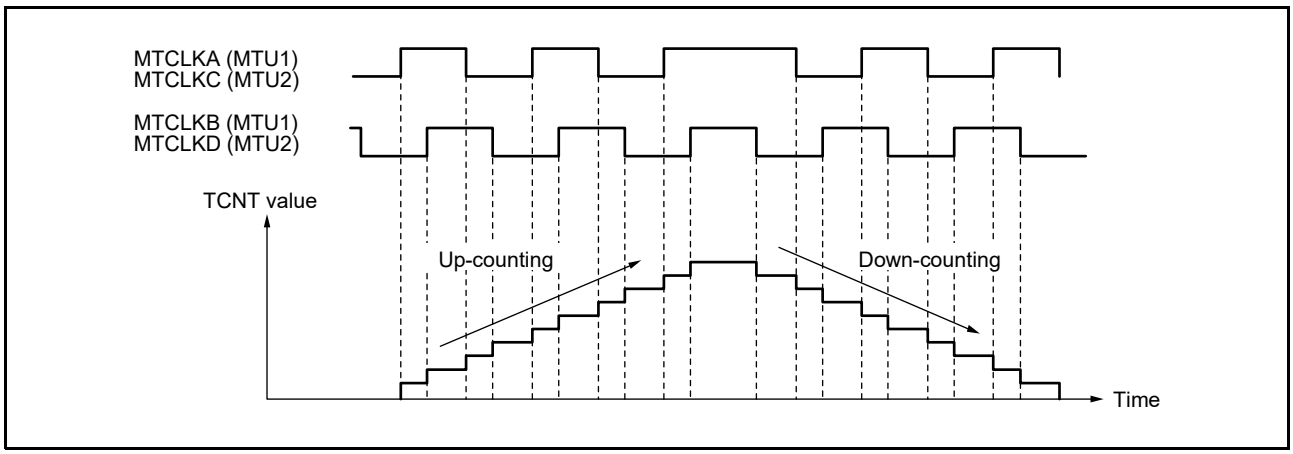


Figure 10.31 Example of Operation in Phase Counting Mode 1

Table 10.67 Up-Counting and Down-Counting Conditions in Phase Counting Mode 1

MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
High	↑	Up-counting
Low	↓	
↑	Low	
↓	High	
High	↓	Down-counting
Low	↑	
↑	High	
↓	Low	

↑ : Rising edge

↓ : Falling edge

(b) Phase Counting Mode 2

Figure 10.32 to Figure 10.34 show the examples of operation in phase counting mode 2 and Table 10.68 summarizes the TCNT up-counting and down-counting conditions.

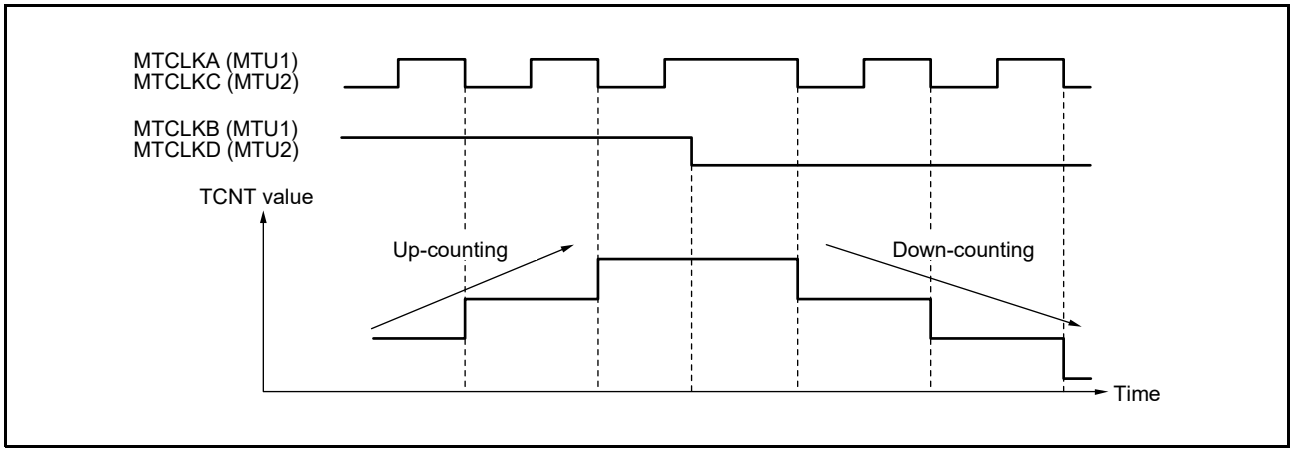


Figure 10.32 Example of Operation in Phase Counting Mode 2 (When MTUn.TCR2.PCB[1:0] is 00b (n = 1, 2))

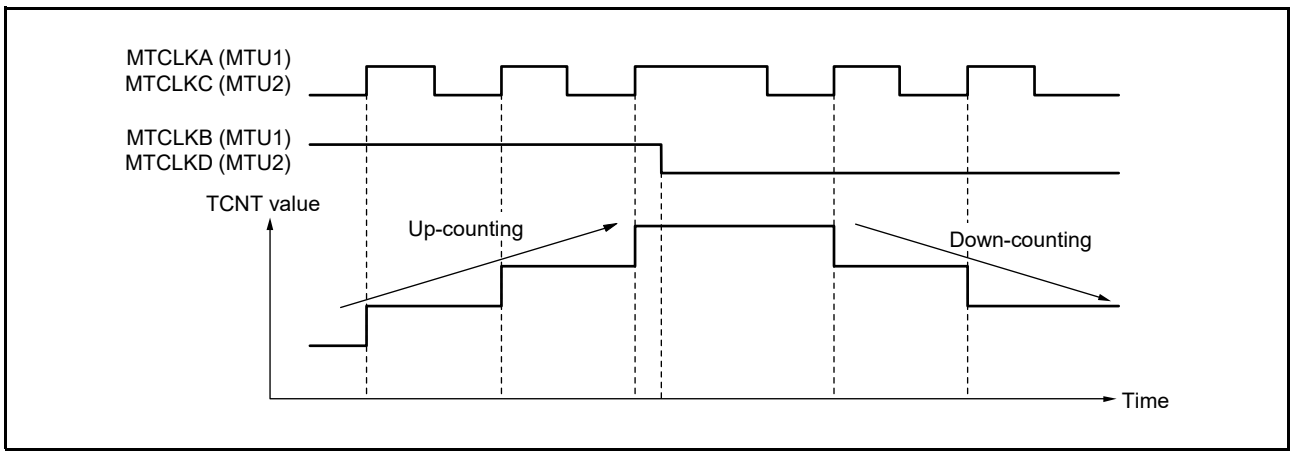


Figure 10.33 Example of Operation in Phase Counting Mode 2 (When MTUn.TCR2.PCB[1:0] is 01b (n = 1, 2))

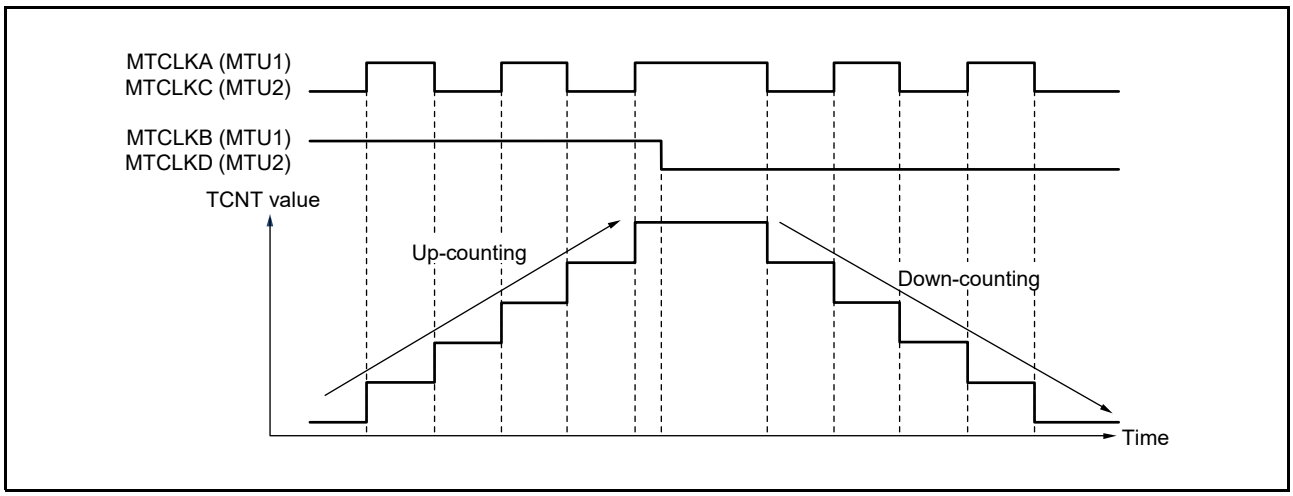


Figure 10.34 Example of Operation in Phase Counting Mode 2 (When MTUn.TCR2.PCB[1:0] is 1xb (n = 1, 2))

Table 10.68 Up-Counting and Down-Counting Conditions in Phase Counting Mode 2

PCB[1:0]	MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
00b	High	↑	Not counted (Don't care)
	Low	↓	
	↑	Low	
	↓	High	Up-counting
	High	↓	
	Low	↑	
	↑	High	
	↓	Low	Down-counting
01b	High	↑	Not counted (Don't care)
	Low	↓	
	↑	Low	Down-counting
	↓	High	Not counted (Don't care)
	High	↓	
	Low	↑	
	↑	High	Up-counting
	↓	Low	Not counted (Don't care)
1xb	High	↑	Not counted (Don't care)
	Low	↓	
	↑	Low	Down-counting
	↓	High	Up-counting
	High	↓	Not counted (Don't care)
	Low	↑	
	↑	High	Up-counting
	↓	Low	Down-counting

↑ : Rising edge

↓ : Falling edge

(c) Phase Counting Mode 3

Figure 10.35 to Figure 10.37 show the examples of operation in phase counting mode 3 and Table 10.69 summarizes the TCNT up-counting and down-counting conditions.

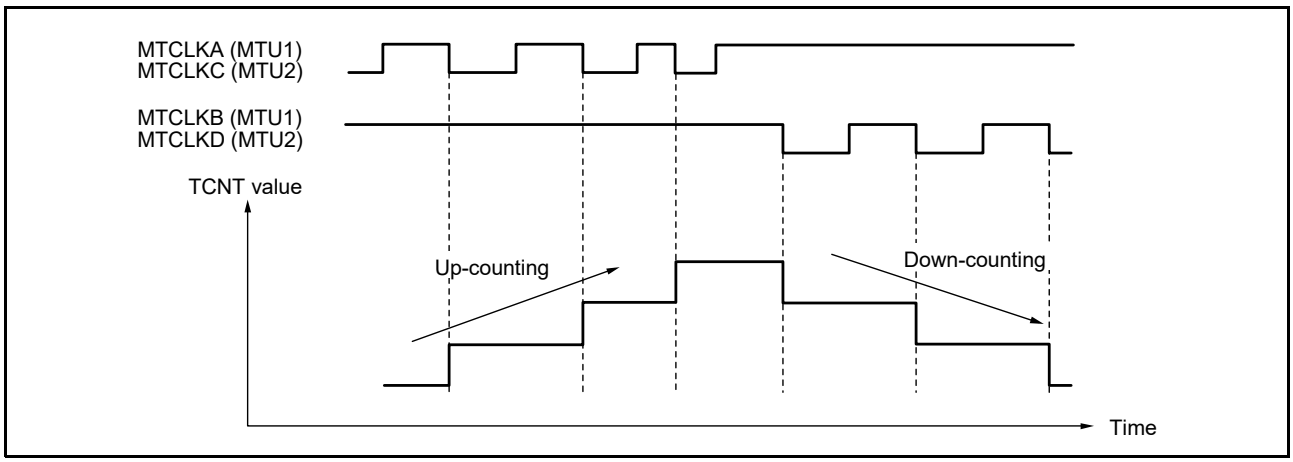


Figure 10.35 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] is 00b (n = 1, 2))

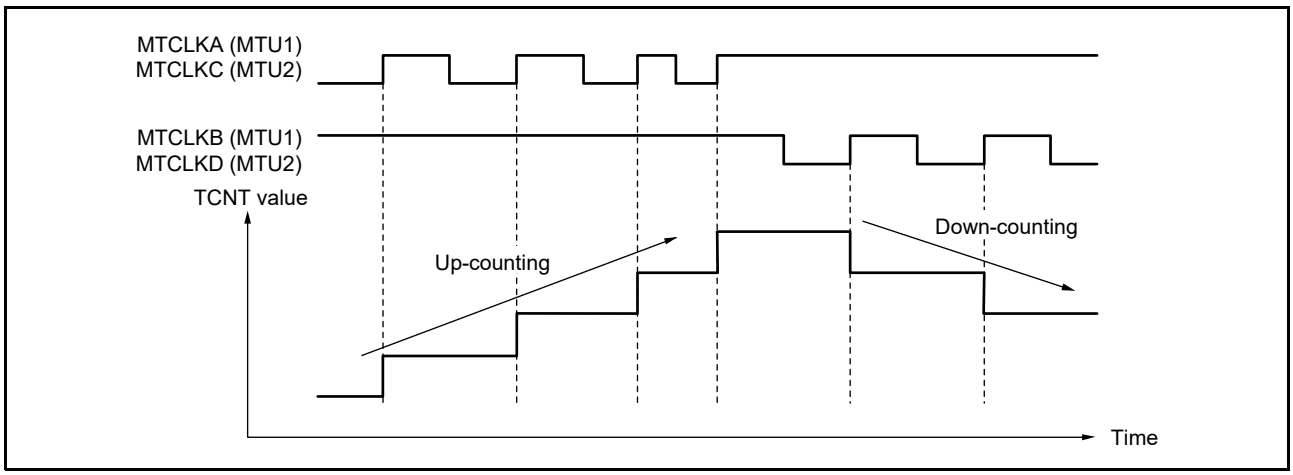


Figure 10.36 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] is 01b (n = 1, 2))

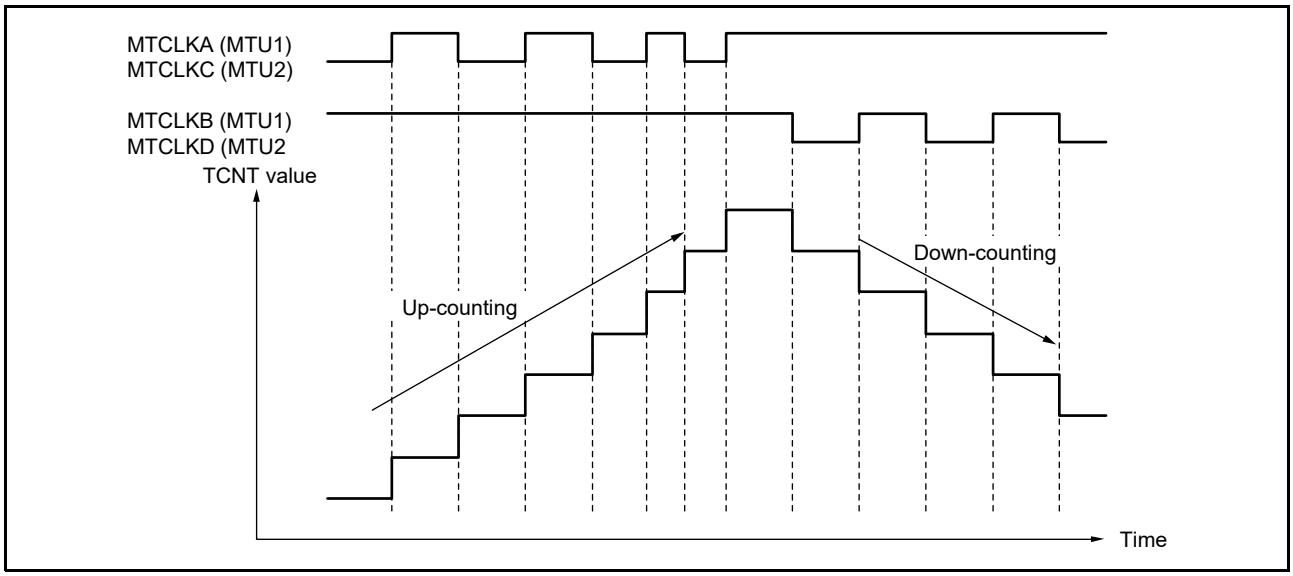


Figure 10.37 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] is 1xb (n = 1, 2))

Table 10.69 Up-Counting and Down-Counting Conditions in Phase Counting Mode 3

PCB[1:0]	MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
00b	High		Not counted (Don't care)
	Low		
		Low	
		High	Up-counting
	High		Down-counting
	Low		Not counted (Don't care)
		High	
		Low	
01b	High		Down-counting
	Low		Not counted (Don't care)
		Low	
		High	
	High		
	Low		
		High	Up-counting
		Low	Not counted (Don't care)
1xb	High		Down-counting
	Low		Not counted (Don't care)
		Low	
		High	Up-counting
	High		Down-counting
	Low		Not counted (Don't care)
		High	Up-counting
		Low	Not counted (Don't care)

 : Rising edge

 : Falling edge

(d) Phase Counting Mode 4

Figure 10.38 shows an example of operation in phase counting mode 4, and Table 10.70 summarizes the TCNT up-counting and down-counting conditions.

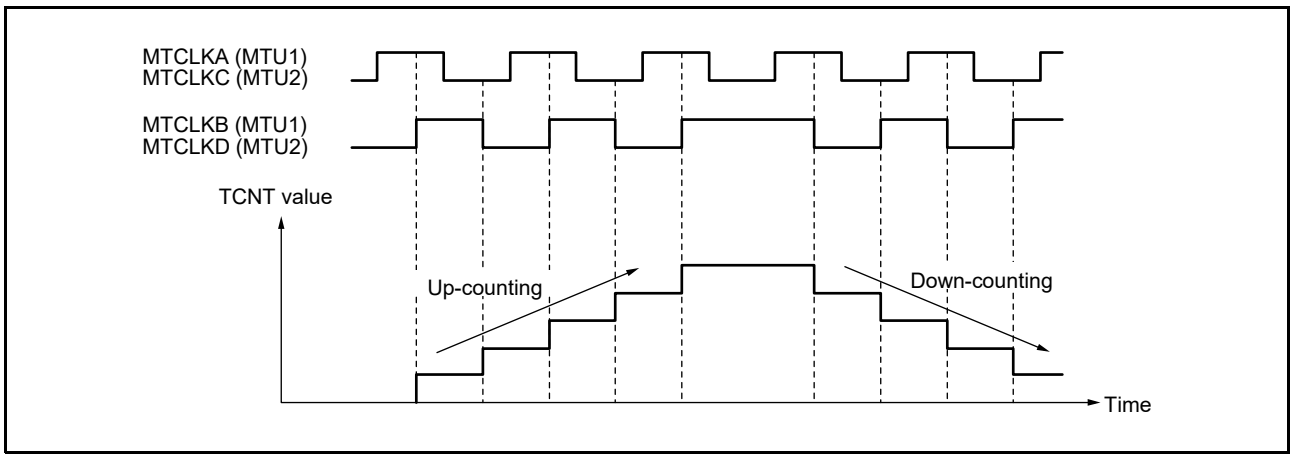


Figure 10.38 Example of Operation in Phase Counting Mode 4

Table 10.70 Up-Counting and Down-Counting Conditions in Phase Counting Mode 4

MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
High	↑	Up-counting
Low	↓	
↑	Low	Not counted (Don't care)
↓	High	
High	↓	Down-counting
Low	↑	
↑	High	Not counted (Don't care)
↓	Low	

↑ : Rising edge

↓ : Falling edge

(e) Phase Counting Mode 5

Figure 10.39 and Figure 10.40 show the examples of operation in phase counting mode 5 and Table 10.71 summarizes the TCNT up-counting and down-counting conditions.

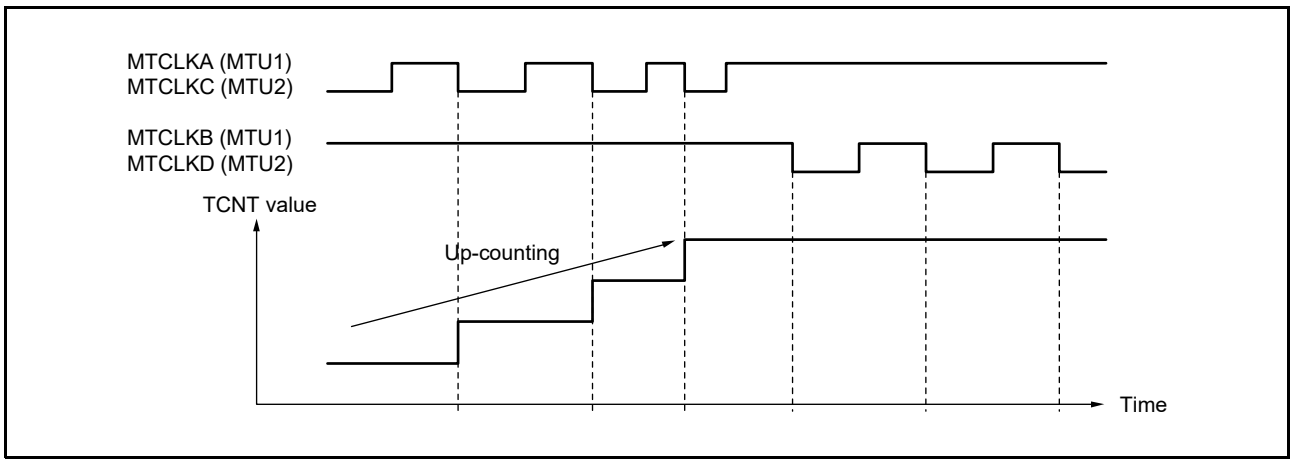


Figure 10.39 Example of Operation in Phase Counting Mode 5 (When MTUn.TCR2.PCB[1:0] = 0xb (n = 1, 2))

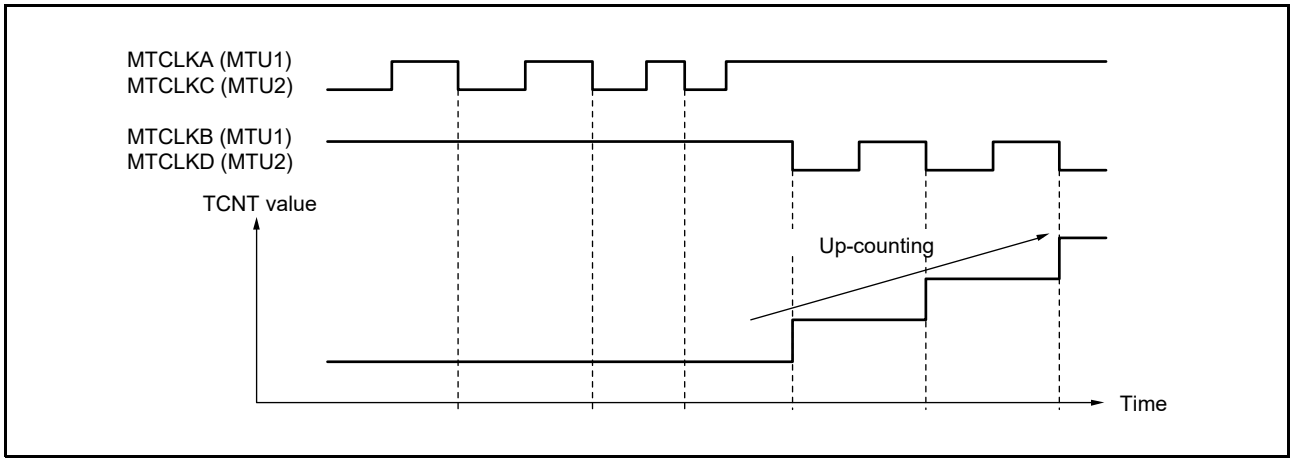


Figure 10.40 Example of Operation in Phase Counting Mode 5 (When MTUn.TCR2.PCB[1:0] = 1xb (n = 1, 2))

Table 10.71 Up-Counting and Down-Counting Conditions in Phase Counting Mode 5

PCB[1:0]	MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
0xb	High		Not counted (Don't care)
	Low		
		Low	
		High	Up-counting
	High		Not counted (Don't care)
	Low		
		High	
		Low	Up-counting
1xb	High		Not counted (Don't care)
	Low		Up-counting
		Low	Not counted (Don't care)
		High	
	High		Up-counting
	Low		Not counted (Don't care)
		High	
		Low	

 : Rising edge

 : Falling edge

(3) 16-Bit Phase Counting Mode Application Example

Figure 10.41 shows an example in which MTU1 is in phase counting mode, and MTU1 is coupled with MTU0 to input 2-phase encoder pulses of a servo motor in order to detect position or speed.
MTU1 is set to phase counting mode 1, and the encoder pulse A-phase and B-phase are input to MTCLKA and MTCLKB.
In MTU0, MTU0.TGRC compare match is specified as the TCNT clearing source and MTU0.TGRA and MTU0.TGRC are used for the compare match function and are set with the speed control cycle and position control cycle.
MTU0.TGRB is used for input capture, with MTU0.TGRB and MTU0.TGRD operating in buffer mode. The MTU1 count clock is designated as the MTU0.TGRB input capture source, and the widths of 2-phase encoder 4-multiplication pulses are detected.
MTU1.TGRA and MTU1.TGRB for MTU1 are designated for the input capture function and MTU0.TGRA and MTU0.TGRC compare matches in MTU0 are selected as the input capture sources to store the up/down-counter values for the control cycles.
This procedure enables the accurate detection of position and speed.

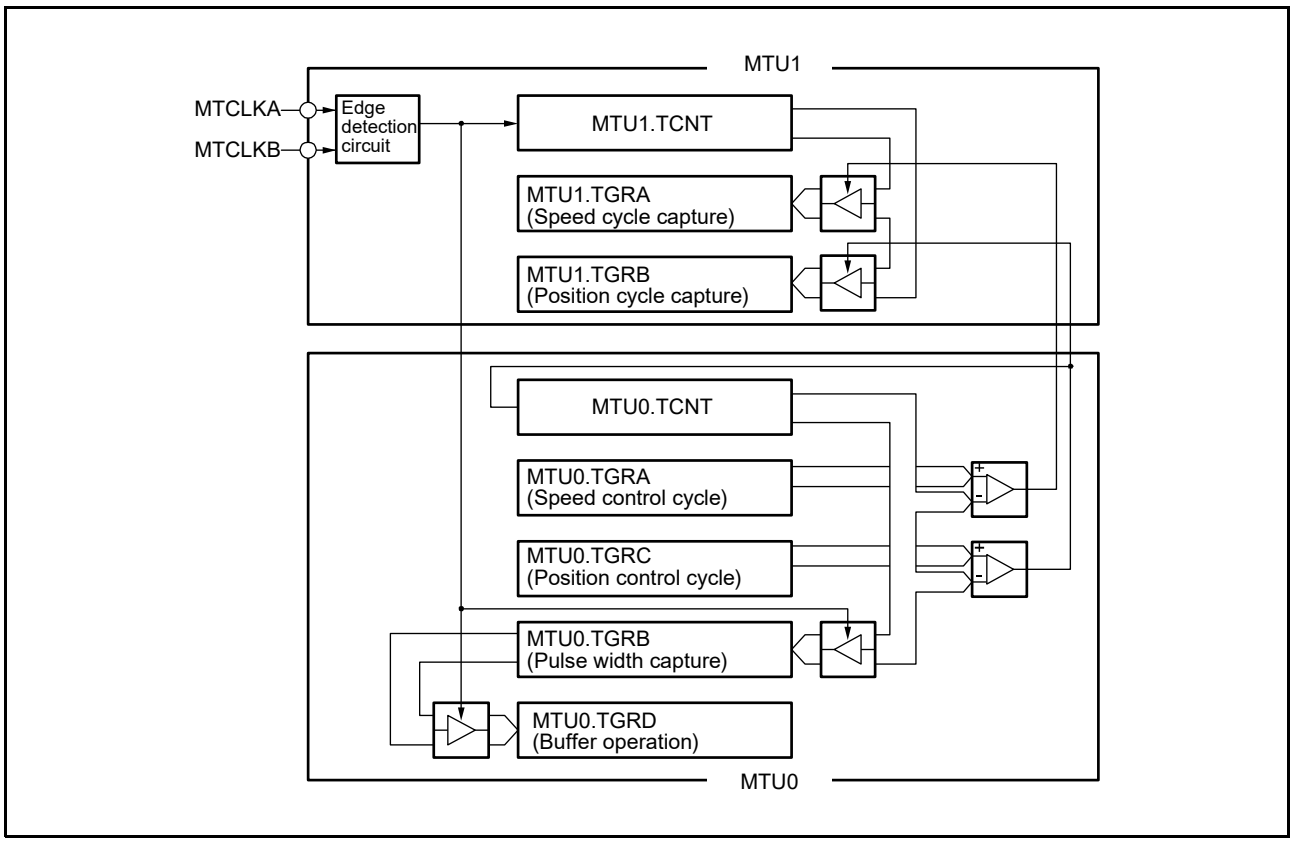


Figure 10.41 16-Bit Phase Counting Mode Application Example

10.3.6.2 Cascade Connection 32-Bit Phase Counting Mode

When MTU1 is set to phase counting mode by setting MTU1.TMDR3.LWA = 1, MTU1 and MTU2 are connected to operate in cascade connection 32-bit phase counting mode as shown in Figure 10.42. When this mode is used, the TCR, TCR2, TIOR, TIER, TGR, and TSR registers are controlled by MTU1 and the settings of MTU2 are disabled. See Figure 10.43 for the procedure for setting cascade connection 32-bit phase counting mode.

In this mode, three-phase (A, B, and Z) signals can be input. As an encoder pulse signal, the external input phase clocks MTCLKA and MTCLKB or MTCLKC and MTCLKD can be selected for A-phase and B-phase, and MTIOC1A can be selected for Z-phase, respectively. See Table 10.70 for selecting external clock input of A-phase and B-phase. A counter event is generated using an A-phase or B-phase pulse and counted by the 32-bit counter MTU1.TCNTLW.

An input capture can also be generated using a Z-phase signal; thus, an angular velocity can be measured using the value captured in the general register.

Furthermore, MTU8 can be used as a channel for measuring a 1-ms interval, and a compare match signal can be output at a 1-ms interval to the MTU1 and MTU2, which operate in cascade connection 32-bit phase counting mode. That is, a compare match signal of MTU8 is used as a capture signal of MTU1 and MTU2, and the number of A-phase and B-phase pulses for a 1-ms period can be measured.

When MTU0 or MTU5 is specified as the channel for measuring Z-phase signal pulses, this compare match signal of TGRC in MTU8 can be output as a capture signal or a clear signal to MTU0 or MTU5; thus, the Z-phase count can be measured at 1-ms intervals.

In addition, a counter event signal of combined MTU1 and MTU2 can be used as a capture signal of TGRD in MTU8, and measurement can be performed including the intervals of A, B, or both phase pulses. In this case, the TGRD register in MTU8 should be set to buffer operation.

Refer to section 10.3.4, Cascaded Operation, for details on the cascade connection function for connecting MTU1 and MTU2 in a mode other than cascade connection 32-bit phase counting mode.

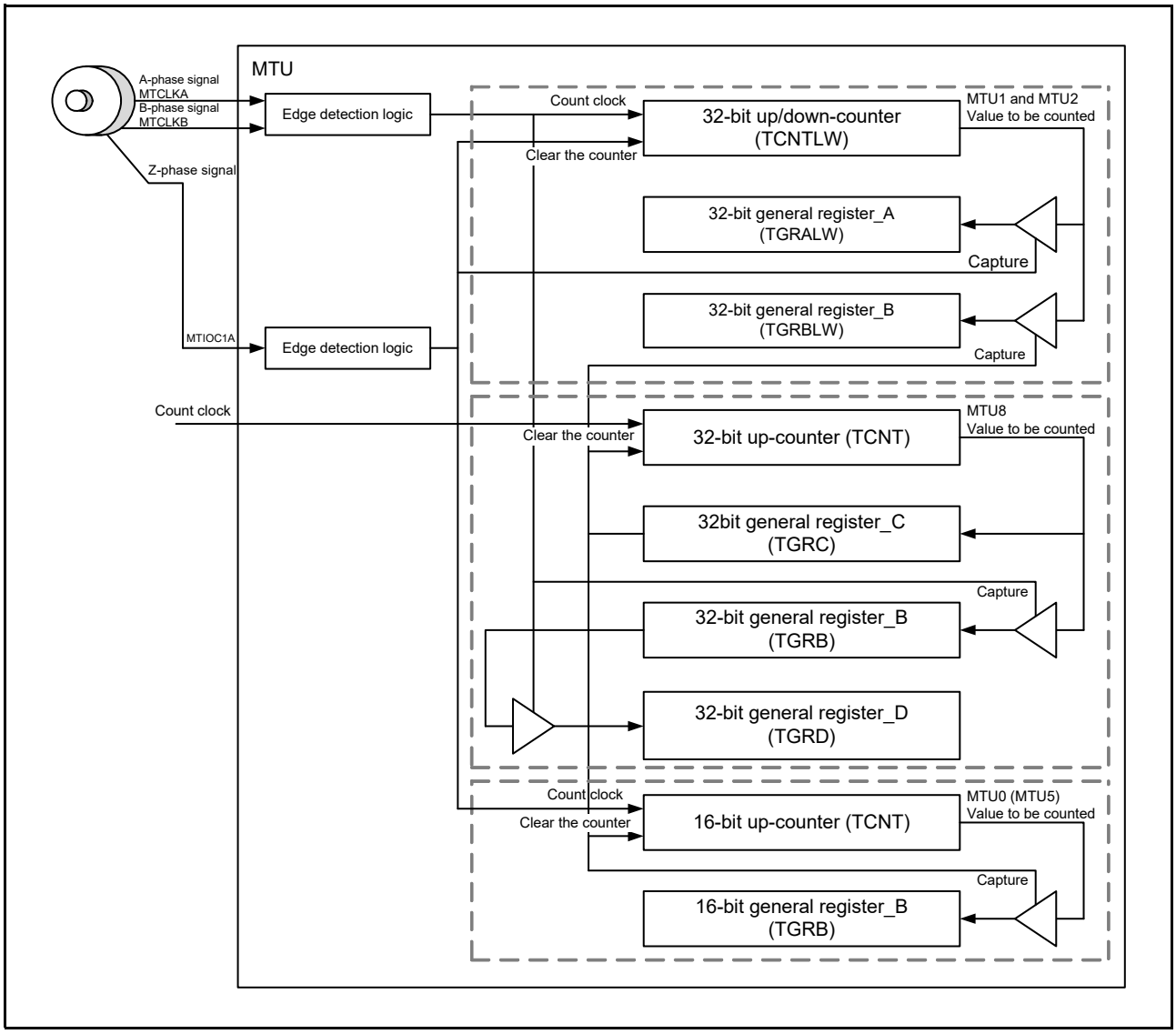


Figure 10.42 Block Diagram for Operation in Cascade Connection 32-Bit Phase Counting Mode

(1) Example of Setting Cascade Connection 32-Bit Phase Counting Mode

Figure 10.43 shows an example of the procedure for setting cascade connection 32-bit phase counting mode.

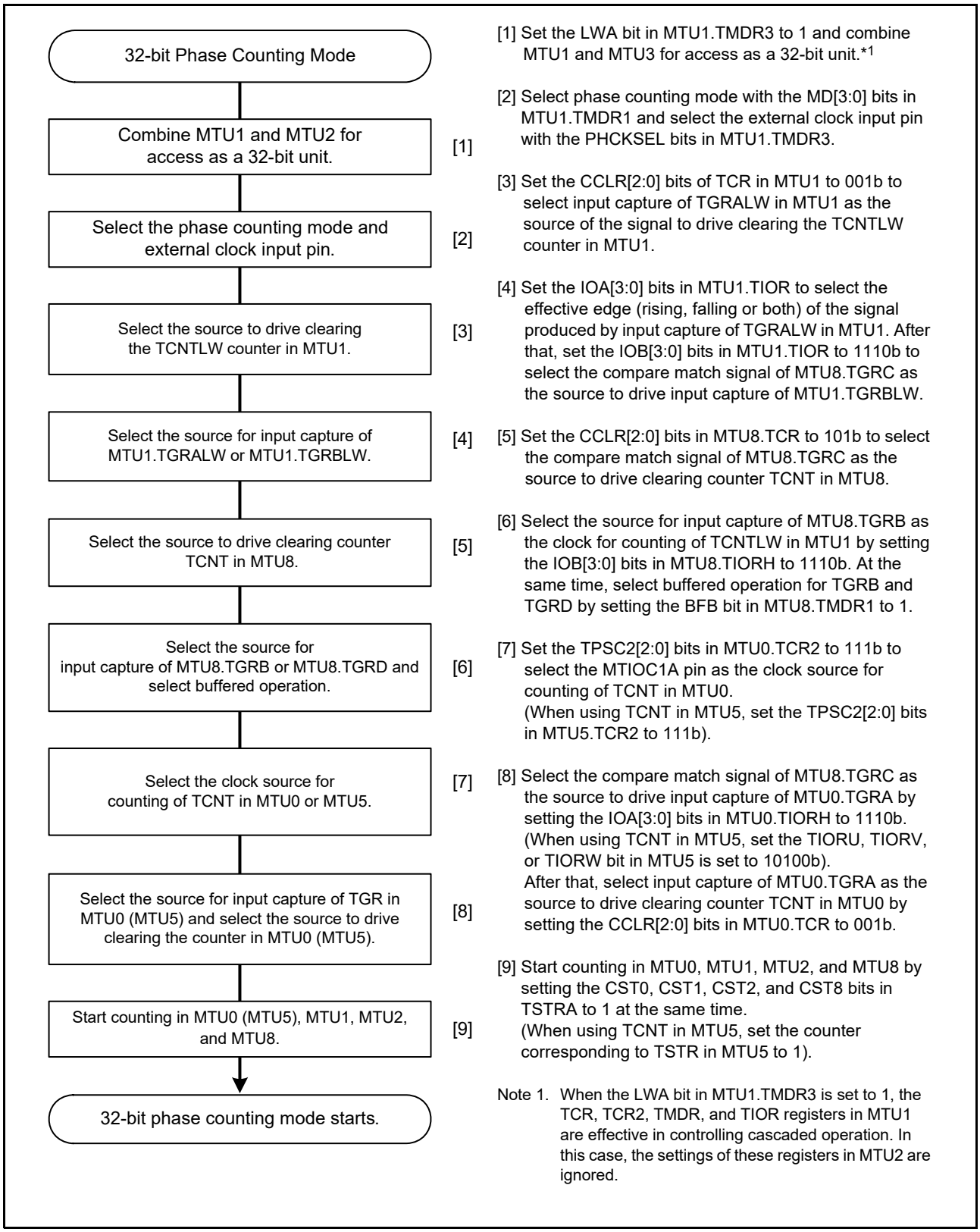


Figure 10.43 Procedure for Setting Cascade Connection 32-Bit Phase Counting Mode

10.3.7 Reset-Synchronized PWM Mode

In the reset-synchronized PWM mode, three phases of positive and negative PWM waveforms (six phases in total) that share a common wave transition point can be output by combining MTU3 and MTU4 and MTU6 and MTU7.
When set for reset-synchronized PWM mode, the MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, MTIOC4D, MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7C, MTIOC7B, and MTIOC7D pins function as PWM output pins and timer counters 6 and 12 (MTU3.TCNT and MTU6.TCNT) functions as an up-counter.

Table 10.72 shows the PWM output pins used. Table 10.73 shows the settings of the registers.

Table 10.72 Output Pins for Reset-Synchronized PWM Mode

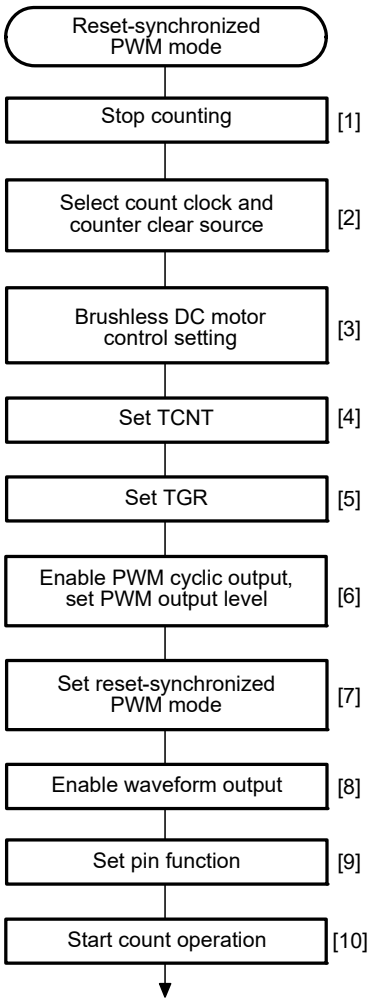
Channel	Output Pin	Description
MTU3	MTIOC3B	PWM output pin 1
	MTIOC3D	PWM output pin 1' (negative-phase waveform of PWM output 1)
MTU4	MTIOC4A	PWM output pin 2
	MTIOC4C	PWM output pin 2' (negative-phase waveform of PWM output 2)
	MTIOC4B	PWM output pin 3
	MTIOC4D	PWM output pin 3' (negative-phase waveform of PWM output 3)
MTU6	MTIOC6B	PWM output pin 4
	MTIOC6D	PWM output pin 4' (negative-phase waveform of PWM output 4)
MTU7	MTIOC7A	PWM output pin 5
	MTIOC7C	PWM output pin 5' (negative-phase waveform of PWM output 5)
	MTIOC7B	PWM output pin 6
	MTIOC7D	PWM output pin 6' (negative-phase waveform of PWM output 6)

Table 10.73 Register Settings for Reset-Synchronized PWM Mode

Register	Setting
MTU3.TCNT	Initial setting (0000h)
MTU4.TCNT	Initial setting (0000h)
MTU3.TGRA	Set the count cycle for MTU3.TCNT
MTU3.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC3B and MTIOC3D pins
MTU4.TGRA	Set the transition point of the PWM waveform to be output from the MTIOC4A and MTIOC4C pins
MTU4.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC4B and MTIOC4D pins
MTU6.TCNT	Initial setting (0000h)
MTU7.TCNT	Initial setting (0000h)
MTU6.TGRA	Set the count cycle for MTU6.TCNT
MTU6.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC6B and MTIOC6D pins
MTU7.TGRA	Set the transition point of the PWM waveform to be output from the MTIOC7A and MTIOC7C pins
MTU7.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC7B and MTIOC7D pins

(1) Example of Procedure for Setting Reset-Synchronized PWM Mode

Figure 10.44 shows an example of the procedure for specifying the reset-synchronized PWM mode.



[1] Clear the TSTRA.CST3 (TSTRB.CST6) and TSTRA.CST4 (TSTRB.CST7) bits to 0 to stop the TCNT count operation. Specify the reset-synchronized PWM mode while MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) are stopped.

[2] Set the TCR.TPSC[2:0] and TCR.CKEG[1:0] bits of MTU3 (MTU6) to select the count clock source and clock edge for MTU3 (MTU6). Set the TCR.CCLR[2:0] bits to select TGRA compare match as the counter clear source.

[3] To control a brushless DC motor, set the TGCRA.BDC bit and set the feedback signal input source and output chopping or gate signal direct output (only when MTU3 and MTU4 are used).

[4] Set MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) to 0000h.

[5] MTU3.TGRA (MTU6.TGRA) is the cycle register. Set the PWM cycle in MTU3.TGRA (MTU6.TGRA).
Set the PWM output transition timing in MTU3.TGRB (MTU6.TGRB), MTU4.TGRA (MTU7.TGRA), and MTU4.TGRB (MTU7.TGRB).
Note that the setting values should be within the range of compare match with MTU3.TCNT (MTU6.TCNT).

$X \leq$ MTU3.TGRA (X: Set value)

[6] Set the PSYE bit in the TOCR1A (TOCR1B) register to enable or disable toggle output synchronized with the PMW cycle, and set the OLSP and OLSN bits to select the PWM output level.
When specifying the PWM output level using the TOCR2A (TOCR2B) and TOLBRA (TOLBRB) in buffer operation, see Timer Output Level Buffer Registers (TOLBRA and TOLBRB).

[7] Set the MD[3:0] bits in MTU3.TMDR1 (MTU6.TMDR1) register to 1000b to select the reset-synchronized PWM mode. Specify the buffer operation (BFA = 1) of the MTU3.TGRA (MTU6.TGRA) register and the buffer operation (BFB = 1) of the MTU3.TGRB register when necessary. Set the MTU4.TMDR1 (MTU7.TMDR1) register to the initial value (normal operation; BFA = 0 and BFB = 0).

[8] Set the TOERA (TOERB) register to enable or disable output from the PWM output pin. Then, set the MTU3.TIORH (MTU6.TIORH), MTU3.TIORL (MTU6.TIORL), MTU4.TIORH (MTU7.TIORH), and MTU4.TIORL (MTU7.TIORL) registers to 00h.

[9] Set the port mode registers (PMR) and Pmn pin function control register (PmnPFS) of the GPIO ports.

[10] Set the TSTRA.CST3 (TSTRB.CST6) bit to 1 to start count operation.

Note: The output waveform starts to toggle at the point of MTU3.TCNT (MTU6.TCNT) = MTU3.TGRA (MTU6.TGRA) = X by setting X = MTU3.TGRA (MTU6.TGRA), i.e., cycle = duty ratio.

Figure 10.44 Procedure for Selecting Reset-Synchronized PWM Mode

(2) Example of Reset-Synchronized PWM Mode Operation

Figure 10.45 shows an example of operation in the reset-synchronized PWM mode.
MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT) operate as up-counters. The counters are cleared when a compare match occurs between MTU3.TCNT (MTU6.TCNT) and MTU3.TGRA (MTU6.TGRA), and then begin incrementing from 0000h. The output from the PWM pins toggles every time a compare match occurs in MTU3.TGRB (MTU6.TGRB), MTU4.TGRA (MTU7.TGRA), and MTU4.TGRB (MTU7.TGRB) and the counters are cleared.

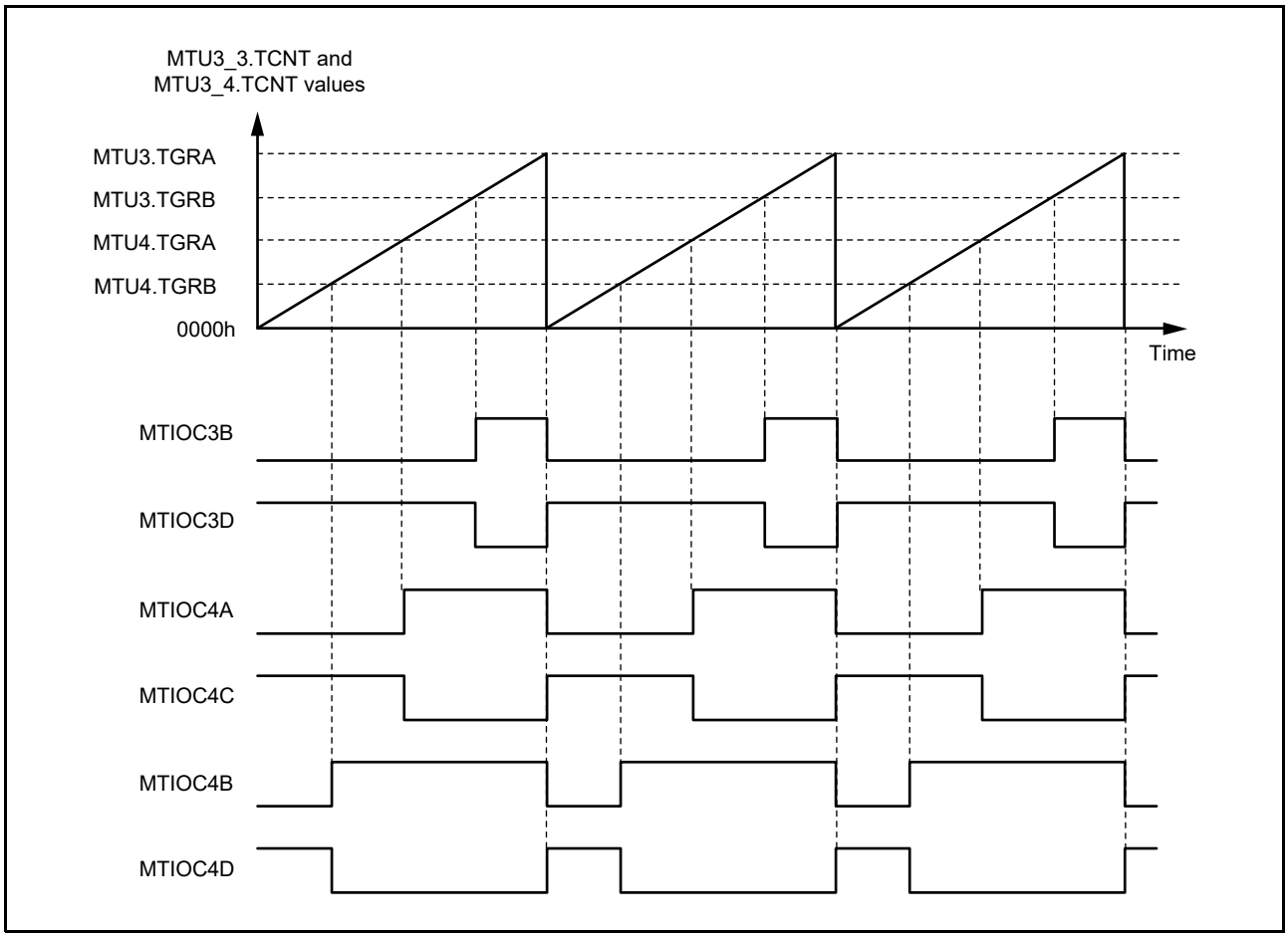


Figure 10.45 Example of Reset-Synchronized PWM Mode Operation
(When OLSN = 1 and OLSP = 1 in MTU3.TOCR1 and MTU4.TOCR1)

10.3.8 Complementary PWM Mode

In complementary PWM mode, dead time can be set for PWM waveforms to be output. The dead time is the period during which the upper and lower arm transistors are set to the inactive level in order to prevent short-circuiting of the arms.
Six positive-phase and six negative-phase PWM waveforms (12 phases in total) with dead time can be output by combining MTU3/ MTU4 and MTU6/MTU7. PWM waveforms without dead time can also be output.
In complementary PWM mode, MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D, MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7B, MTIOC7C, and MTIOC7D pins function as PWM output pins, and the MTIOC3A and MTIOC6A pins can be set for toggle output synchronized with the PWM cycle.
MTU3.TCNT, MTU4.TCNT, MTU6.TCNT, and MTU7.TCNT function as up/down-counters.

Table 10.74 shows the PWM output pins used. Table 10.75 shows the settings of the registers used.
A function to directly cut off the PWM output by using an external signal is supported as a port function.

Table 10.74 Output Pins for Complementary PWM Mode

Channel	Output Pin	Description
MTU3	MTIOC3A	Toggle output synchronized with PWM cycle (or I/O port)
	MTIOC3B	PWM output pin 1
	MTIOC3C	I/O port*1
	MTIOC3D	PWM output pin 1' (negative-phase waveform output of PWM output 1)
MTU4	MTIOC4A	PWM output pin 2
	MTIOC4C	PWM output pin 2' (negative-phase waveform output of PWM output 1)
	MTIOC4B	PWM output pin 3
	MTIOC4D	PWM output pin 3' (negative-phase waveform output of PWM output 1)
MTU6	MTIOC6A	Toggle output synchronized with PWM cycle (or I/O port)
	MTIOC6B	PWM output pin 4
	MTIOC6C	I/O port*1
	MTIOC6D	PWM output pin 4' (negative-phase waveform output of PWM output 4)
MTU7	MTIOC7A	PWM output pin 5
	MTIOC7C	PWM output pin 5' (negative-phase waveform output of PWM output 5)
	MTIOC7B	PWM output pin 6
	MTIOC7D	PWM output pin 6' (negative-phase waveform output of PWM output 6)

Note 1. Avoid setting the MTIOC3C and MTIOC6C pins as timer I/O pins in complementary PWM mode.

Table 10.75 Register Settings for Complementary PWM Mode (1/2)

Channel	Counter / Register	Description	Read/Write from CPU
MTU3	TCNT	Starts up-counting from the value set in the dead time register	Maskable by TRWERA setting*1
	TGRA	Set MTU3.TCNT upper limit value (1/2 carrier cycle + dead time)	Maskable by TRWERA setting*1
	TGRB	PWM output 1 compare register	Maskable by TRWERA setting*1
	TGRC	MTU3.TGRA buffer register	Readable/writable
	TGRD	PWM output 1/MTU3.TGRB buffer register	Readable/writable
	TGRE	MTU3.TGRB buffer register B (when double buffer function is used)	Readable/writable
MTU4	TCNT	Starts up-counting after being initialized to 0000h	Maskable by TRWERA setting*1
	TGRA	PWM output 2 compare register	Maskable by TRWERA setting*1
	TGRB	PWM output 3 compare register	Maskable by TRWERA setting*1
	TGRC	PWM output 2/MTU4.TGRA buffer register	Readable/writable
	TGRD	PWM output 3/MTU4.TGRB buffer register	Readable/writable
	TGRE	MTU4.TGRA buffer register B (when double buffer function is used)	Readable/writable
	TGRF	MTU4.TGRB buffer register B (when double buffer function is used)	Readable/writable
MTU6	TCNT	Starts up-counting from the value set in the dead time register	Maskable by TRWERB setting*2
	TGRA	Set MTU6.TCNT upper limit value (1/2 carrier cycle + dead time)	Maskable by TRWERB setting*2
	TGRB	PWM output 4 compare register	Maskable by TRWERB setting*2
	TGRC	MTU6.TGRA buffer register	Readable/writable
	TGRD	PWM output 4/MTU6.TGRB buffer register	Readable/writable
	TGRE	MTU6.TGRB buffer register B (when double buffer function is used)	Readable/writable
MTU7	TCNT	Starts up-counting after being initialized to 0000h	Maskable by TRWERB setting*2
	TGRA	PWM output 5 compare register	Maskable by TRWERB setting*2
	TGRB	PWM output 6 compare register	Maskable by TRWERB setting*2
	TGRC	PWM output 5/MTU7.TGRA buffer register	Readable/writable
	TGRD	PWM output 6/MTU7.TGRB buffer register	Readable/writable
	TGRE	MTU7.TGRA buffer register B (when double buffer function is used)	Readable/writable
	TGRF	MTU7.TGRB buffer register B (when double buffer function is used)	Readable/writable

Note 1. Access can be enabled or disabled according to the setting in TRWERA (timer read/write enable register A).
Note 2. Access can be enabled or disabled according to the setting in TRWERB (timer read/write enable register B).

Table 10.76 Register Settings for Complementary PWM Mode (2/2)

Counter / Register	Description	Read/Write from CPU
Timer dead time data register A (TDDRA)	Set MTU4.TCNT and MTU3.TCNT offset value (dead time value)	Maskable by TRWERA setting*1
Timer dead time data register B (TDDRB)	Set MTU7.TCNT and MTU6.TCNT offset value (dead time value)	Maskable by TRWERB setting*2
Timer cycle data register A (TCDRA)	Set MTU4.TCNT upper limit value (1/2 carrier cycle)	Maskable by TRWERA setting*1
Timer cycle data register B (TCDRB)	Set MTU7.TCNT upper limit value (1/2 carrier cycle)	Maskable by TRWERB setting*2
Timer cycle buffer register A (TCBRA)	TCDRA buffer register	Readable/writable
Timer cycle buffer register B (TCBRB)	TCDRB buffer register	Readable/writable
Subcounter A (TCNTSA)	Subcounter A for dead time generation	Read-only
Subcounter B (TCNTSB)	Subcounter B for dead time generation	Read-only
Temporary register 1A (TEMP1A)	PWM output 1/MTU3.TGRB temporary register A	Not readable or writable
Temporary register 1B (TEMP1B)	PWM output 1/MTU3.TGRB temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 2A (TEMP2A)	PWM output 2/MTU4.TGRA temporary register A	Not readable or writable
Temporary register 2B (TEMP2B)	PWM output 2/MTU4.TGRA temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 3A (TEMP3A)	PWM output 3/MTU4.TGRB temporary register A	Not readable or writable
Temporary register 3B (TEMP3B)	PWM output 3/MTU4.TGRB temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 4A (TEMP4A)	PWM output 4/MTU6.TGRB temporary register A	Not readable or writable
Temporary register 4B (TEMP4B)	PWM output 4/MTU6.TGRB temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 5A (TEMP5A)	PWM output 5/MTU7.TGRA temporary register A	Not readable or writable
Temporary register 5B (TEMP5B)	PWM output 5/MTU7.TGRA temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 6A (TEMP6A)	PWM output 6/MTU7.TGRB temporary register A	Not readable or writable
Temporary register 6B (TEMP6B)	PWM output 6/MTU7.TGRB temporary register B (when double buffer function is used)	Not readable or writable

Note 1. Access can be enabled or disabled according to the setting in TRWERA (timer read/write enable register A).
Note 2. Access can be enabled or disabled according to the setting in TRWERB (timer read/write enable register B).

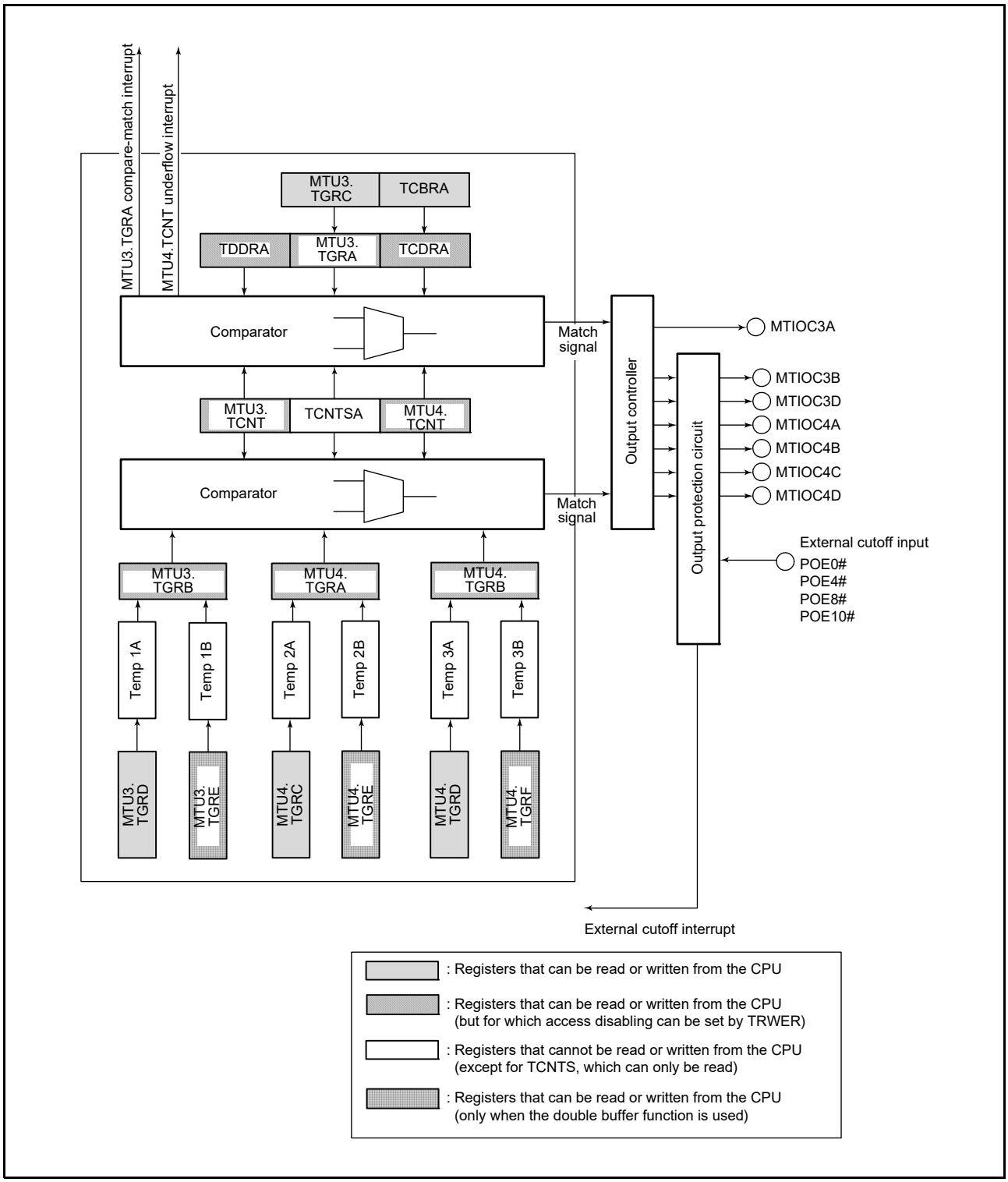


Figure 10.46 Block Diagram of MTU3 and MTU4 in Complementary PWM Mode

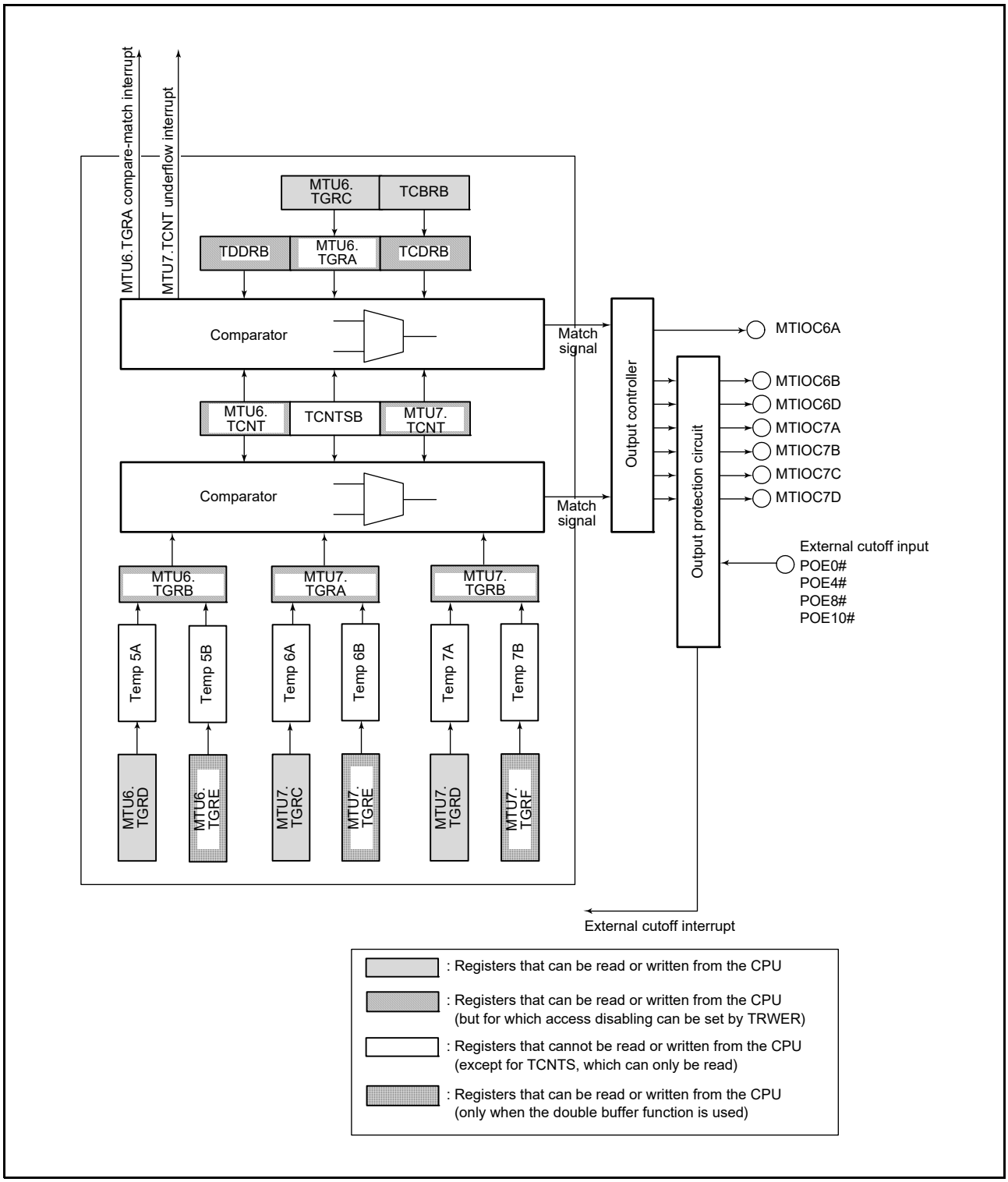
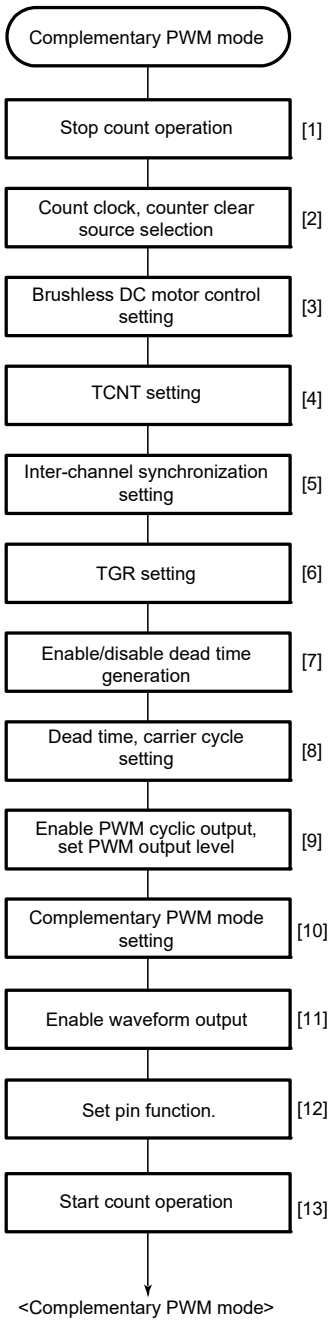


Figure 10.47 Block Diagram of MTU6 and MTU7 in Complementary PWM Mode

(1) Example of Complementary PWM Mode Setting Procedure

Figure 10.48 shows an example of the complementary PWM mode setting procedure.



[1] Clear the TSTRA.CST3 (TSTRB.CST6) and TSTRA.CST4 (TSTRB.CST7) bits to 0 to stop the TCNT count operation.
Specify complementary PWM mode while MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) are stopped.

[2] Set the TCR.TPSC[2:0] and TCR.CKEG[1:0] bits of MTU3 and MTU4 (MTU6 and MTU7) to select the same count clock source and clock edge for MTU3 and MTU4 (MTU6 and MTU7).
Set the TCR.CCLR[2:0] bits only when restarting by synchronous clearing with another channel during complementary PWM operation.

[3] To control a brushless DC motor, set the TGCRA.BDC bit and set the feedback signal input source and output chopping or gate signal direct output (only when MTU3 and MTU4 are used).

[4] Set the dead time in MTU3.TCNT (MTU6.TCNT). Set MTU4.TCNT (MTU7.TCNT) to 0000h.

[5] Make this setting only when restarting by synchronous clearing with another channel during complementary PWM mode operation. In this case, set the TSYRA register (TSYRB) to synchronize the channel generating the synchronous clear with MTU3 and MTU4 (MTU6 and MTU7).

[6] When MTU3 and MTU6 are used, set the PWM output duty ratio in the duty registers (MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB) and buffer registers (MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD).
Be sure to set the same value in each pair of the duty registers (MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD) and buffer registers (MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB).
Only when the double buffer function is used, set the PWM output duty value - 1 in the buffer registers (MTU3.TGRE, MTU4.TGRE, and MTU4.TGRF).
When MTU6 and MTU7 are used, set the value to the following registers:
Duty registers (MTU6.TGRB, MTU7.TGRA, and MTU7.TGRB)
Buffer registers (MTU6.TGRD, MTU7.TGRC, and MTU7.TGRD)
Buffer registers when the double buffer function is used (MTU6.TGRE, MTU7.TGRE, and MTU7.TGRF)

[7] This setting is necessary only when no dead time should be generated. Make appropriate settings in the TDERA (TDERB) register so that no dead time generated.

[8] Set the dead time in the TDDRA (TDDRB) register, 1/2 the PWM cycle in the TCDRA (TCDRB) register and TCBRA (TCBRB) register, and 1/2 the PWM cycle plus the dead time in MTU3.TGRA (MTU6.TGRA) and MTU3.TGRC (MTU6.TGRC).
When no dead time generation is selected, set 1 in the TDDRA (TDDRB) register and 1/2 the PWM cycle + 1 in MTU3.TGRA (MTU6.TGRA) and MTU3.TGRC (MTU6.TGRC).

[9] Set the PSYE bit in the TOCR1A register to enable or disable toggle output synchronized with the PWM cycle, and set the OLSP and OLSN bits to select the PWM output level.
When specifying the PWM output level using the TOCR2A (TOCR2B) and TOLBRA (TOLBRA) in buffer operation, see Timer Output Level Buffer Registers (TOLBRA and TOLBRB).

[10] Select complementary PWM mode in MTU3.TMDR1 (MTU6.TMDR1). Specify the buffer operation (BFA = 1) of MTU3.TGRA (MTU6.TGRA) and the buffer operation (BFB = 1) of MTU3.TGRB (MTU6.TGRB).
Only when the double buffer function is used, set the TMDR2A (TMDR2B) register to enable the double buffer function.

[11] Set the TOERA (TOERB) register to enable or disable output from the PWM output pin. Then, set the MTU3.TIORH (MTU6.TIORH), MTU3.TIORL (MTU6.TIORL), MTU4.TIORH (MTU7.TIORH), and MTU4.TIORL (MTU7.TIORL) to 00h.

[12] Set the port mode registers (PMR) and Pmn pin function control register (PmnPFS) of the GPIO ports.

[13] Set the TSTRA.CST3 (TSTRB.CST6) and TSTRA.CST4 (TSTRB.CST7) bits to 1 simultaneously to start count operation.

Figure 10.48 Example of Complementary PWM Mode Setting Procedure

(2) Outline of Complementary PWM Mode Operation

In complementary PWM mode, six phases (three positive and three negative) PWM waveforms can be output. Figure 10.49 illustrates counter operation in complementary PWM mode (MTU3 and MTU4), and Figure 10.50 shows an example of operation in complementary PWM mode.

(a) Counter Operation

In complementary PWM mode, three counters—MTU3.TCNT, MTU4.TCNT, and TCNTSA (MTU6.TCNT, MTU7.TCNT, and TCNTSB) —in each unit perform up-/down-count operations.
MTU3.TCNT (MTU6.TCNT) is automatically initialized to the value set in TDDRA (TDDRB) when complementary PWM mode is selected and the CST3 bit in TSTRA (TSTRB) is 0. When the CST3 bit is set to 1, MTU3.TCNT (MTU6.TCNT) counts up to the value set in MTU3.TGRA (MTU6.TGRA), then switches to down-counting when it matches MTU3.TGRA (MTU6.TGRA). When the MTU4.TCNT (MTU7.TCNT) value matches 0000h, MTU3.TCNT (MTU6.TCNT) switches to up-counting, and the operation is repeated in this way.
TCNT in MTU4 (MTU7) should be initialized to 0000h after a reset. When the CST4 bit is set to 1, MTU4.TCNT (MTU7.TCNT) counts up in synchronization with MTU3.TCNT (MTU6.TCNT), and switches to down-counting when MTU3.TCNT (MTU6.TCNT) matches MTU3.TGRA (MTU6.TGRA). On reaching 0000h, MTU4.TCNT (MTU7.TCNT) switches to up-counting, and the operation is repeated in this way. TCNTSA (TCNTSB) is a read-only counter. It does not need to be initialized after a reset.
In counting up by MTU3.TCNT and MTU4.TCNT (or MTU6.TCNT and MTU7.TCNT), MTU3.TCNT (or MTU6.TCNT) starts counting up when it matches TCDRA (or TCDRB) and switches to counting down when it matches MTU3.TGRA (or MTU6.TGRA). Furthermore, when MTU4.TCNT (or MTU7.TCNT) matches TDDRA (or TDDRB), TCNTSA (or TCNTSB) is set to the value in MTU3.TGRA (or MTU6.TGRA) and counting is stopped.
When MTU4.TCNT (MTU7.TCNT) matches TDDRA (TDDRB) during down-counting of MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT), TCNTSA (TCNTSB) starts up-counting, and when MTU4.TCNT (MTU7.TCNT) matches 0000h, the operation switches to down-counting. When MTU3.TCNT (MTU6.TCNT) matches TCDRA (TCDRB), TCNTSA (TCNTSB) is cleared to 0000h and stops counting.
TCNTSA (TCNTSB) is compared with the compare register and temporary register, in which the PWM duty is specified, only during the count operation.

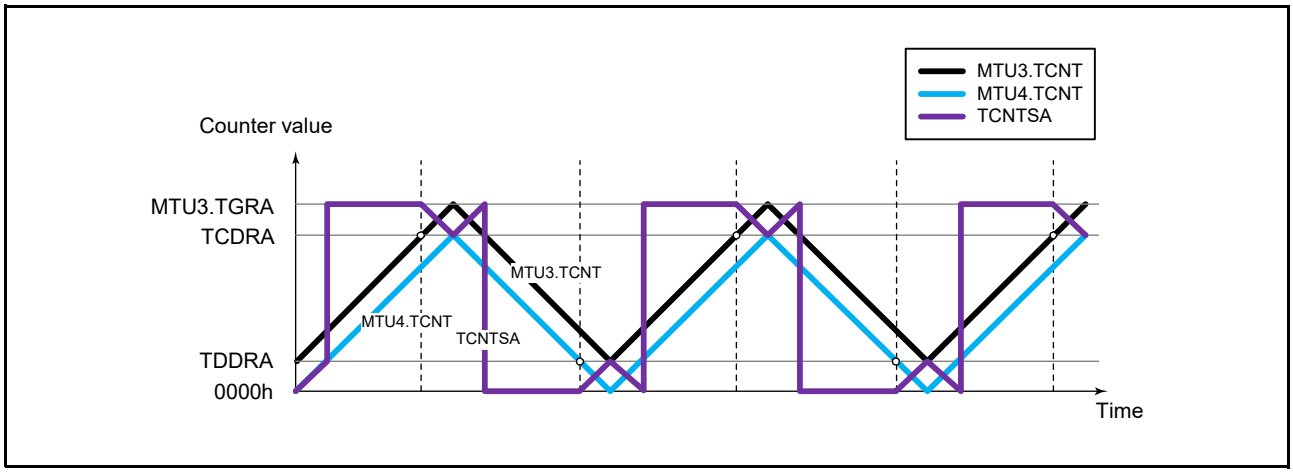


Figure 10.49 Count Operation in Complementary PWM Mode (MTU3 and MTU4)

(b) Register Operation

In complementary PWM mode, nine registers (compare registers, buffer registers, and temporary registers) are used to control the duty ratio for the PWM output. Figure 10.50 shows an example of operation in complementary PWM mode (MTU3 and MTU4).

MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB (MTU6.TGRB, MTU7.TGRA, and MTU7.TGRB) are constantly compared with the counters to generate PWM waveforms. When these registers match the counter, the value set in the OLSN and OLSP bits in the timer output control registers (TOCR1A and TOCR1B) is output from the PWM output pin.

MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD (MTU6.TGRD, MTU7.TGRC, and MTU7.TGRD) are buffer registers for these compare registers.

When the double buffer function is used, MTU3.TGRE, MTU4.TGRE, and MTU4.TGRF (MTU6.TGRE, MTU7.TGRE, and MTU7.TGRF) are also used as buffer registers B. For details of double buffer operation, refer to section 10.3.8 (2) (s), Double Buffer Function in Complementary PWM Mode.

Data in a compare register can be changed by writing new data to the corresponding buffer register. The buffer registers can be read or written at any time.

When modifying data in a buffer register, write to MTU4.TGRD (MTU7.TGRD) last and enable data transfer from the buffer register to a temporary register. At this time, transfer from the TCBRA (TCBRB) register and MTU3.TGRC (MTU6.TGRC) register, which operate as buffer registers for the timer cycle registers, to temporary registers is also enabled. Data is transferred to all five temporary registers at the same time.

When transfer is enabled in the Ta interval, data written to a buffer register is transferred to the temporary register. The data is not transferred to the temporary register in the Tb1 and Tb2 intervals. Data enabled for transfer in this interval is transferred to the temporary register at the end of this interval.

The value transferred to a temporary register is transferred to the compare register at the end of the Tb1 interval (when matches MTU3.TGRA (MTU6.TGRA) while TCNTSA (TCNTSB) is counting up), or at the end of the Tb2 interval (when matches 0000h while TCNTSA (TCNTSB) is counting down). The timing for transfer from the temporary register to the compare register can be selected with bits MD[3:0] in the timer mode register 1 (TMDR1). Figure 10.50 shows an example in which the trough is selected for the transfer timing.

In the Tb interval in which data is not transferred to the temporary register (Tb1 in Figure 10.50), the temporary register has the same function as the compare register and is compared with the counter. In this interval, therefore, there are two compare match registers for one output phase; the compare register contains the pre-change data and the temporary register contains new data. In this interval, three counters MTU3.TCNT, MTU4.TCNT, and TCNTSA (MTU6.TCNT, MTU7.TCNT, and TCNTSB) and two registers (compare register and temporary register) are compared, and PWM output is controlled accordingly.

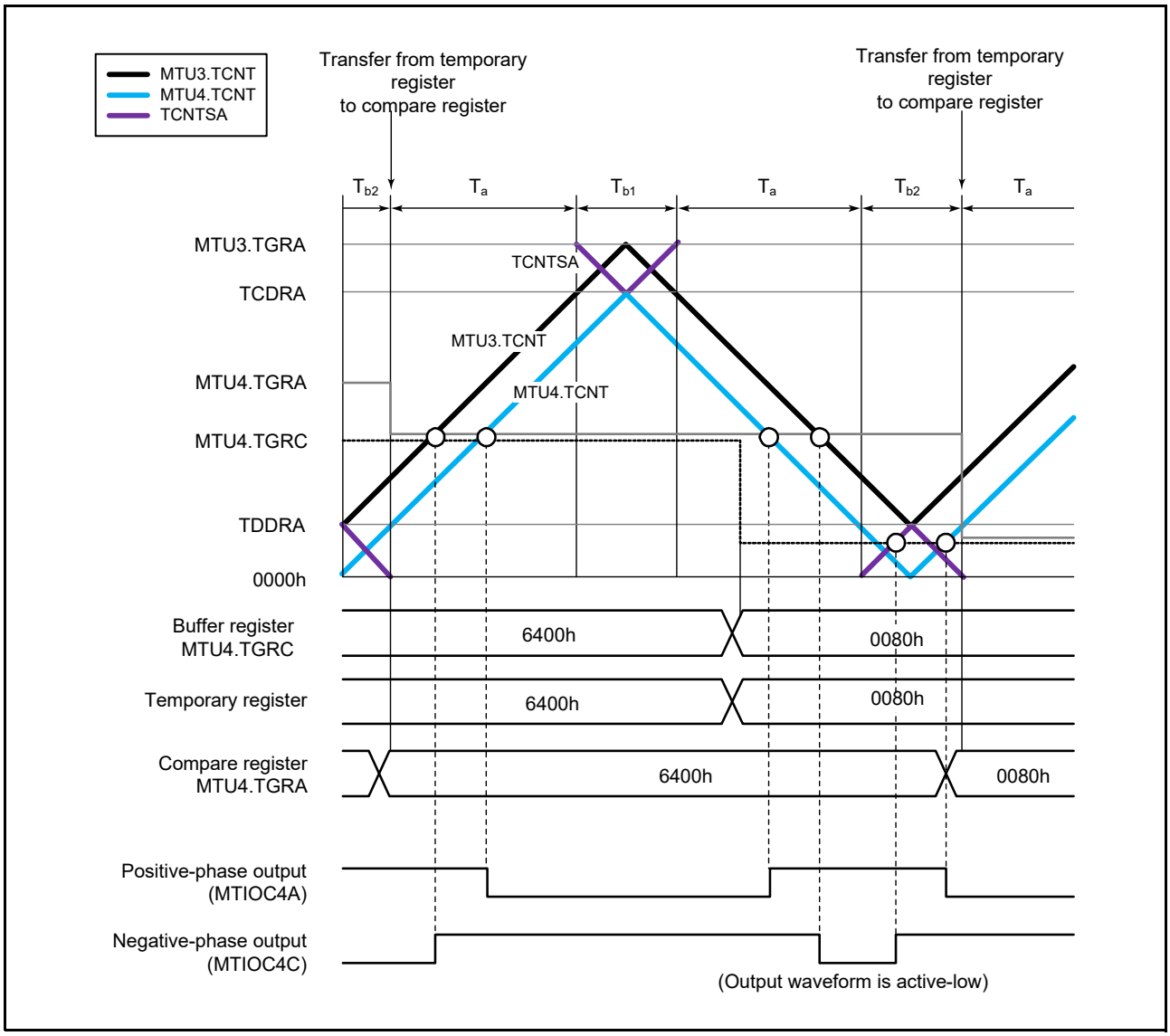


Figure 10.50 Example of Operation in Complementary PWM Mode (MTU3 and MTU4)

(c) Initial Setting

In complementary PWM mode, there are nine registers that require initial setting. In addition, there is a register that specifies whether to generate dead time (it should be used only when dead time generation should be disabled).
Before setting complementary PWM mode with MTU3.TMDR1.MD[3:0] (MTU6.TMDR1.MD[3:0]) bits, initial values should be set in the following registers.
The TOCR1A, TOCR2A, TOCR1B, and TOCR2B registers are used to set the PWM output level. MTU3.TGRC (MTU6.TGRC) operates as the buffer register for MTU3.TGRA (MTU6.TGRA), and should be set with 1/2 the PWM cycle + dead time Td. The timer cycle buffer register (TCBRA or TCBRB) operates as the buffer register for the timer cycle data register (TCDRA or TCDRB), and should be set with 1/2 the PWM cycle. Set dead time Td in the timer dead time data register (TDDRA or TDDRB).
When dead time is not needed, the TDER bit in the timer dead time enable register (TDERA or TDERB) should be cleared to 0, MTU3.TGRC and MTU3.TGRA (MTU6.TGRC and MTU6.TGRA) should be set to 1/2 the PWM carrier cycle + 1, and TDDRA (TDDRB) should be set to 1.
Set the respective initial PWM duty values in three buffer registers A (MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD (MTU6.TGRD, MTU7.TGRC, and MTU7.TGRD)).
Set the respective (initial PWM duty – 1) values in three buffer registers B (MTU3.TGRE, MTU4.TGRE, and MTU4.TGRF (MTU6.TGRE, MTU7.TGRE, and MTU7.TGRF)) only when the double buffer function is used.
The values set in the five buffer registers excluding TDDRA (TDDRB) are transferred to the corresponding compare registers as soon as complementary PWM mode is set.
Set MTU4.TCNT (MTU7.TCNT) to 0000h before setting complementary PWM mode.

Table 10.77 Registers and Counters Requiring Initial Setting

Register and Counter	Setting
TOCR1A, TOCR2A, TOCR1B, TOCR2B	PWM output level
MTU3.TGRC MTU6.TGRC	1/2 PWM cycle + dead time Td (1/2 PWM cycle + 1 when dead time generation is disabled by the TDERA or TDERB setting)
TDDRA, TDDRB	Dead time Td (1 when dead time generation is disabled by the TDERA or TDERB setting)
TCBRA, TCBRB	1/2 PWM cycle
MTU3.TGRD, MTU4.TGRC, MTU4.TGRD MTU6.TGRD, MTU7.TGRC, MTU7.TGRD	Initial PWM duty ratio value for each phase
MTU3.TGRE, MTU4.TGRE, MTU4.TGRF MTU6.TGRE, MTU7.TGRE, MTU7.TGRF	Initial PWM duty ratio – 1 value for each phase (only when double buffer function is used)
MTU4.TCNT MTU7.TCNT	0000h

Note: The value set in MTU3.TGRC (MTU6.TGRC) should be the sum of 1/2 the PWM cycle set in TCBRA (TCBRB) and dead time Td set in TDDRA (TDDRB). When dead time generation is disabled by TDERA (TDERB), TGRC should be set to 1/2 the PWM cycle + 1.

(d) PWM Output Level Setting

In complementary PWM mode, the PWM output level is set with bits OLSN and OLSP in timer output control register 1 (TOCR1A or TOCR1B) or bits OLS1P to OLS3P and OLS1N to OLS3N in timer output control register 2 (TOCR2A or TOCR2B).
The output level can be set for each of the three positive phases and three negative phases of 6-phase output.
Complementary PWM mode should be cleared before setting or changing output levels.

(e) Dead Time Setting

In complementary PWM mode, dead time can be set for PWM output.
The dead time is set in the timer dead time data register (TDDRA or TDDRB). The value set in TDDRA (TDDRB) is used as the MTU3.TCNT (MTU6.TCNT) counter start value and creates a non-overlapping interval between MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT). Complementary PWM mode should be cleared before changing the contents of TDDRA (TDDRB).

(f) Dead Time Suppressing

Dead time generation is suppressed by clearing the TDER bit in the timer dead time enable register (TDERA or TDERB) to 0. TDERA (TDERB) can be cleared to 0 only when 0 is written to it after reading TDER = 1.
MTU3.TGRA and MTU4.TGRC (MTU6.TGRA and MTU7.TGRC) should be set to 1/2 PWM cycle + 1 and the timer dead time data register (TDDRA or TDDRB) should be set to 1.
By the above settings, PWM waveforms without dead time can be obtained. Figure 10.51 shows an example of operation without dead time (MTU3 and MTU4).

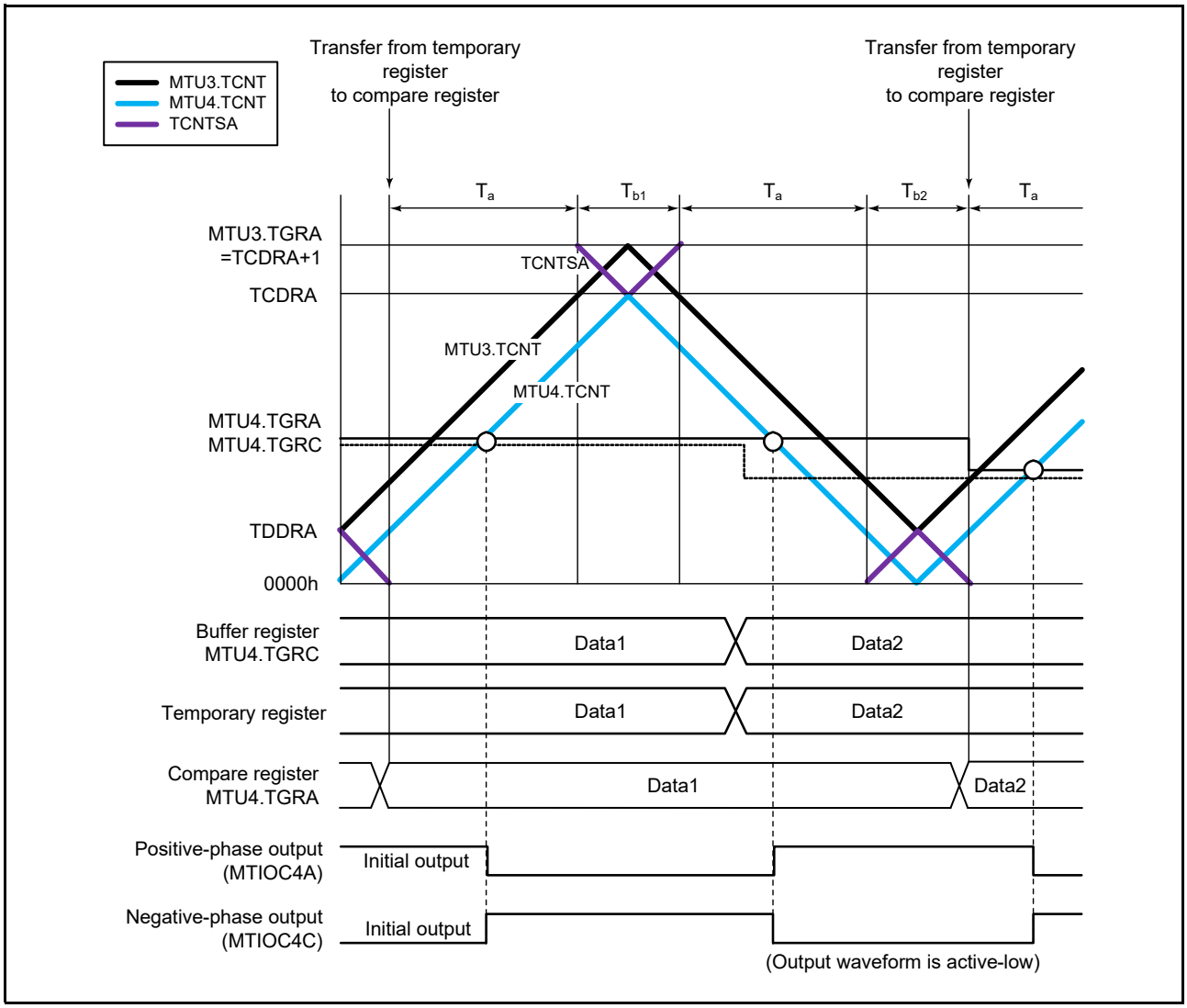


Figure 10.51 Example of Operation without Dead Time (MTU3 and MTU4)

(g) PWM Cycle Setting

In complementary PWM mode, the PWM cycle is set in two registers—MTU3.TGRA (MTU6.TGRA), in which the MTU3.TCNT (MTU6.TCNT) upper limit value is set, and TCDRA (TCDRB), in which the MTU4.TCNT (MTU7.TCNT) upper limit value is set. The settings should be made so as to achieve the following relationship between these two registers:

With dead time: MTU3.TGRA (MTU6.TGRA) setting = TCDRA (TCDRB) setting + TDDRA (TDDRB) setting
Without dead time: MTU3.TGRA (MTU6.TGRA) setting = TCDRA (TCDRB) setting + 1

In addition, the settings should be made so as to achieve the following relationship between the TCDRA (TCDRB) register and the TDDRA (TDDRB) register:

TCDRA (TCDRB) setting > TDDRA (TDDRB) setting × 2 + 2

The MTU3.TGRA and TCDRA (MTU6.TGRA and TCDRB) settings are made by setting values in buffer registers MTU3.TGRC and TCBRA (MTU6.TGRC and TCBRB). When data is written to MTU4.TGRD (MTU7.TGRD) to enable transfers, the values set in MTU3.TGRC and TCBRA (MTU6.TGRC and TCBRB) are transferred simultaneously to the MTU3.TGRA and TCDRA (MTU6.TGRA and TCDRB) with the transfer timing selected with the MTU3.TMDR1.MD[3:0] (MTU6.TMDR1.MD[3:0]) bits.
The new PWM cycle is reflected from the next cycle when data is updated at the crest, or from the current cycle when updated in the trough. Figure 10.52 illustrates the operation when the PWM cycle is updated at the crest.
See the following section, (h), Register Data Updating, for the method of updating the data in each buffer register.

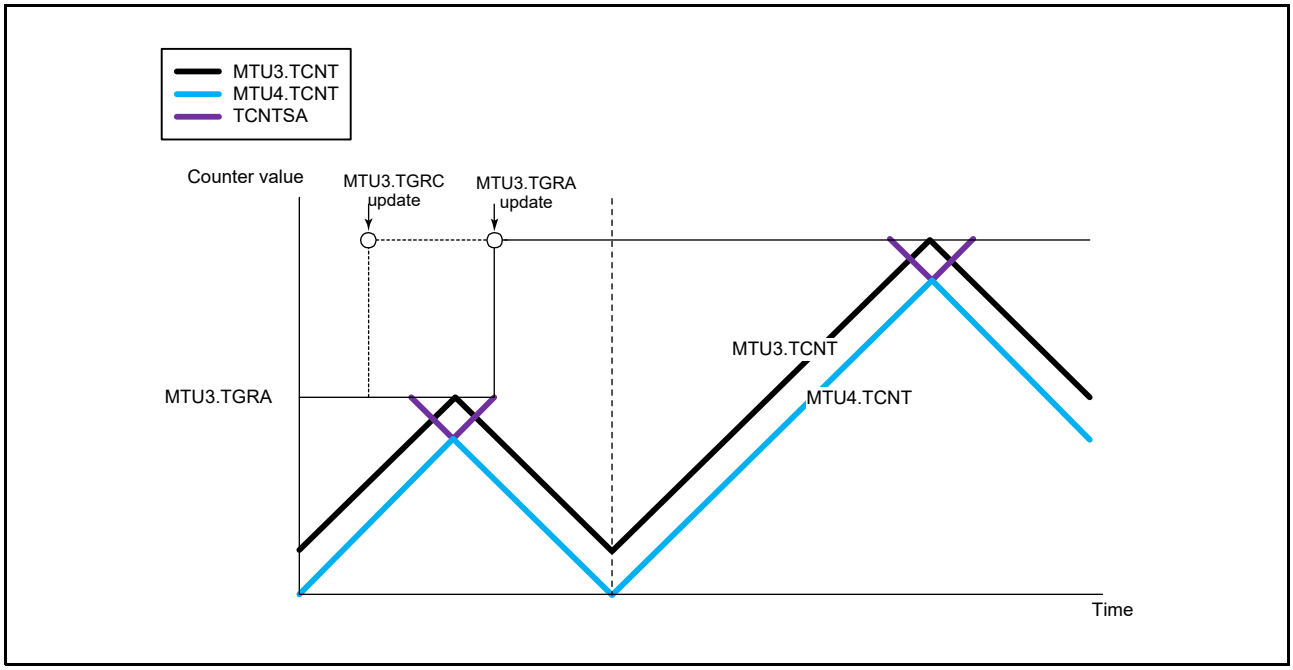


Figure 10.52 Example of PWM Cycle Updating (MTU3 and MTU4)

(h) Register Data Updating

In complementary PWM mode, buffer registers are used to update the data in five compare registers for PWM duty and PWM cycle. The update data can be written to the buffer registers at any time.

There is a temporary register between each of these registers and its buffer register. While subcounter TCNTSA (TCNTSB) is not counting, if buffer register data is updated, the temporary register value also changes. Data is not transferred from buffer registers to temporary registers while TCNTSA (TCNTSB) is counting; in this case, the value written to a buffer register is transferred after TCNTSA (TCNTSB) halts.

The temporary register value is transferred to the compare register at the data update timing set with MTU3.TMDR1.MD[3:0] (MTU6.TMDR1.MD[3:0]) bits. Figure 10.53 shows an example of data updating in complementary PWM mode (MTU3 and MTU4). This example shows the mode in which data is updated at both the counter crest and trough.

When updating buffer register data, be sure to write to MTU4.TGRD (MTU7.TGRD) at the end of the update. Data is transferred from buffer registers to the temporary registers simultaneously for all five registers after the write to MTU4.TGRD (MTU7.TGRD).

Even when not updating all five registers or when not updating the MTU4.TGRD (MTU7.TGRD) data, be sure to write to MTU4.TGRD (MTU7.TGRD) after writing data to the registers to be updated. In this case, the data written to MTU4.TGRD (MTU7.TGRD) should be the same as the data prior to the write operation.

See section 10.3.8 (2) (s) Double Buffer Function in Complementary PWM Mode, for data updating when the double buffer function is used.

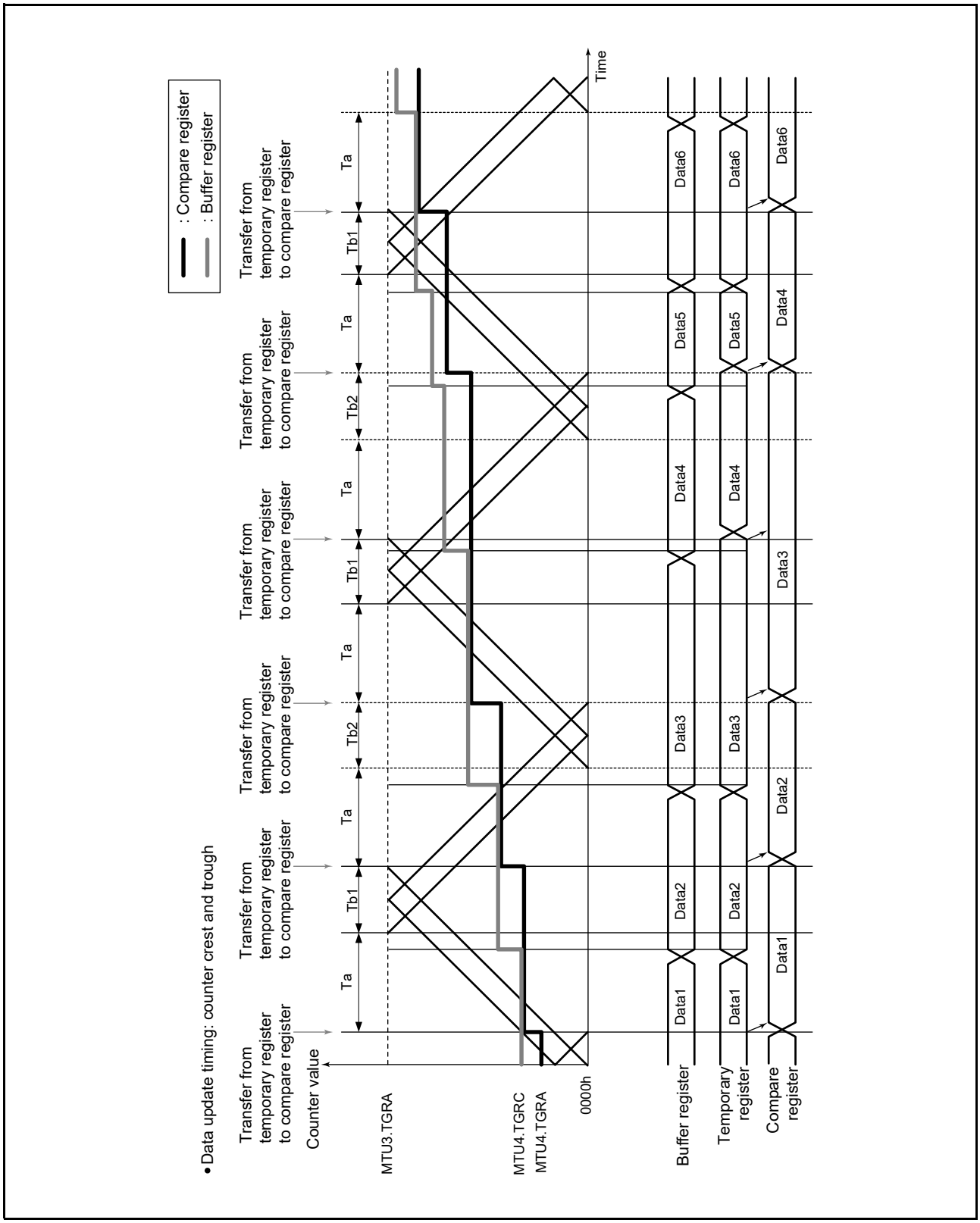


Figure 10.53 Example of Data Updating in Complementary PWM Mode (MTU3 and MTU4)

(i) Initial Output in Complementary PWM Mode

In complementary PWM mode, the initial output is determined by the setting of the OLSN and OLSP bits in the TOCR1A (TOCR1B) register or the OLS1N to OLS3N and OLS1P to OLS3P bits in the TOCR2A register (TOCR2B). This initial output is the non-active level of the PWM output and continues from when complementary PWM mode is set with the MTU3.TMDR1 (MTU6.TMDR1) until MTU4.TCNT (MTU7.TCNT) exceeds the value set in the TDDRA (TDDRB) register. Figure 10.54 shows an example of the initial output in complementary PWM mode.
An example of the waveform when the initial PWM duty ratio value is smaller than the TDDRA (TDDRB) value is shown in Figure 10.55.

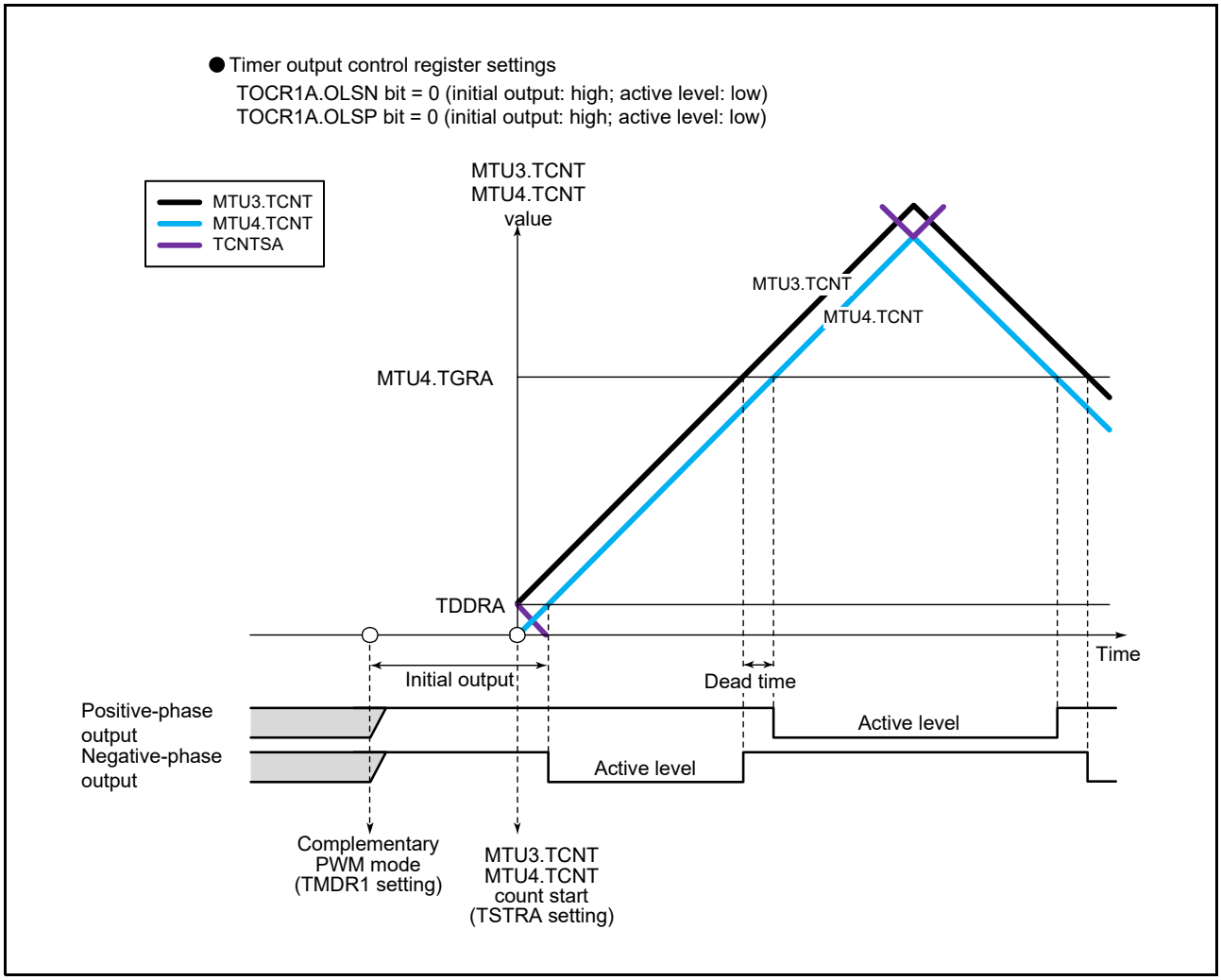


Figure 10.54 Example of Initial Output in Complementary PWM Mode (MTU3 and MTU4) (1)

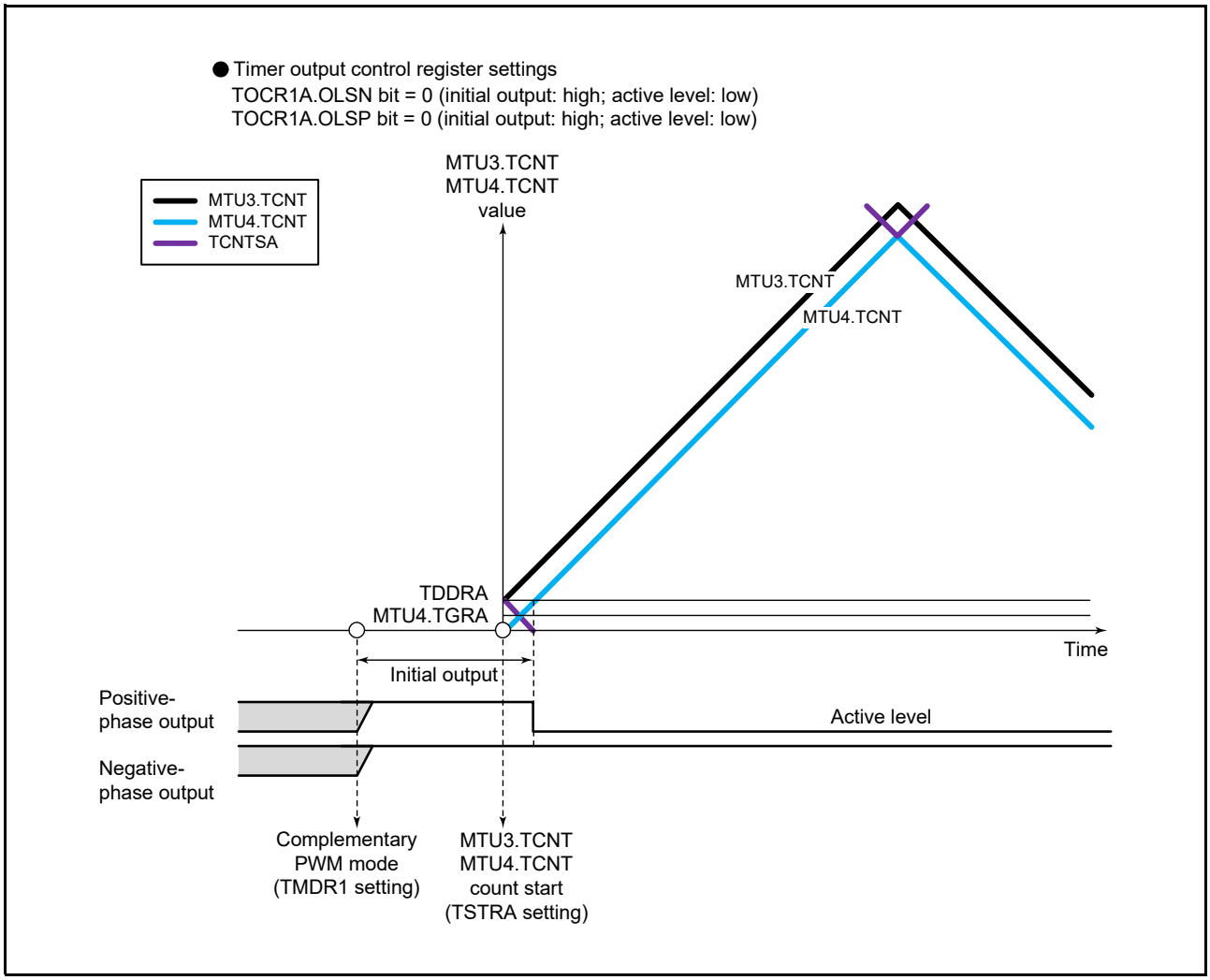


Figure 10.55 Example of Initial Output in Complementary PWM Mode (MTU3 and MTU4) (2)

(j) Method for Generating PWM Output in Complementary PWM Mode

In complementary PWM mode, six phases (three positive and three negative) PWM waveforms can be output. Dead time can be set for PWM waveforms to be output.

A PWM waveform is generated by output of the level selected in the timer output control register in the event of a compare match between a counter and a compare register. While TCNTSA (TCNTSB) is counting, the compare register and temporary register values are simultaneously compared to create consecutive PWM output from 0 to 100% duty ratio. The relative timing of turn-on and turn-off compare match occurrence may vary, but the compare match that turns off each phase takes precedence to secure the dead time and ensure that the positive-phase and negative-phase turn-on times do not overlap. Figure 10.56 to Figure 10.58 show examples of waveform generation in complementary PWM mode.

The positive-phase and negative-phase turn-off timing is generated by a compare match with the counter indicated by a solid line, and the turn-on timing is generated by a compare match with the counter indicated by a dotted line, which operates with a delay equal to the dead time behind the counter indicated by a solid line. In the T1 period, compare match a that turns off the negative phase has the highest priority, and compare matches before a are ignored. In the T2 period, compare match c that turns off the positive phase has the highest priority, and compare matches before c are ignored.

In most cases, compare matches occur in the order a → b → c → d (or c → d → a' → b') as shown in Figure 10.56.

If compare matches deviate from the a → b → c → d order, since the time for which the negative phase is off is shorter than twice the dead time, the positive phase is not turned on. If compare matches deviate from the c → d → a' → b' order, since the time for which the positive phase is off is shorter than twice the dead time, the negative phase is not turned on.

As shown in Figure 10.57, if compare match c follows compare match a before compare match b, compare match b is ignored and the negative phase is turned on by compare match d. This is because turning off the positive phase has priority due to the occurrence of compare match c (positive-phase off timing) before compare match b (positive-phase on timing) (consequently, the waveform does not change because the positive phase goes from off to off).

Similarly, in the example in Figure 10.58, turning off the negative phase has priority due to the occurrence of compare match a' (negative-phase off timing) before compare match d (negative-phase on timing). As a result, the negative phase is not turned on.

Thus, in complementary PWM mode, compare matches at turn-off timings take precedence, and turn-on timing compare matches that occur before a turn-off timing compare match are ignored.

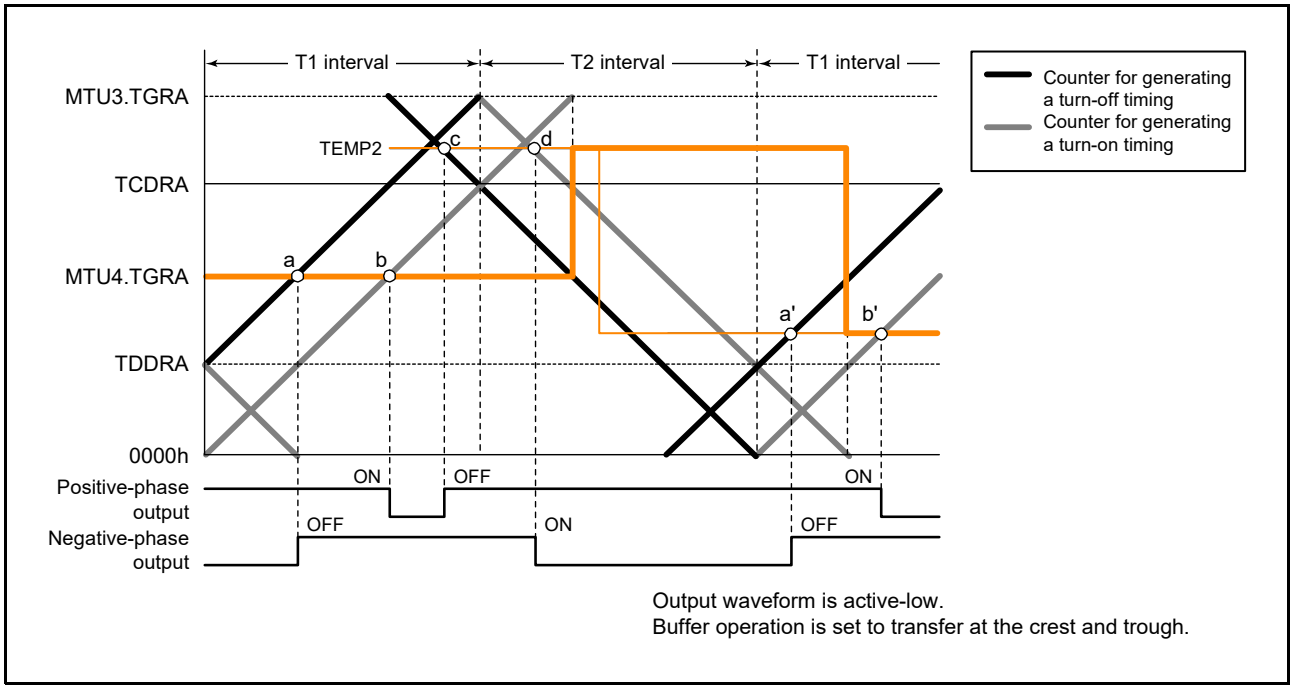


Figure 10.56 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (1)

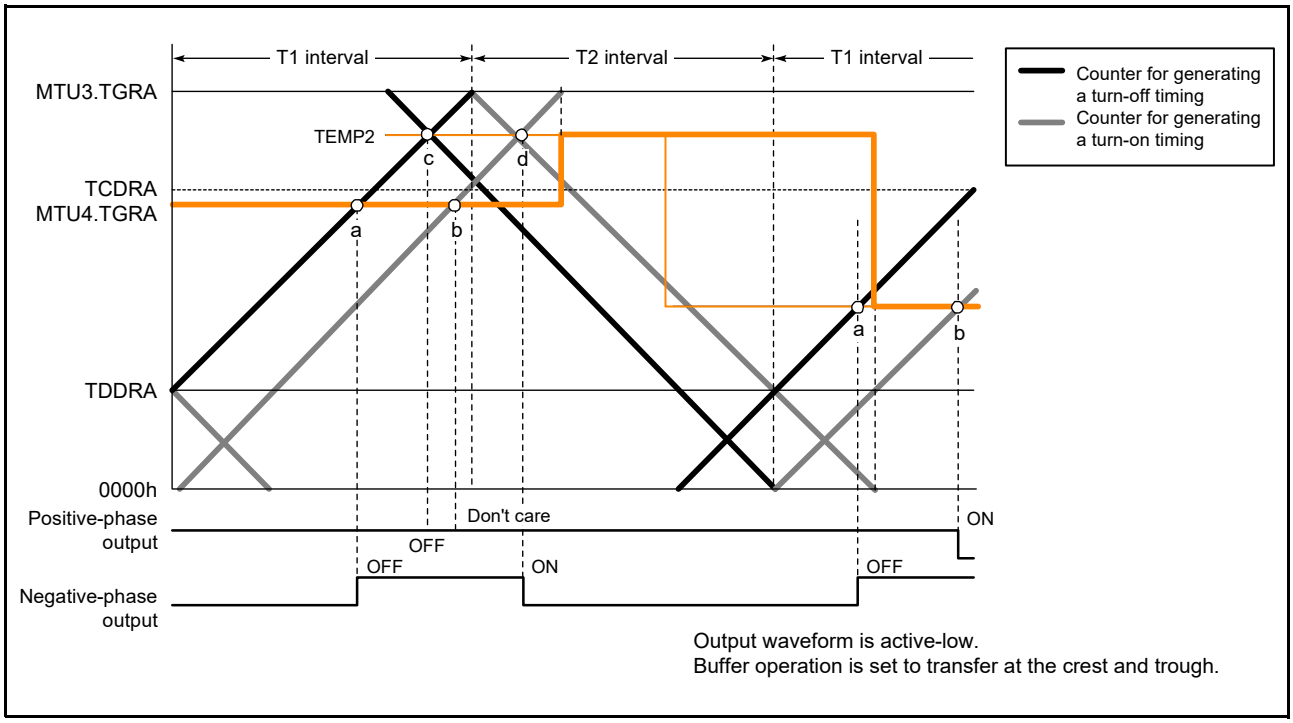


Figure 10.57 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (2)

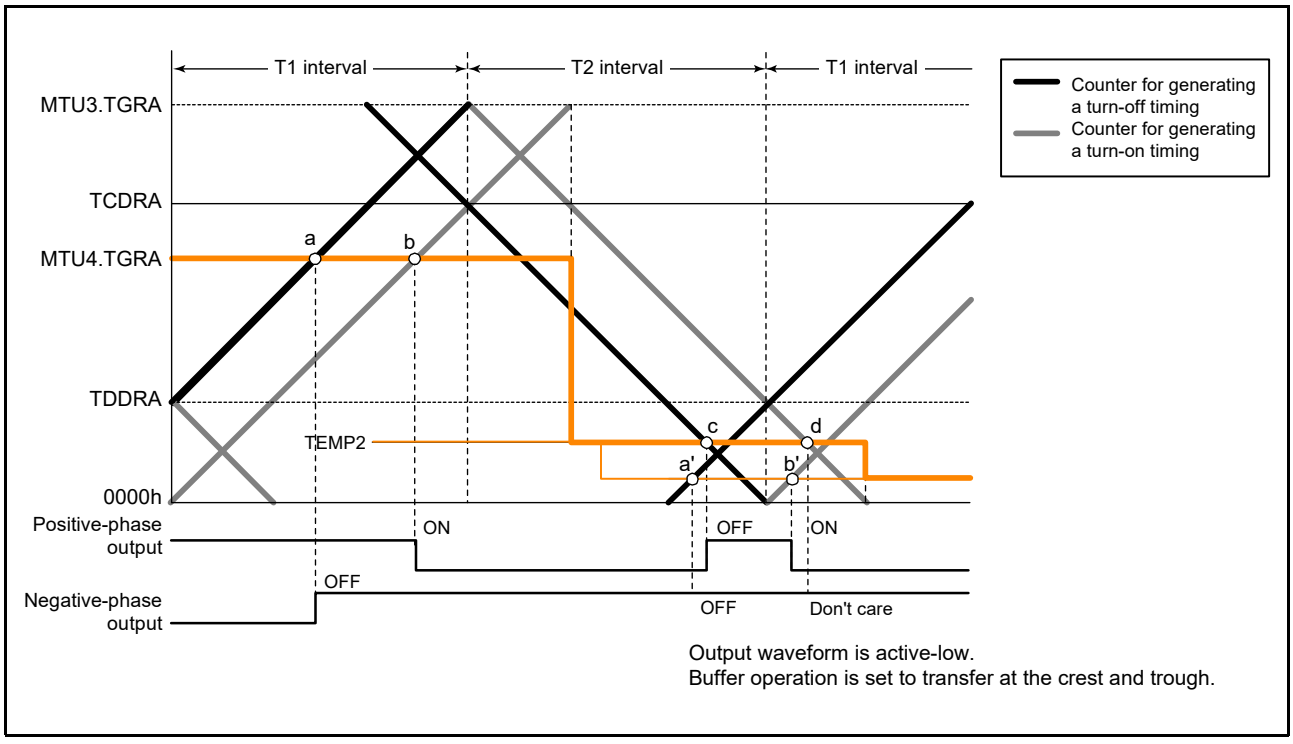


Figure 10.58 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (3)

(k) 0% and 100% Duty Ratio Output in Complementary PWM Mode

In complementary PWM mode, 0% and 100% duty PWM output can be output as required. Figure 10.59 to Figure 10.63 show output examples.

A 100% duty waveform is output when the compare register value is set to 0000h. The waveform in this case has a positive phase with a 100% on-state. A 0% duty waveform is output when the compare register value is set to the same value as MTU3.TGRA (MTU6.TGRA). The waveform in this case has a positive phase with a 100% off-state.

Turn-on and turn-off compare matches occur simultaneously, but if a turn-on compare match and turn-off compare match for the same phase occur simultaneously, both compare matches are ignored and the waveform does not change.

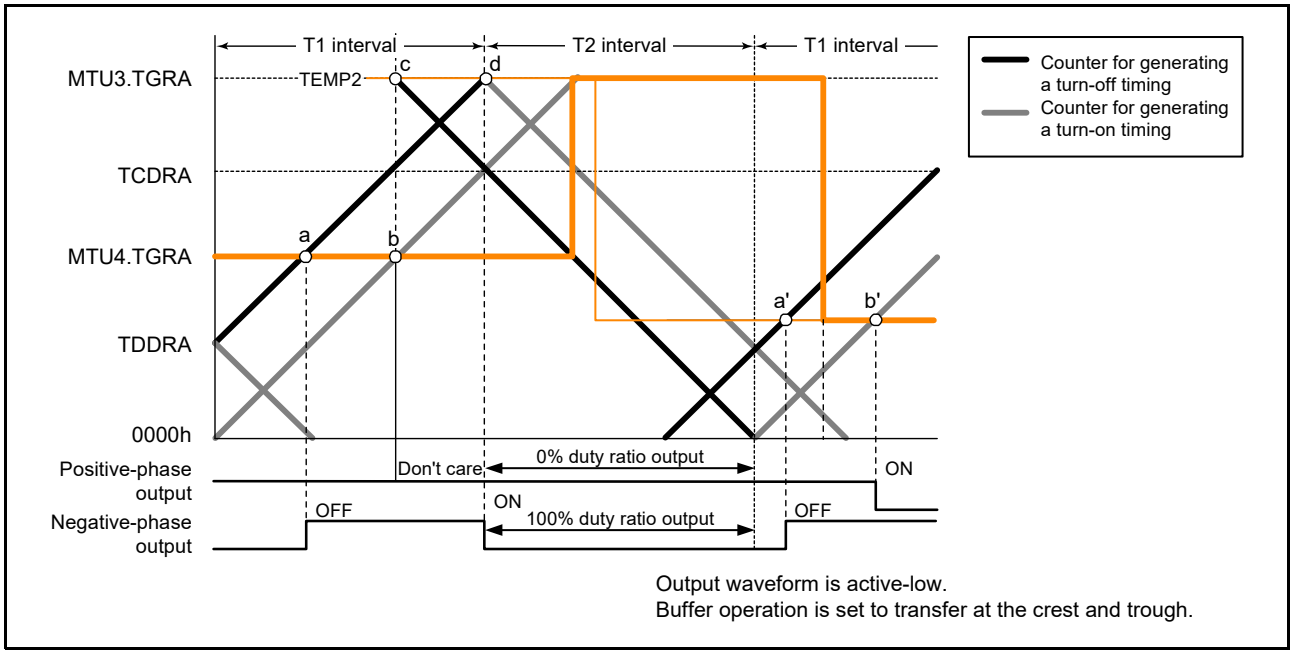


Figure 10.59 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (1)

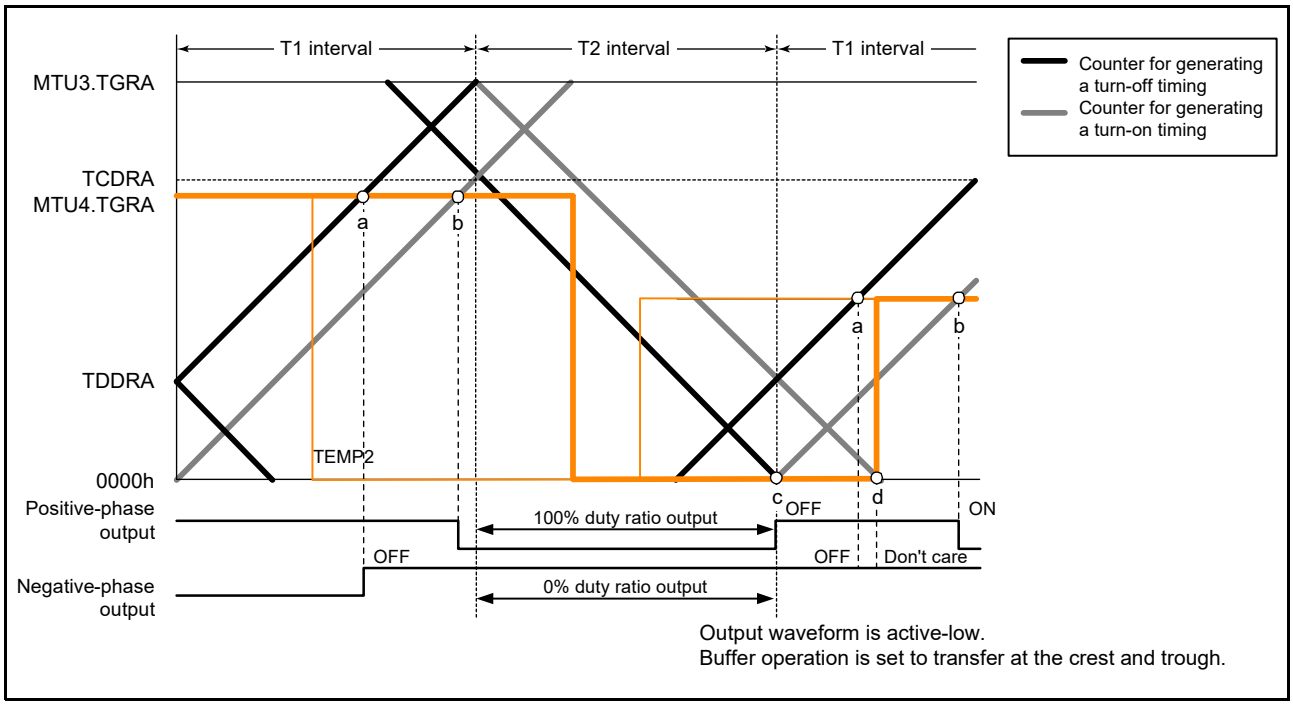


Figure 10.60 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (2)

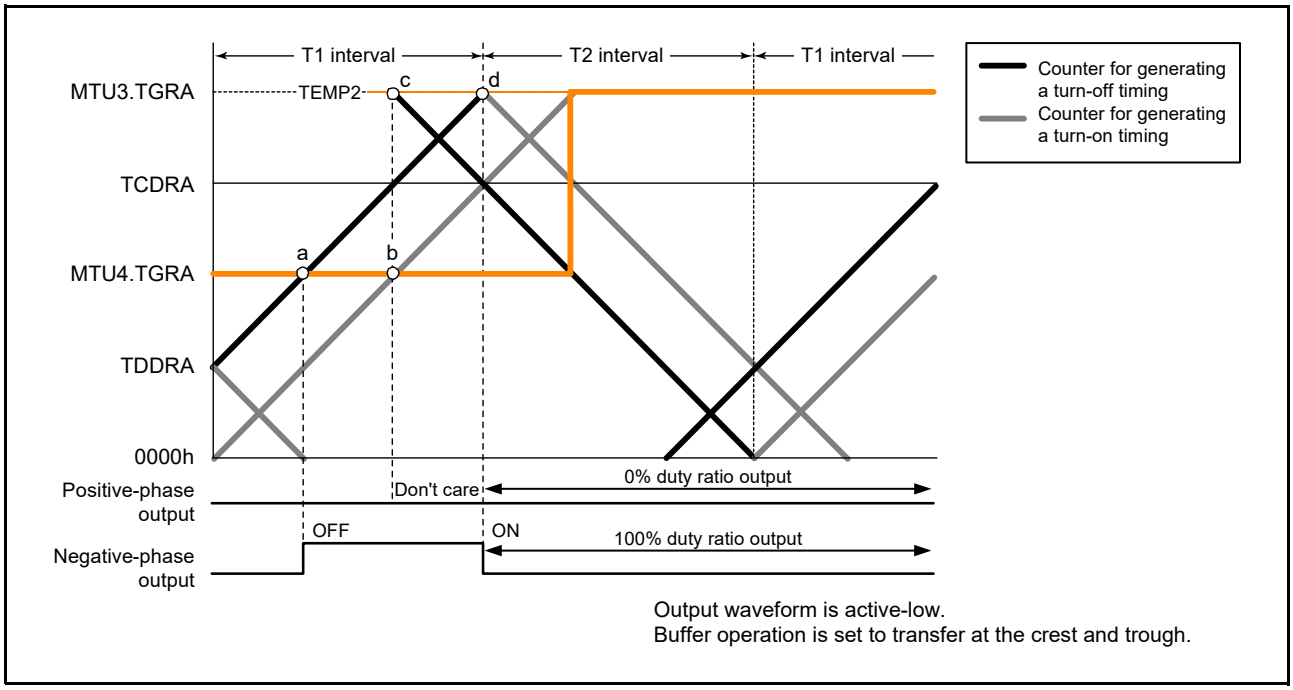


Figure 10.61 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (3)

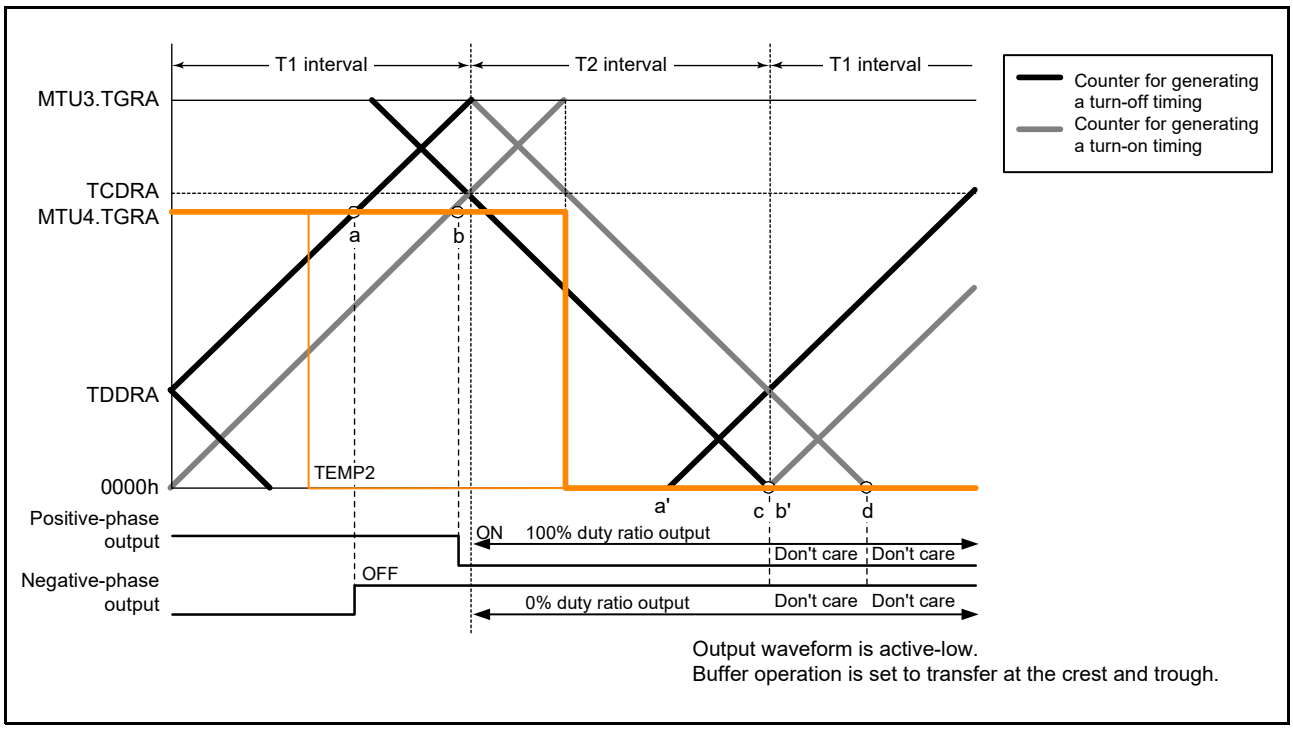


Figure 10.62 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (4)

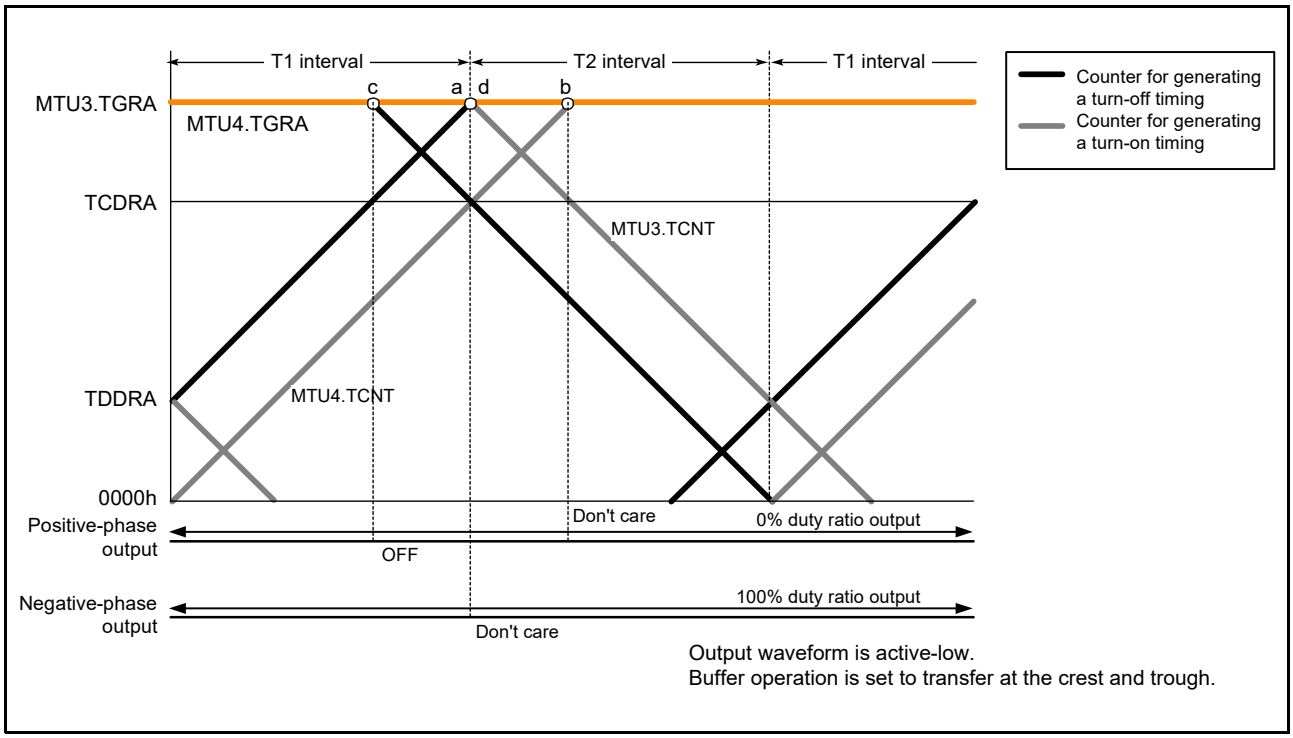


Figure 10.63 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (5)

(l) Toggle Output Synchronized with PWM Cycle

In complementary PWM mode, toggle output from the PWM output pin in synchronization with the PWM cycle can be enabled by setting the PSYE bit in the TOCR1A (TOCR1B) register to 1. An example of a toggle output waveform is shown in Figure 10.64.

This output is toggled by a compare match between MTU3.TCNT and MTU3.TGRA (MTU6.TCNT and MTU6.TGRA) and a compare match between MTU4.TCNT (MTU7.TCNT) and 0000h.

The MTIOC3A (MTIOC6A) pin is assigned for this toggle output. The initial output is high-level output.

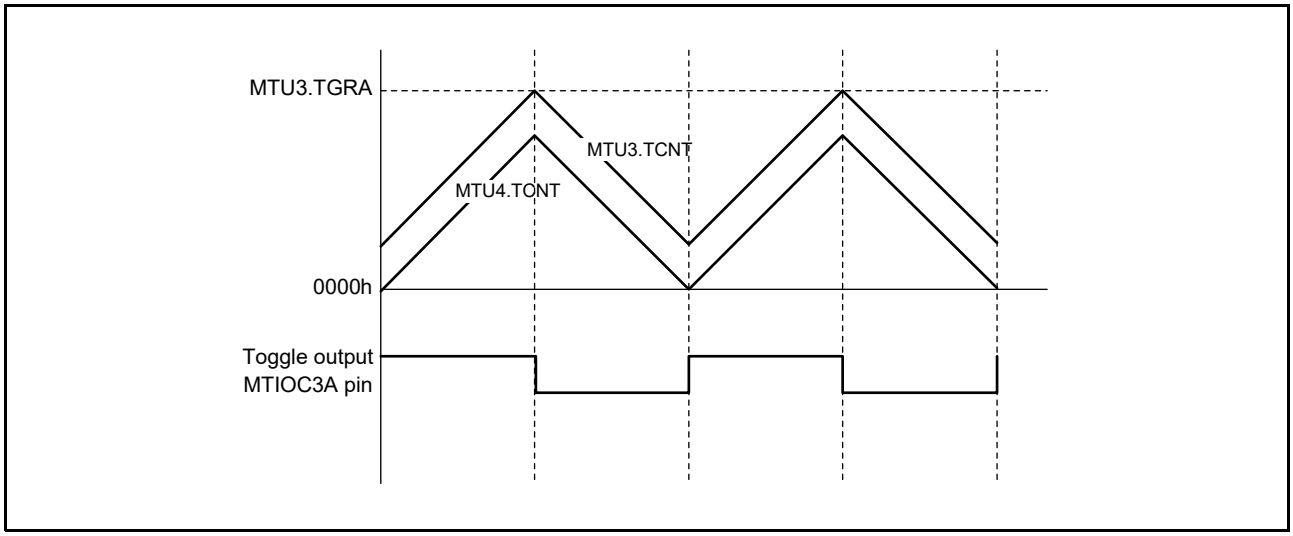


Figure 10.64 Example of Toggle Output Waveform Synchronized with PWM Output (MTU3 and MTU4)

(m) Counter Clearing by Another Channel

In complementary PWM mode, MTU3.TCNT, MTU4.TCNT, and TCNTSA (MTU6.TCNT, MTU7.TCNT, and TCNTSB) can be cleared by another channel source when a mode for synchronization with another channel is specified through the TSYRA (TSYRB) register and synchronous clearing is selected with MTU3.TCR.CCLR[2:0] (MTU6.TCR.CCLR[2:0]) bits.

Figure 10.65 illustrates an example of this operation.

Use of this function enables a counter to be cleared and restarted through an external signal.

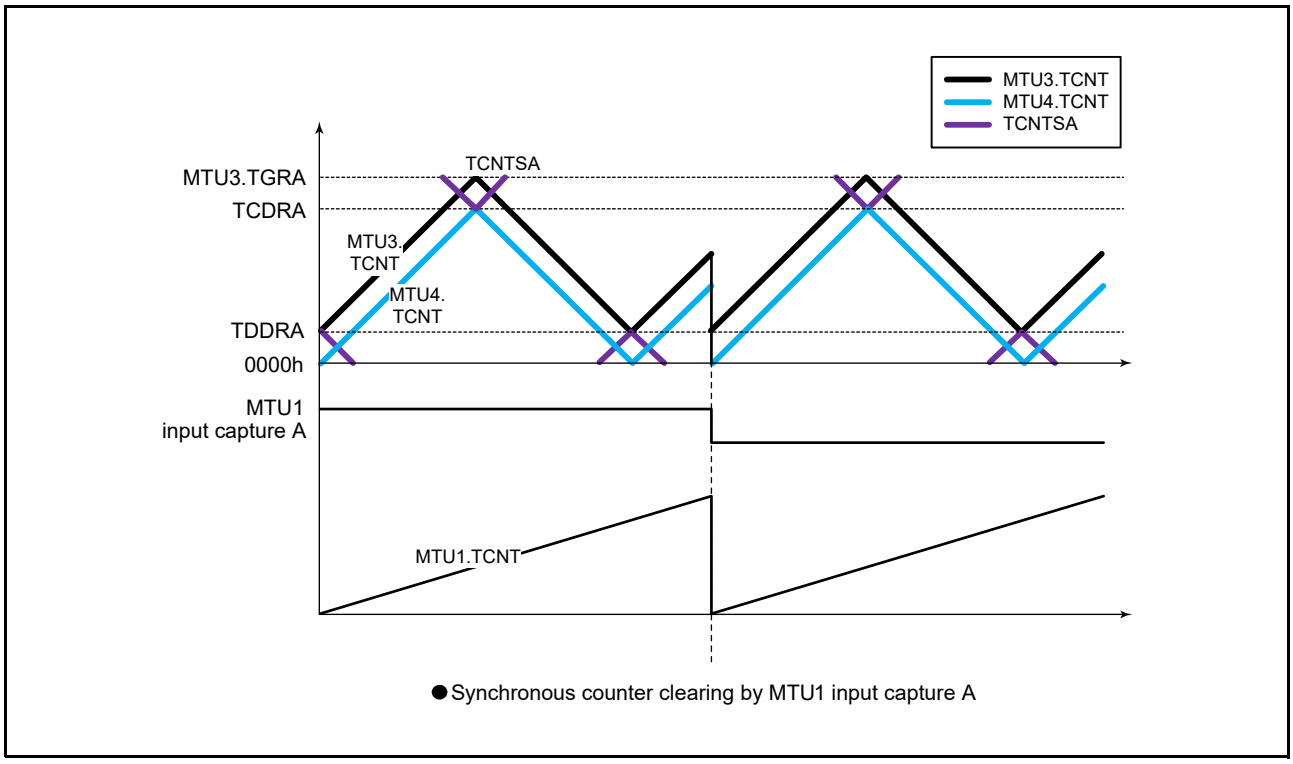


Figure 10.65 Counter Clearing Synchronized with Another Channel (MTU3 and MTU4)

(n) Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Setting the WRE bit in TWCRA (TWCRB) to 1 suppresses initial output when synchronous counter clearing occurs in the Tb interval (Tb2 interval) at the trough in complementary PWM mode and controls abrupt change in duty cycle at synchronous counter clearing.

Initial output suppression through the WRE bit = 1 is applicable only when synchronous clearing occurs in the Tb2 interval as indicated by (10) or (11) in Figure 10.66. When synchronous clearing occurs outside that interval, the initial value specified by the OLSN and OLSP bits in TOCR1A (TOCR1B) is output. Even in the Tb2 interval, if synchronous clearing occurs in the initial value output period (indicated by (1) in Figure 10.66) immediately after the counters start operation, initial value output is not suppressed.

This function can be used in both channel combinations of MTU 3 and MTU4 and MTU6 and MTU7. In MTU3 and MTU4, synchronous clearing in MTU0, MTU1, and MTU2 can cause counter clearing; in MTU6 and MTU7, compare match or input capture generated in MTU0, MTU1, and MTU2 can cause counter clearing.

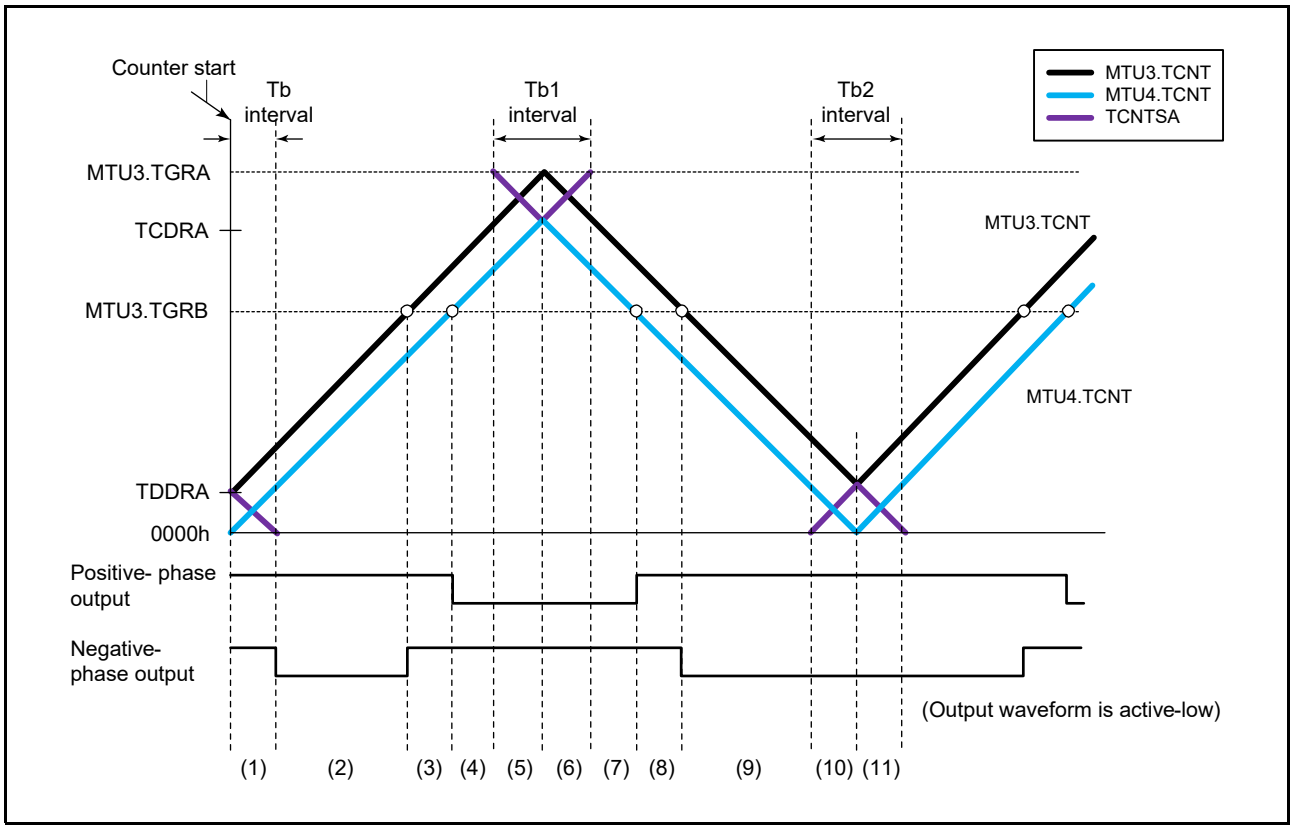


Figure 10.66 Timing for Synchronous Counter Clearing (MTU3 and MTU4)

- Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode.

An example of the procedure for setting output waveform control at synchronous counter clearing in complementary PWM mode is shown in Figure 10.67.

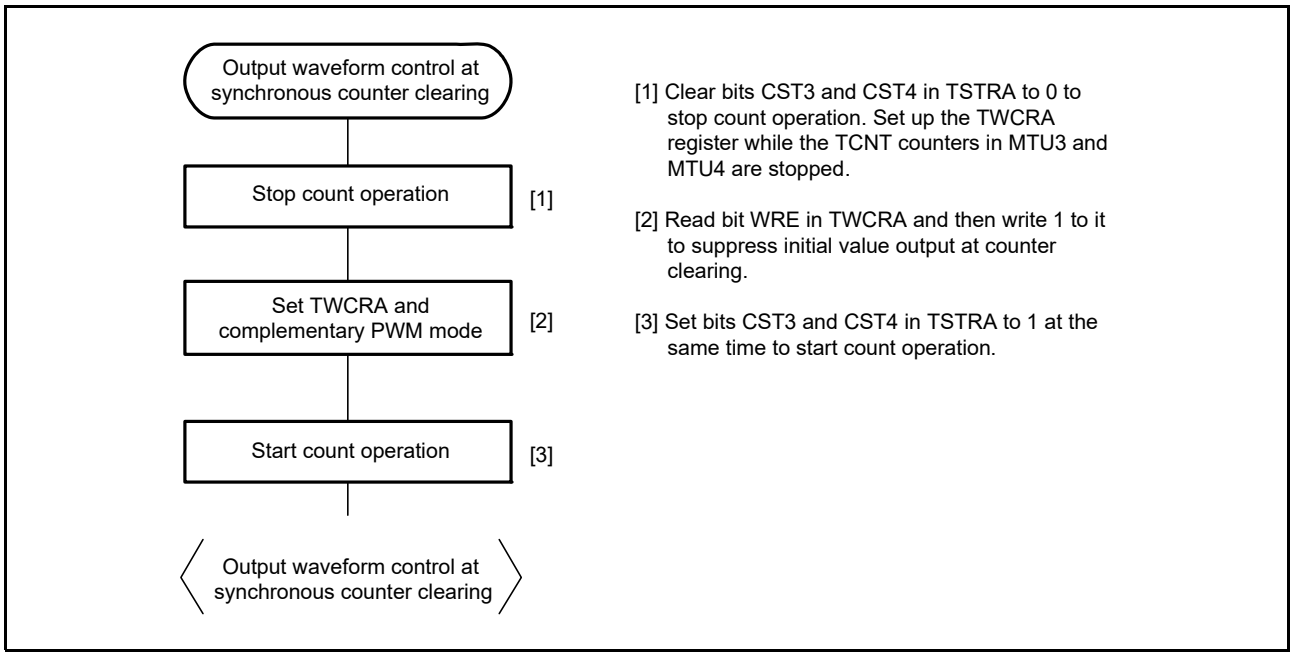


Figure 10.67 Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode (MTU3 and MTU4)

- Examples of Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Figure 10.68 to Figure 10.71 show examples of output waveform control in which MTU3 and MTU4 operate in complementary PWM mode and synchronous counter clearing is generated while the WRE bit in TWCRA is set to 1. In the examples shown in Figure 10.68 to Figure 10.71, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in Figure 10.66, respectively.
In MTU6 and MTU7, these examples are equivalent to the cases when MTU6 and MTU7 operate in complementary PWM mode and synchronous counter clearing is generated while the SCC bit is cleared to 0 and the WRE bit is set to 1 in TWCRB.

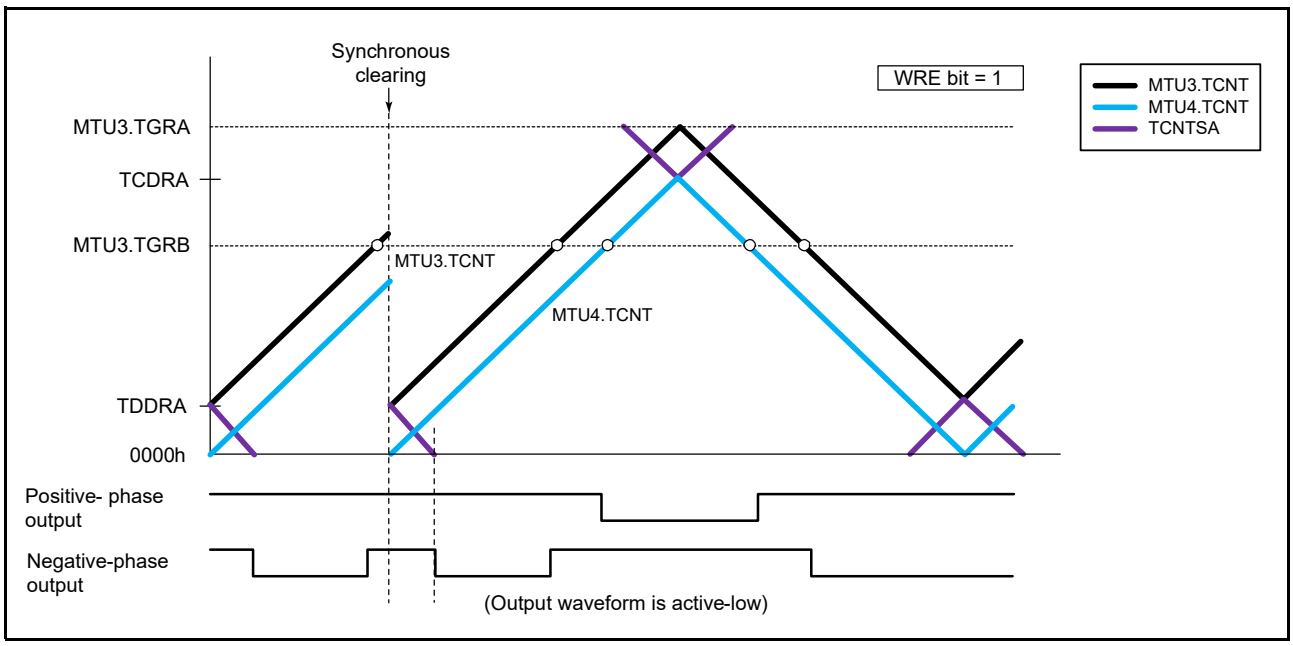


Figure 10.68 Example of Synchronous Clearing in Dead Time during Up-Counting (Timing (3) in Figure 10.66; TWCRA.WRE Bit is 1)

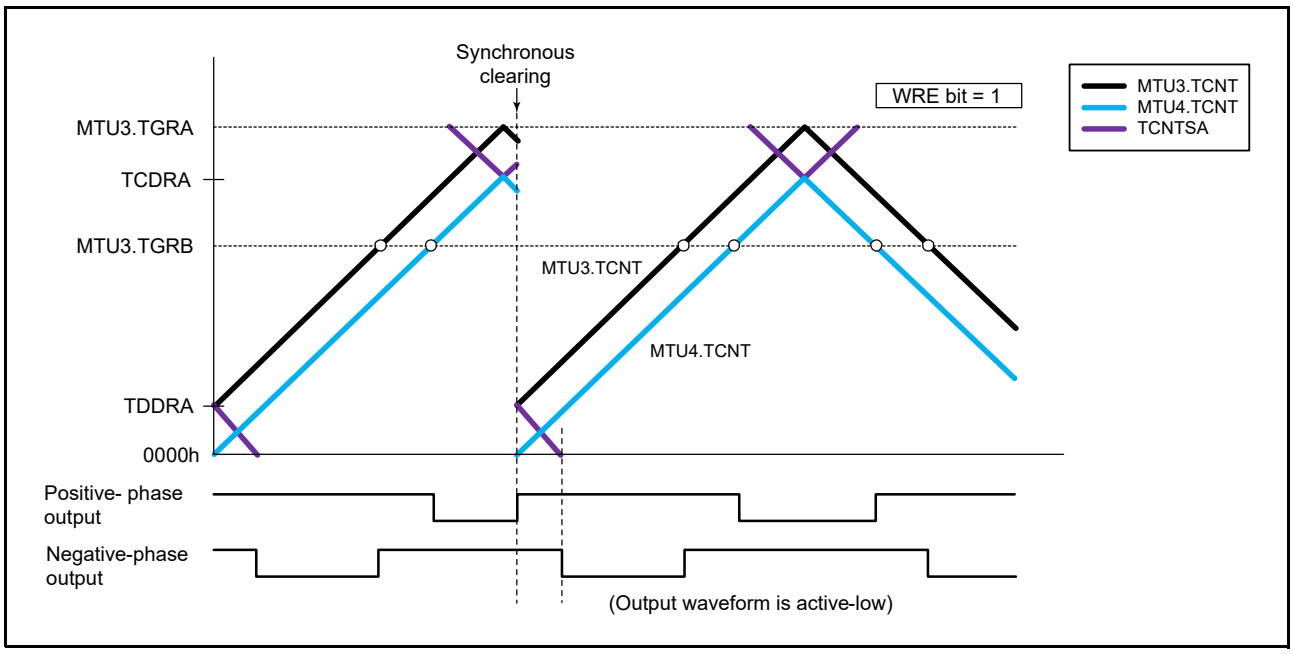


Figure 10.69 Example of Synchronous Clearing in Tb1 interval (Timing (6) in Figure 10.66; TWCRA.WRE Bit is 1)

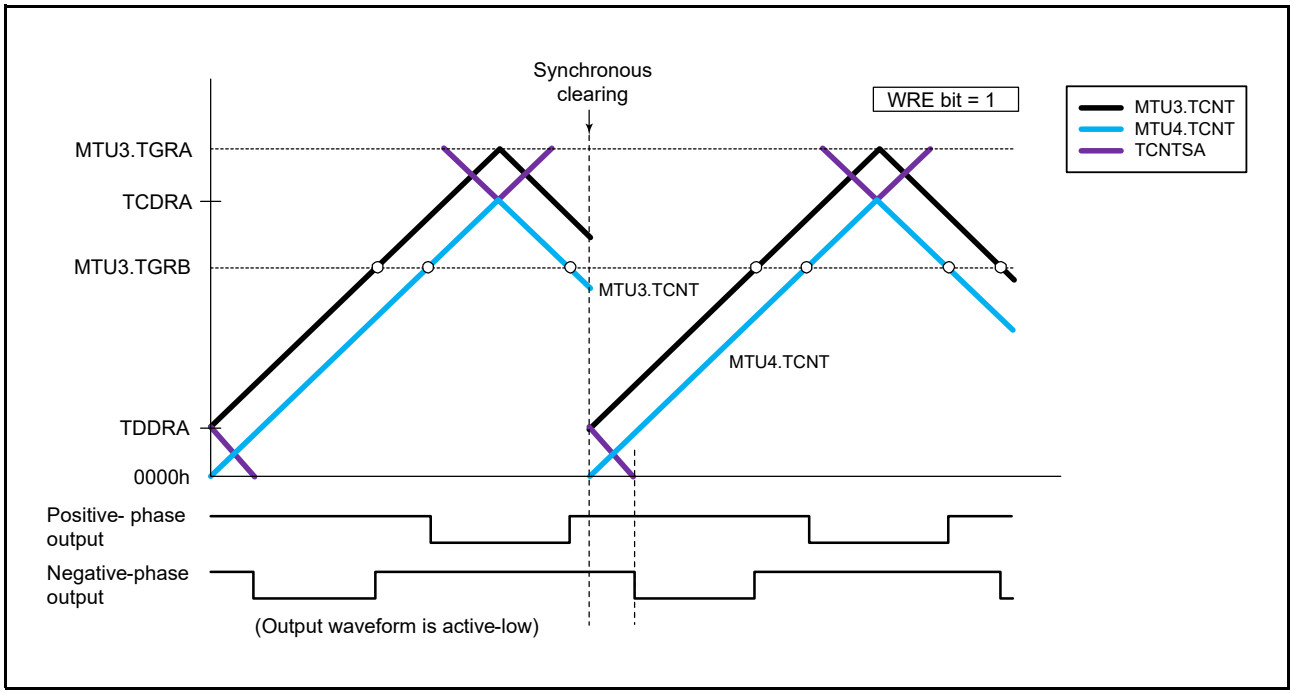


Figure 10.70 Example of Synchronous Clearing in Dead Time during Down-Counting
(Timing (8) in Figure 10.66; TWCRA.WRE Bit is 1)

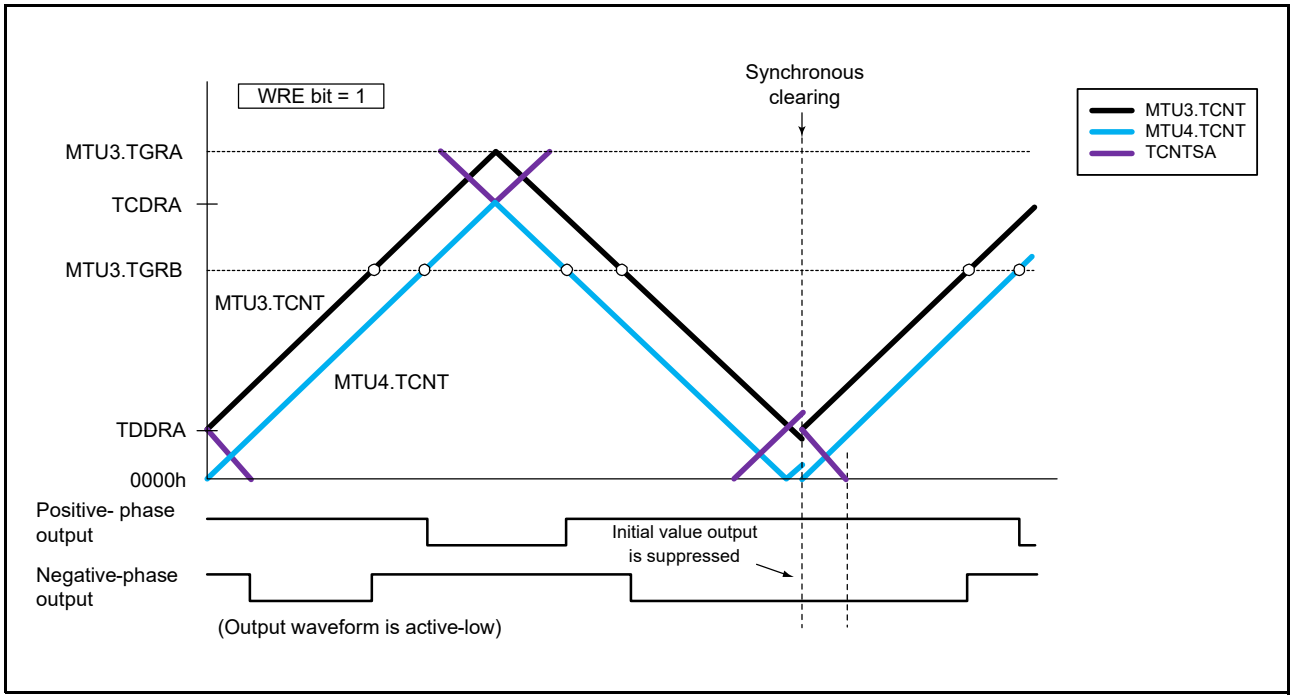


Figure 10.71 Example of Synchronous Clearing in Tb2 interval
(Timing (11) in Figure 10.66; TWCRA.WRE Bit is 1)

(o) Suppressing Synchronous Counter Clearing for MTU0 to MTU2 and MTU6 and MTU7

In MTU6 and MTU7, setting the SCC bit in TWCRB to 1 suppresses synchronous counter clearing caused by MTU0 to MTU2.

Synchronous counter clearing is suppressed only within the interval shown in Figure 10.72. When using this function, MTU6 and MTU7 should be set to complementary PWM mode.

For details of synchronous clearing caused by MTU0 to MTU2, refer to section 10.3.10 (2), Synchronous Counter Clearing for MTU6 and MTU7.

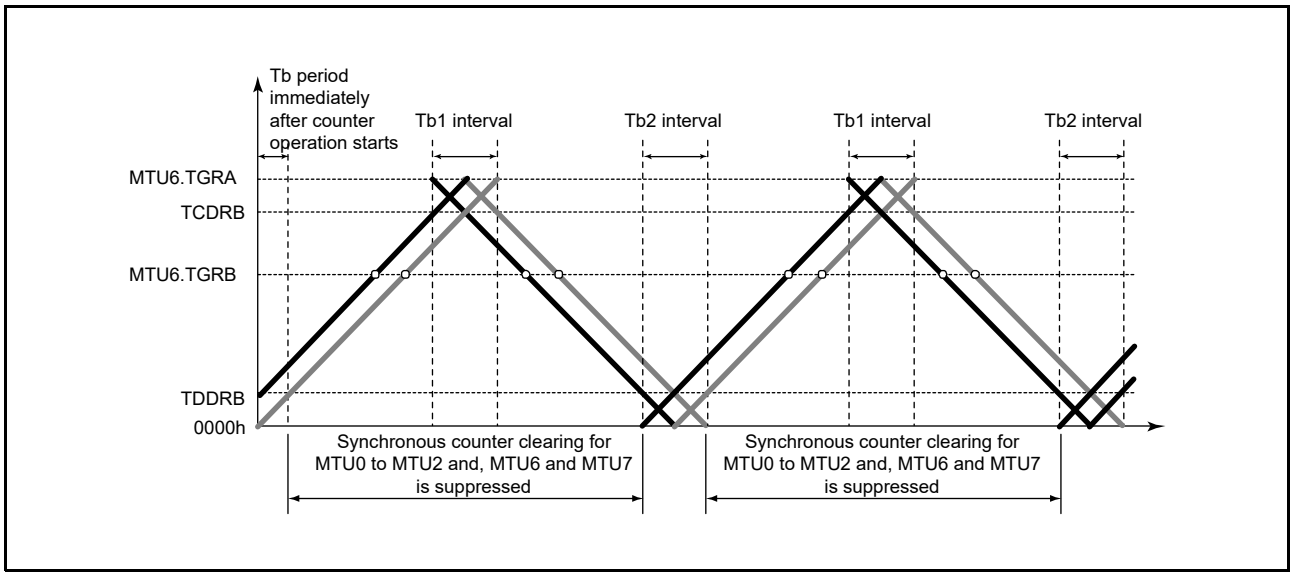


Figure 10.72 Synchronous Clearing-Suppressed Interval Specified by TWCRB.SCC Bit for MTU0, MTU1 and MTU2, and MTU6 and MTU7

- Example of Procedure for Suppressing Synchronous Counter Clearing for MTU0 to MTU2, and MTU6 and MTU7

An example of the procedure for suppressing synchronous counter clearing for MTU0 to MTU2, and MTU6 and MTU7 is shown in Figure 10.73.

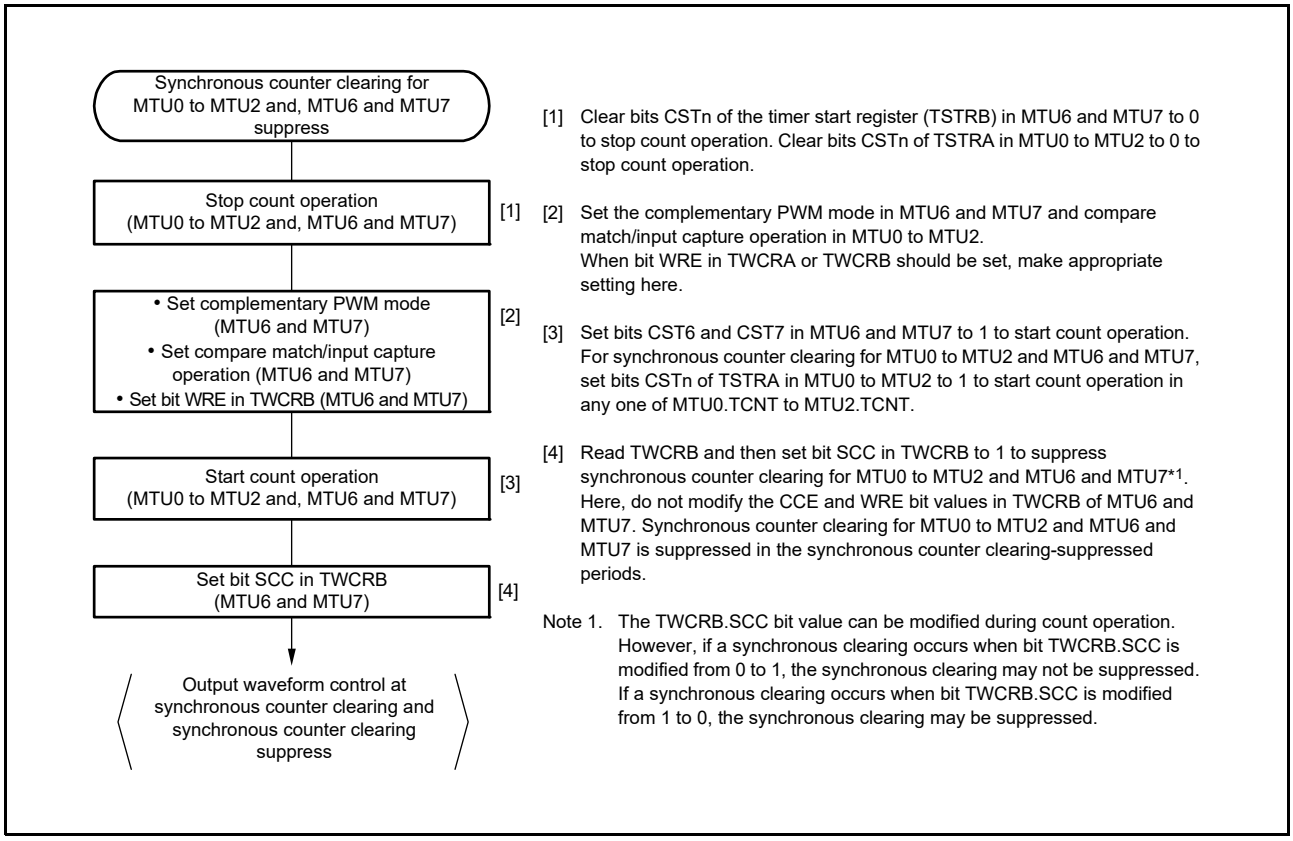


Figure 10.73 Example of Procedure for Suppressing Synchronous Counter Clearing for MTU0 to MTU2, and MTU6 and MTU7

- Examples of Suppression of Synchronous Counter Clearing for MTU 0 to MTU2, and MTU6 and MTU7

Figure 10.74 to Figure 10.77 show examples of operation in which MTU6 and MTU7 operate in complementary PWM mode and synchronous counter clearing for MTU 0 to MTU2, and MTU6 and MTU7 is suppressed by setting the SCC bit in TWCRB in MTU6 and MTU7 to 1. In the examples shown in Figure 10.74 to Figure 10.77, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in Figure 10.66, respectively. In these examples, the WRE bit in TWCRB in MTU6 and MTU7 is set to 1.

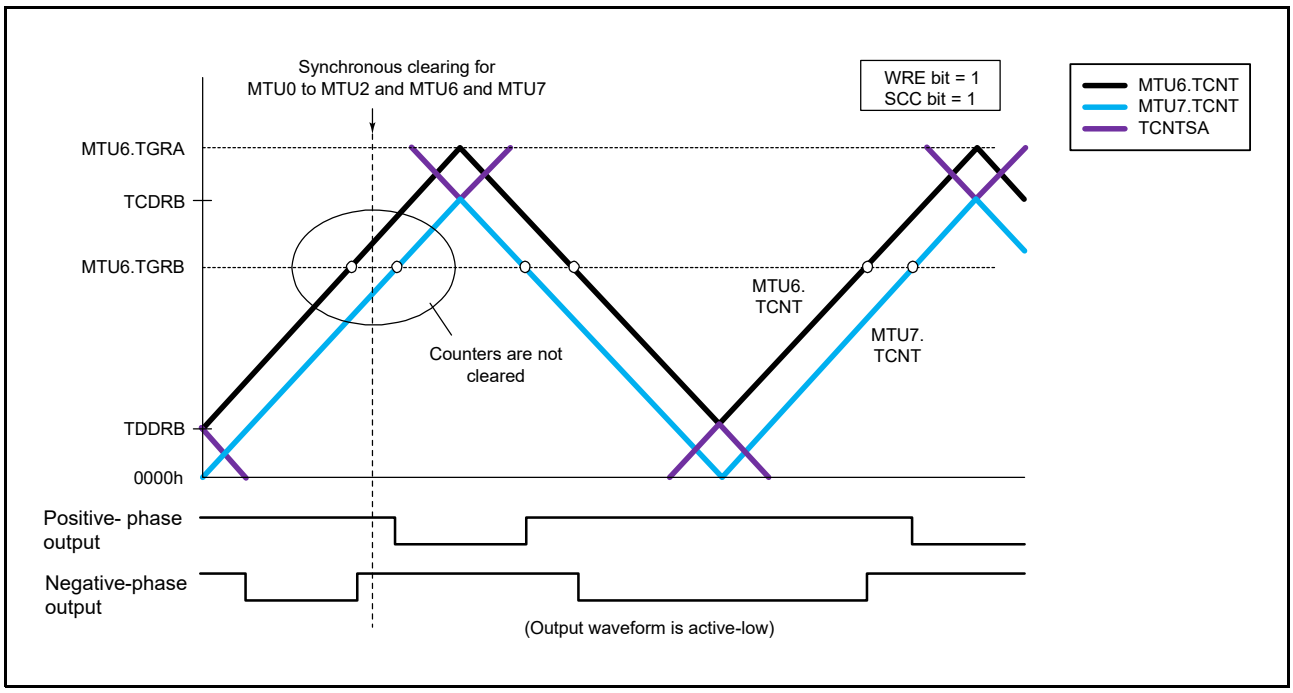


Figure 10.74 Example of Synchronous Clearing in Dead Time during Up-Counting
(Timing (3) in Figure 10.66; TWCRB.WRE Bit is 1 and SCC Bit is 1 in MTU6 and MTU7)

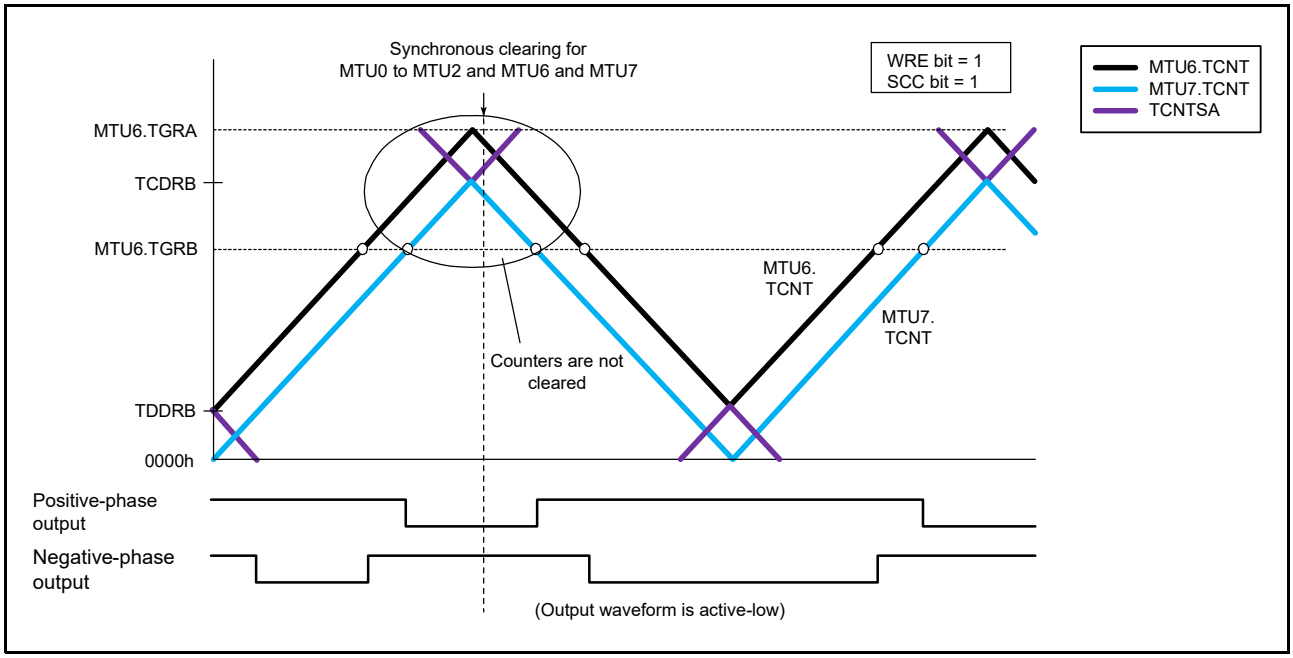


Figure 10.75 Example of Synchronous Clearing in Tb1 interval
(Timing (6) in Figure 10.66; TWCRB.WRE Bit is 1 and SCC Bit is 1 in MTU6 and MTU7)

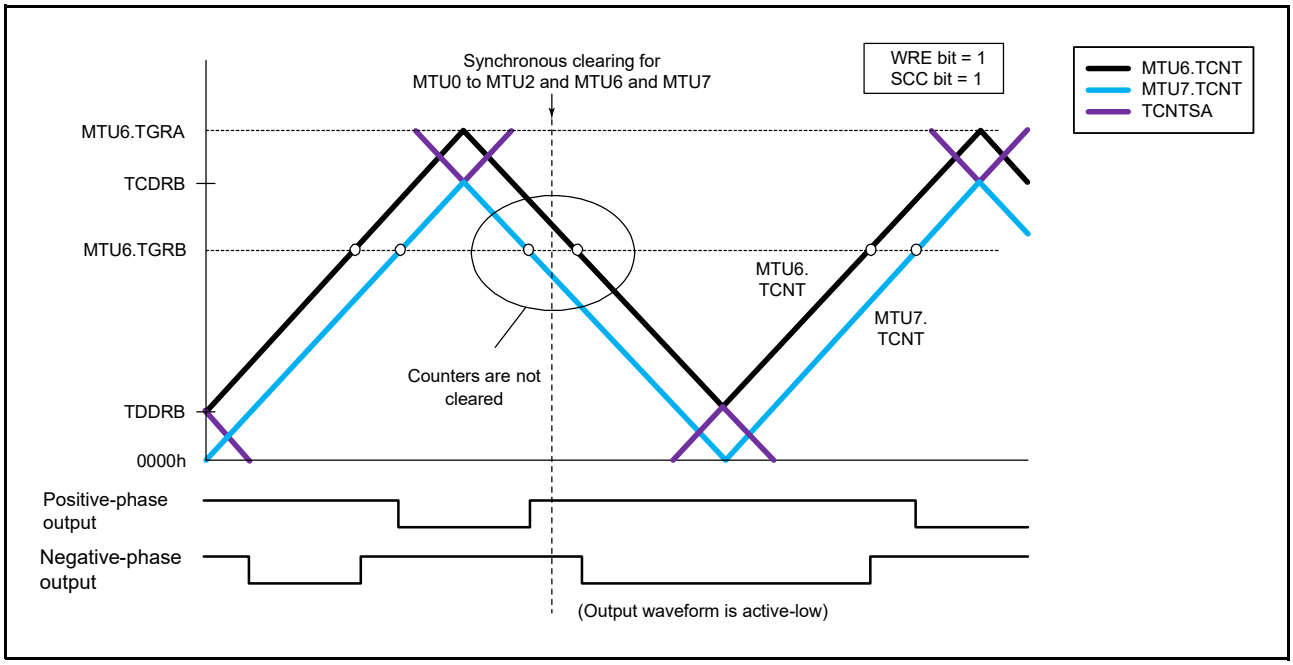


Figure 10.76 Example of Synchronous Clearing in Dead Time during Down-Counting
(Timing (8) in Figure 10.66; TWCRB.WRE Bit is 1 and SCC Bit is 1 in MTU6 and MTU7)

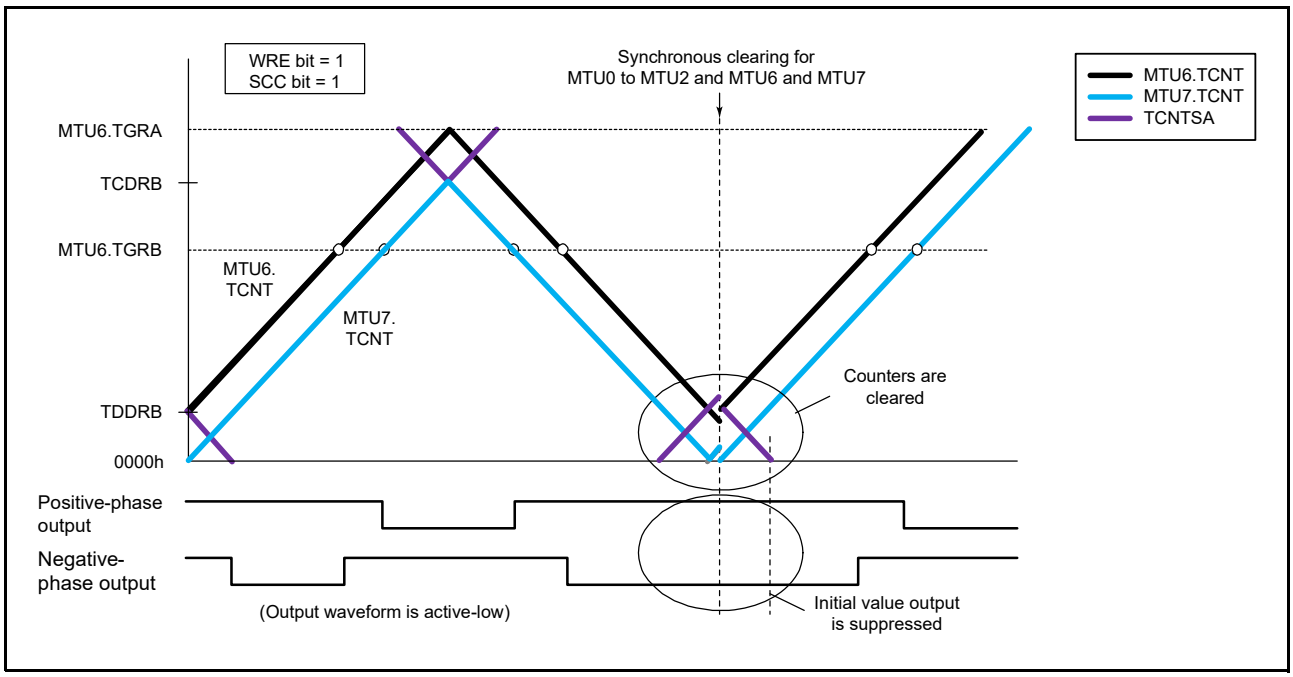


Figure 10.77 Example of Synchronous Clearing in Tb2 interval
(Timing (11) in Figure 10.66; TWCRB.WRE Bit is 1 and SCC Bit is 1 in MTU6 and MTU7)

(p) Counter Clearing by MTU3.TGRA (MTU6.TGRA) Compare Match

In complementary PWM mode, MTU3.TCNT, MTU4.TCNT, and TCNTSA (MTU6.TCNT, MTU7.TCNT, and TCNTSB) can be cleared by MTU3.TGRA (MTU6.TGRA) compare match when the TWCRA.CCE (TWCRB.CCE) bit.

Figure 10.78 illustrates an operation example.

Note 1. Use this function only in complementary PWM mode 1 (transfer at crest).
Note 2. Do not specify synchronous clearing by another channel (do not set 1 in the SYNC0 to SYNC4 or SYNC6 to SYNC7 bits in the timer synchronous register (TSYRA or TSYRB) and the CEOA to CEOD or CEIA to CEID bits in TSYCR).
Note 3. Do not set the PWM duty value to 0000h.
Note 4. Do not set the PSYE bit in timer output control register 1 (TOCR1A or TOCR1B) to 1.

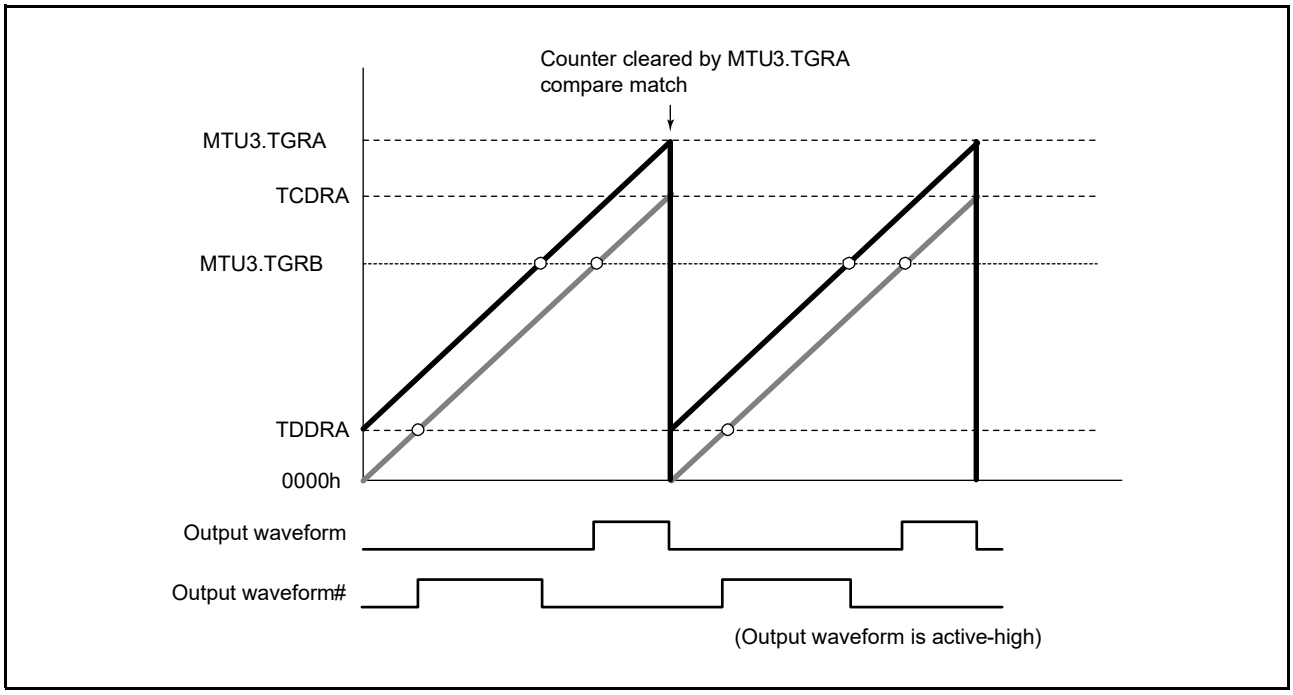


Figure 10.78 Example of Counter Clearing Operation by MTU3.TGRA Compare Match

(q) Example of Waveform Output for Driving AC Synchronous Motor (Brushless DC Motor)

In complementary PWM mode when MTU3 and MTU4 are used, a brushless DC motor can easily be controlled using TGCRA. Figure 10.79 to Figure 10.82 show examples of brushless DC motor driving waveforms created using TGCRA.

To switch the output phases for a 3-phase brushless DC motor by means of external signals detected with a Hall element, etc., clear the TGCRA.FB bit to 0. In this case, the external signals indicating the magnetic pole position should be input to pins MTIOC0A, MTIOC0B, and MTIOC0C in MTU0 (make appropriate settings with the port mode registers (PMR) and Pmn pin function control register (PmnPFS) of the GPIO ports). When an edge is detected at pin MTIOC0A, MTIOC0B, or MTIOC0C, the output on/off state is switched automatically.

When the TGCRA.FB bit is 1, the output on/off state is switched when the UF, VF, or WF bit in TGCRA is cleared to 0 or set to 1.

The driving waveforms are output from the 6-phase PWM output pins for complementary PWM mode.

With this 6-phase output, while the output is turned on, chopping output is available through complementary PWM mode output function by setting the N bit or P bit in TGCRA to 1. When the N bit or P bit is 0, the level output is selected.

The active level of the 6-phase output (on output level) can be set with the TOCR1A.OLSN and TOCR1A.OLSP bits regardless of the setting of the N and P bits.

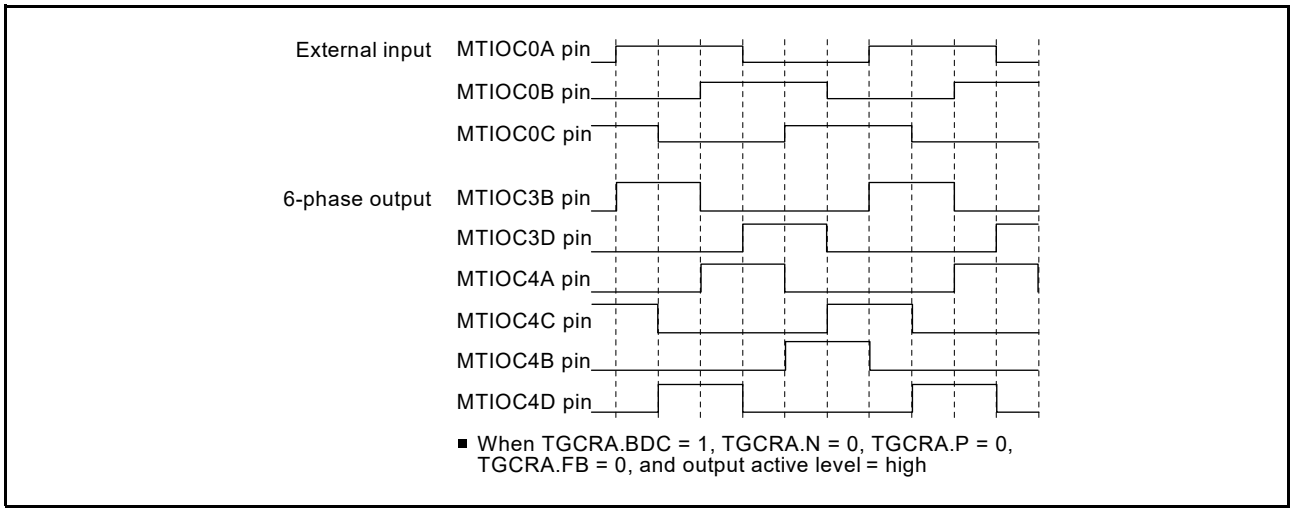


Figure 10.79 Example of Output Phase Switching by External Input (1)

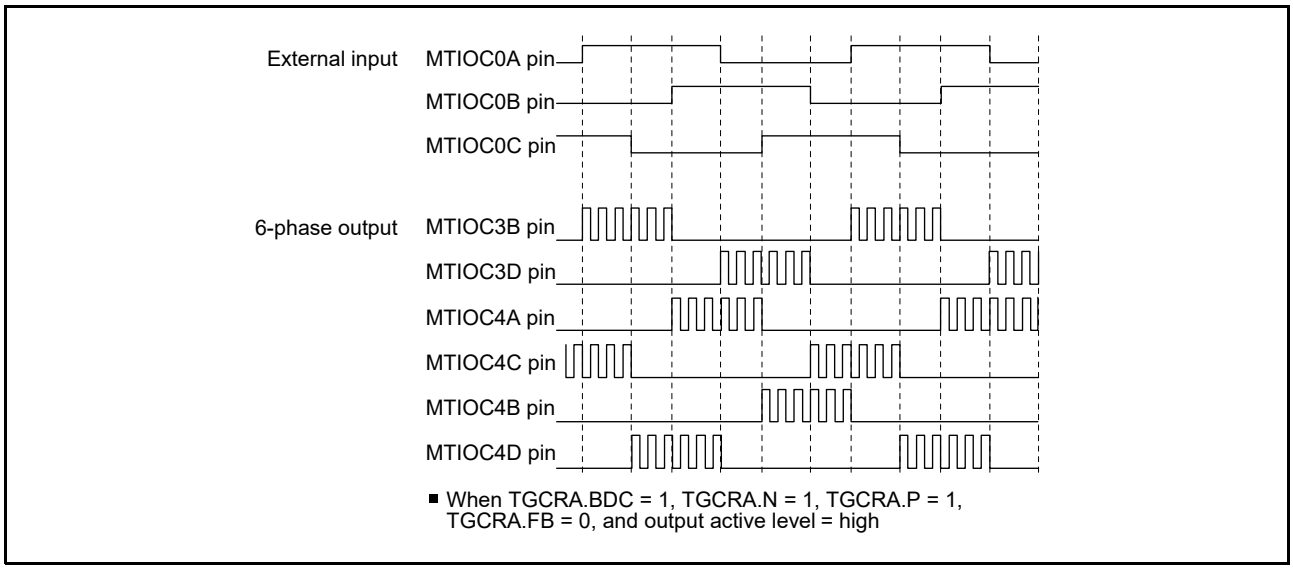


Figure 10.80 Example of Output Phase Switching by External Input (2)

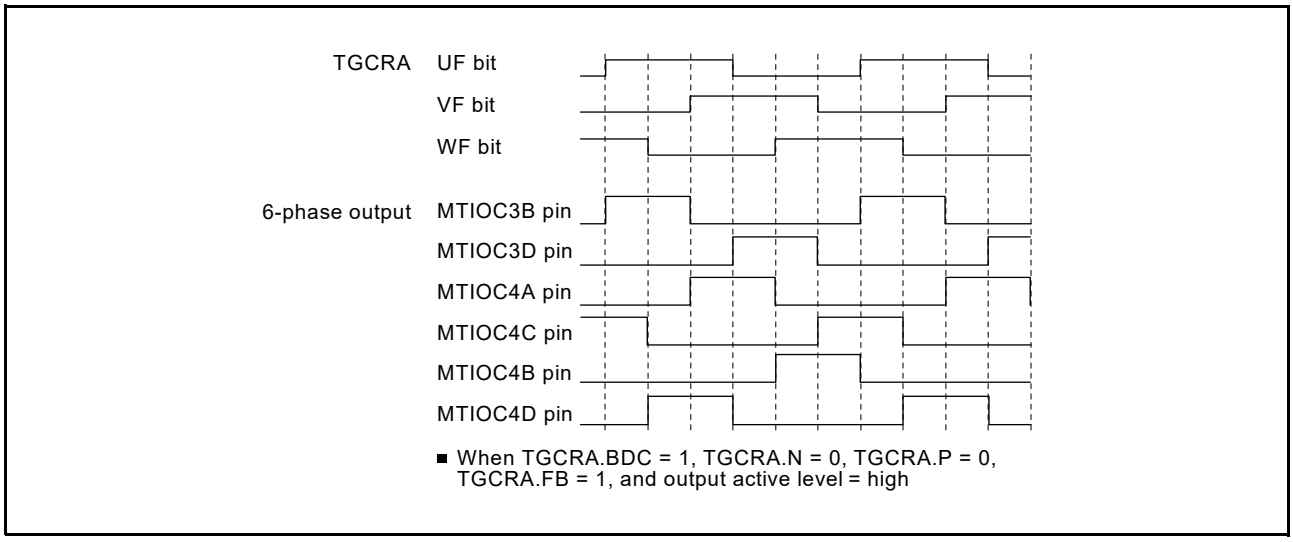


Figure 10.81 Example of Output Phase Switching through UF, VF, and WF Bit Settings (1)

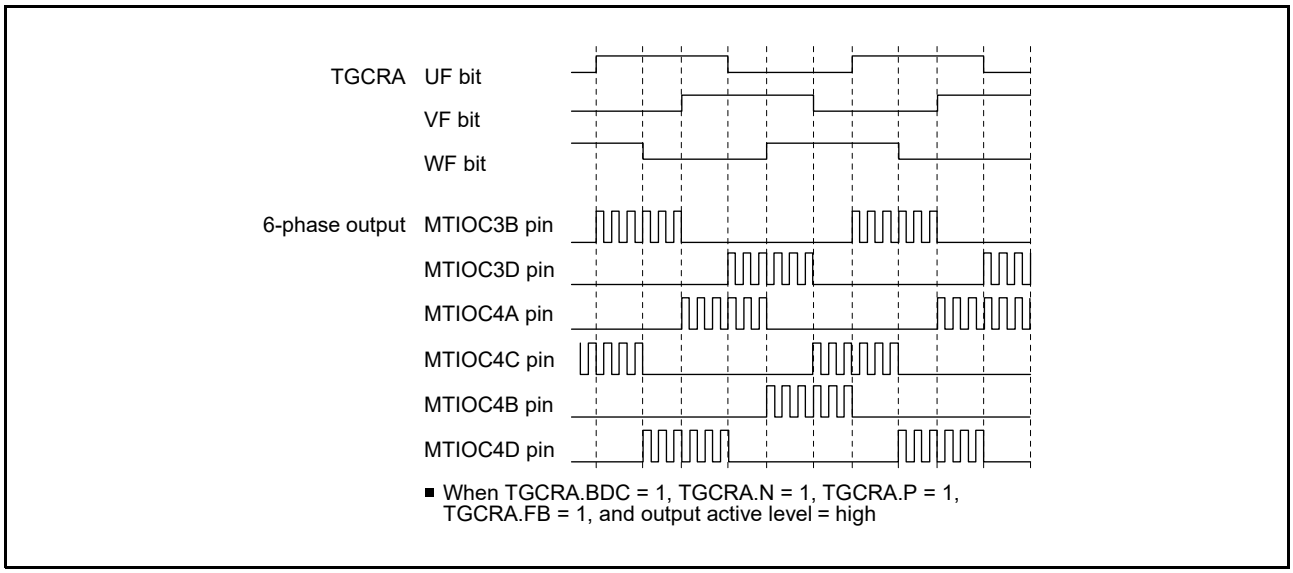


Figure 10.82 Example of Output Phase Switching through UF, VF, and WF Bit Settings (2)

(r) A/D Converter Start Request Setting

In complementary PWM mode, an A/D converter start request can be issued using MTU3.TGRA (MTU6.TGRA) compare match, MTU4.TCNT (MTU7.TCNT) underflow (trough), or compare match on a channel other than MTU3 and MTU4 (MTU6 and MTU7).
When start requests using MTU3.TGRA (MTU6.TGRA) compare match are specified, A/D conversion can be started at the crest of the MTU3.TCNT (MTU6.TCNT) count.
A/D converter start requests can be specified by setting the TIER.TTGE bit. To issue an A/D converter start request at an MTU4.TCNT (MTU7.TCNT) underflow (trough), set the MTU4.TIER.TTGE2 (MTU7.TIER.TTGE2) bit to 1.

(s) Double Buffer Function in Complementary PWM Mode

In complementary PWM mode 3 (transfer at the crest and trough), the PWM output setting resolution can be improved from ±2 to ±1 by setting the TMDR2A.DRS (TMDR2B.DRS) bit to 1.
When setting buffer registers A (TGRD in MTU3, TGRC and TGRD in MTU4, TGRD in MTU6, and TGRC and TGRD in MTU7), set also buffer registers B (TGRE in MTU3, TGRE and TGRF in MTU4, TGRE in MTU6, and TGRE and TGRF in MTU7) at the same time. Each buffer register B should be set to the buffer register A value or (buffer register A value – 1). For details of the setting procedure, refer to section 10.3.8 (1), Example of Complementary PWM Mode Setting Procedure.

Note: When a buffer register B is set to the buffer register A value, symmetric PWM waveforms are output. When a buffer register B is set to (buffer register A value – 1), asymmetric PWM waveforms are output.

Figure 10.83 shows an example of double buffer operation.
Each register data is transferred as follows.

- After TGRD (buffer A) in MTU4 or MTU7 is written to, data is transferred from TGRD (buffer A) in MTU4 or MTU7 to Temp3A or Temp6A (temporary A) and from TGRF (buffer B) in MTU4 or MTU7 to Temp3B or Temp6B (temporary B).
- With timing (1) in the figure, data is transferred from Temp3A or Temp6A (temporary A) to TGRB (compare) in MTU4 or MTU7.
- With timing (2) in the figure, data is transferred from Temp3B or Temp6B (temporary B) to TGRB (compare) in MTU4 or MTU7.

In the crest interval (Tb1 interval), the compare register and temporary register A are valid; in the trough interval (Tb2 interval), the compare register and temporary register B are valid.

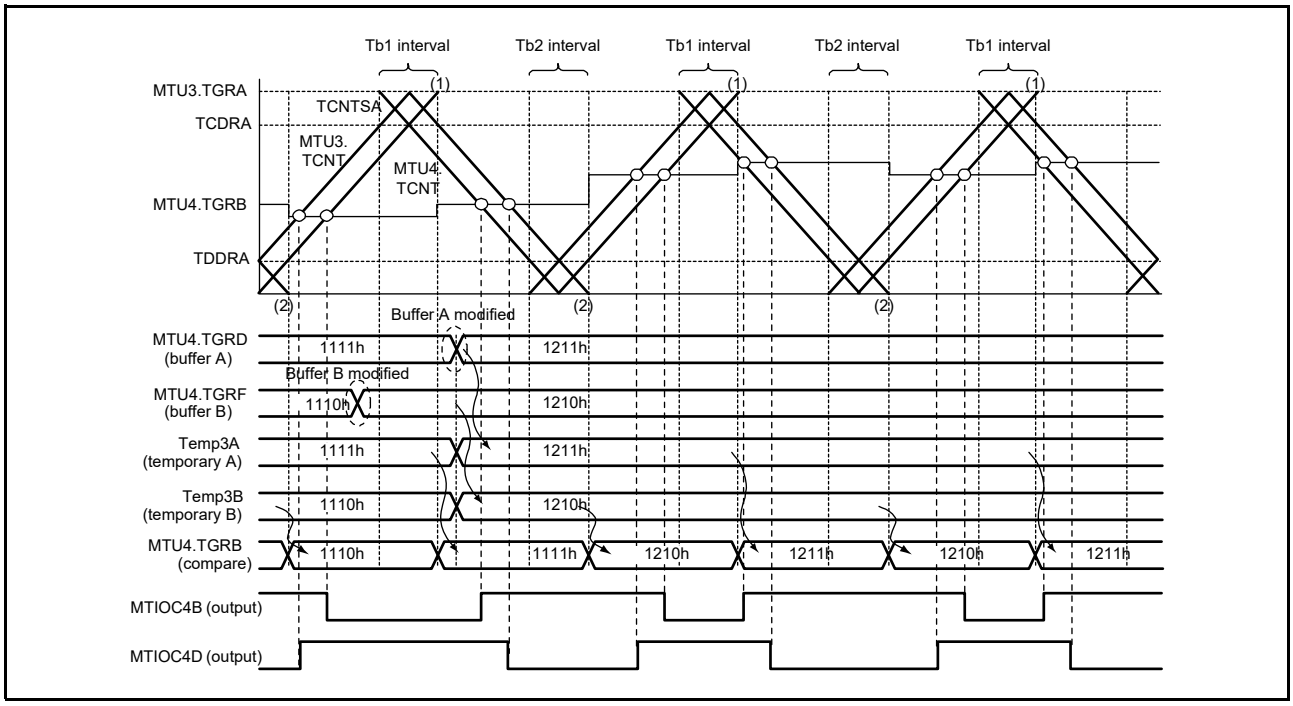


Figure 10.83 Example of Double Buffer Operation

Figure 10.84 shows an example when the buffer write value is smaller than the TDDRA (TDDRB) value, and Figure 10.85 shows an example when the write value is greater than TCDRA (TCDRB).
In the crest interval, the output is controlled according to the compare match with the compare register or temporary register A; in the trough interval, the output is controlled according to the compare match with the compare register or temporary register B.

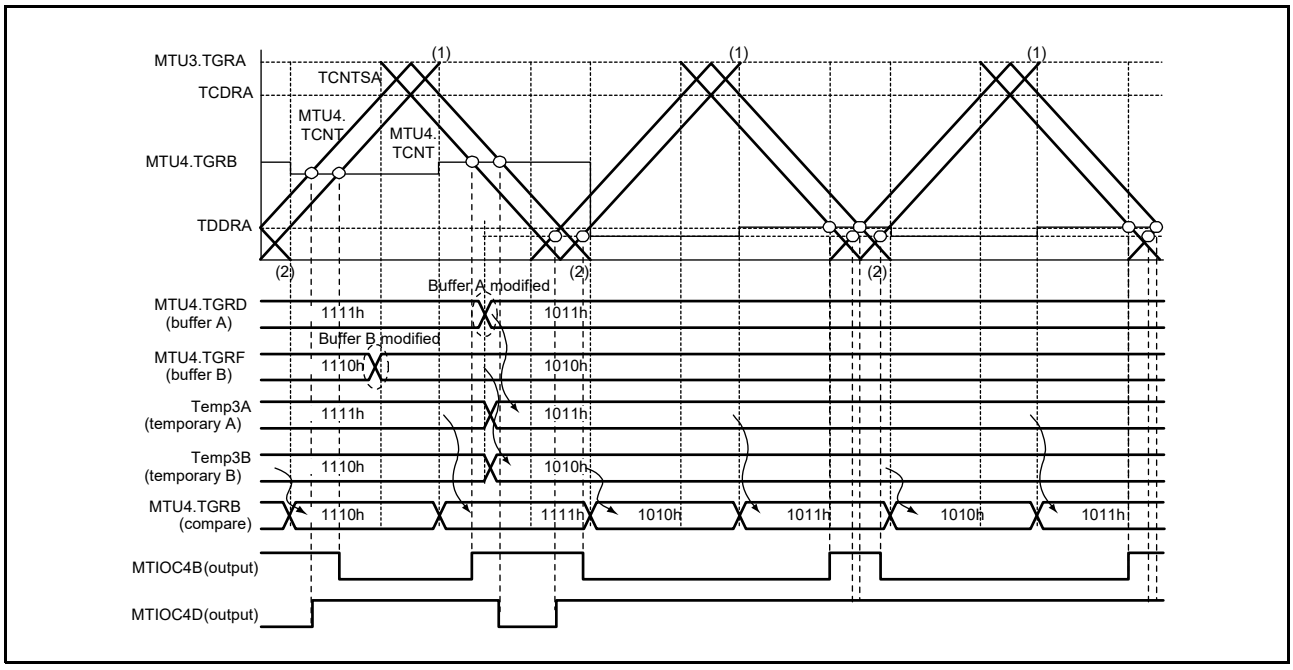


Figure 10.84 Example of Double Buffer Operation (Buffer Write Value is Smaller than TDDRA)

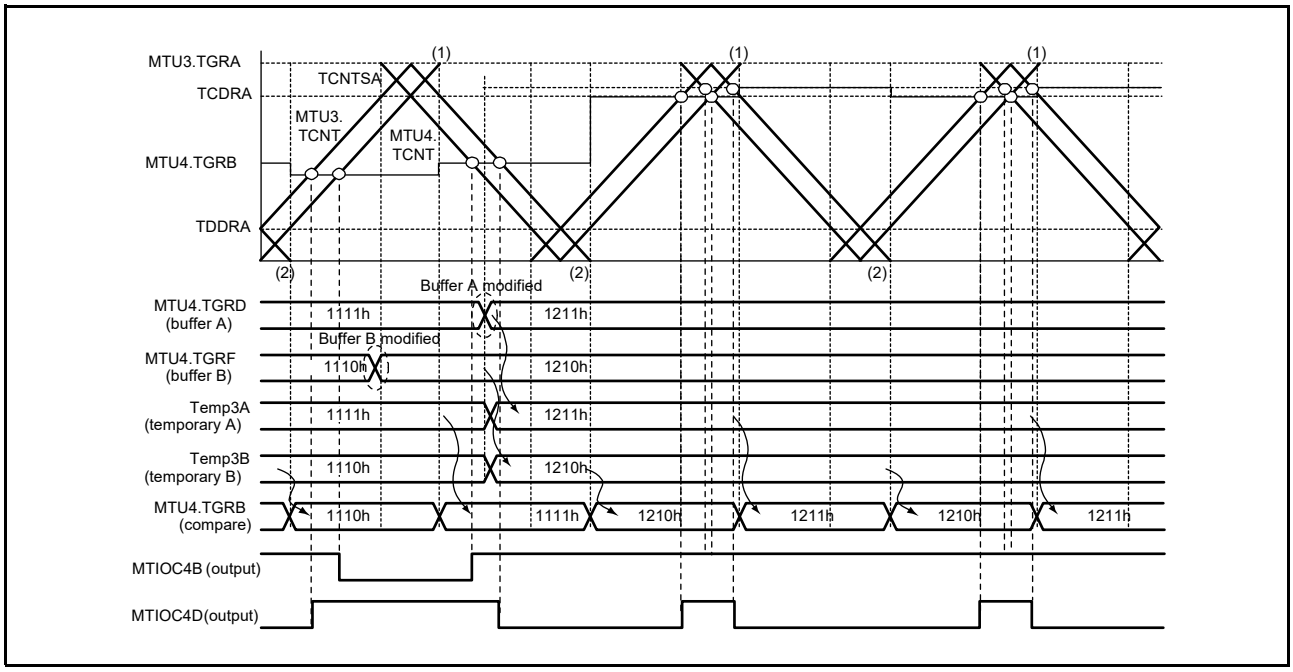


Figure 10.85 Example of Double Buffer Operation (Buffer Write Value is Greater than TCDRA)

(3) Interrupt Skipping Function 1 in Complementary PWM Mode

Interrupts TGIA3 (TGIA6) (at the crest) and TCIV4 (TCIV7) (at the trough) in MTU3 and MTU4 (MTU6 and MTU7) can be skipped up to seven times by making settings in the TITCR1A (TITCR1B) register.
Transfers from a buffer register to a temporary register or a compare register can be skipped in coordination with interrupt skipping by making settings in the TBTERA (TBTERB) register. For the linkage with buffer registers, refer to description (c), Buffer Transfer Control Linked with Interrupt Skipping, below.
A/D converter start requests generated by the A/D converter start request delaying function can also be skipped in coordination with interrupt skipping by making settings in the MTU4.TADCR (MTU7.TADCR) register. For the linkage with the A/D converter start request delaying function, refer to section 10.3.9, A/D Converter Start Request Delaying Function.
The TITCR1A (TITCR1B) register should be set while interrupt skipping function 1 is selected by setting the TITM bit in the timer interrupt skipping mode register (TITMRA or TITMRB) to 0, TGIA3 (TGIA6) interrupt requests are disabled by setting the MTU3.TIER (MTU6.TIER) register, TCIV4 (TCIV7) interrupt requests are disabled by setting the MTU4.TIER (MTU7.TIER) register, and a compare match is not generated. Before changing the skipping count, be sure to clear the T3AEN (T6AEN) and T4VEN (T7VEN) bits to 0 to clear the skipping counter.

(a) Example of Interrupt Skipping Function 1 Setting Procedure

Figure 10.86 shows an example of the interrupt skipping function 1 setting procedure. Figure 10.87 shows the periods during which interrupt skipping count can be changed.

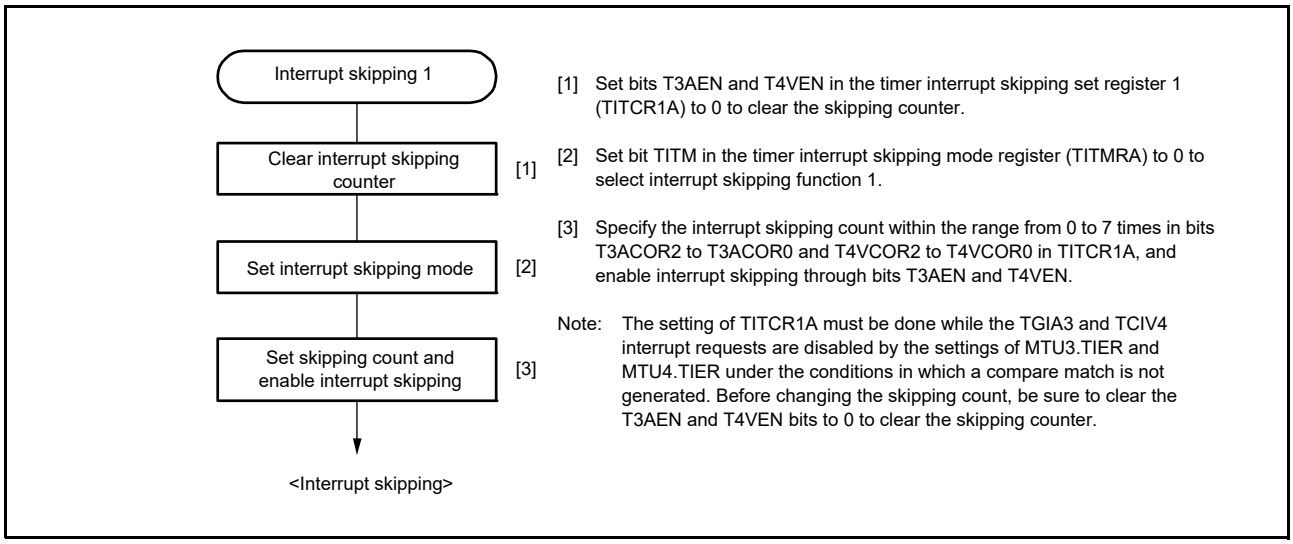


Figure 10.86 Example of Interrupt Skipping Function 1 Setting Procedure

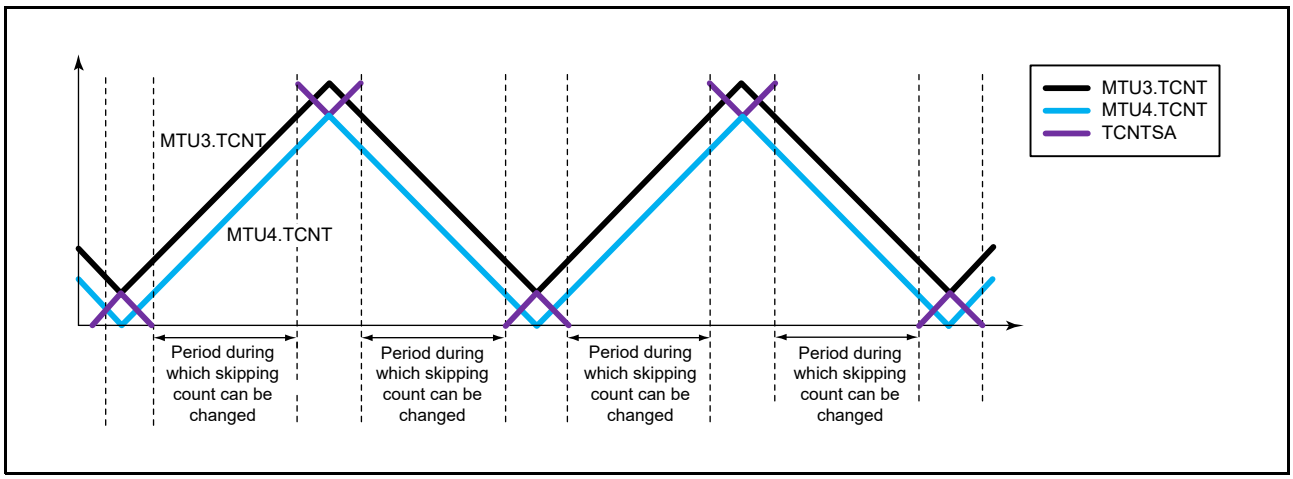


Figure 10.87 Periods during which Interrupt Skipping Count can be Changed

(b) Example of Interrupt Skipping Function 1

Figure 10.88 shows an example of TGIA3 (TGIA6) interrupt skipping in which the interrupt skipping count is set to three by the T3ACOR (T6ACOR) bits and the T3AEN (T6AEN) bit is set to 1 in the TITCR1A (TITCR1B) register.

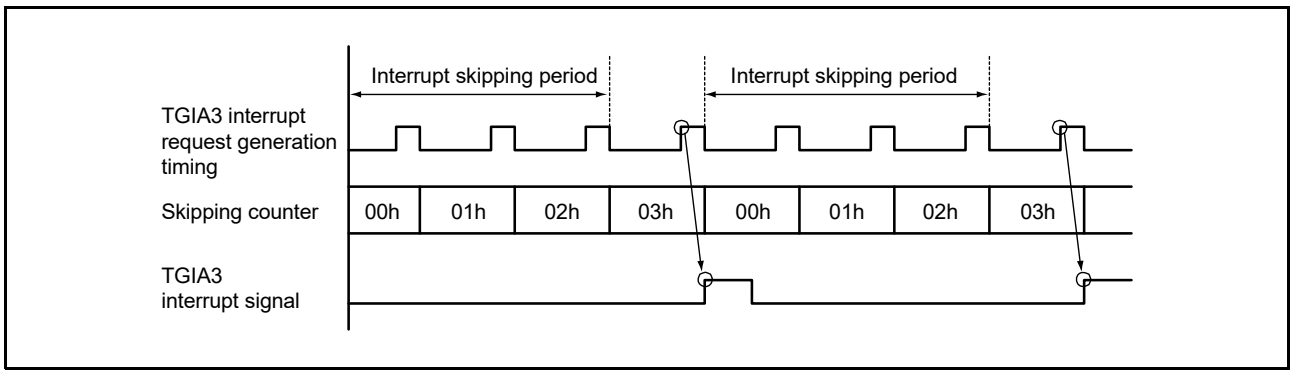
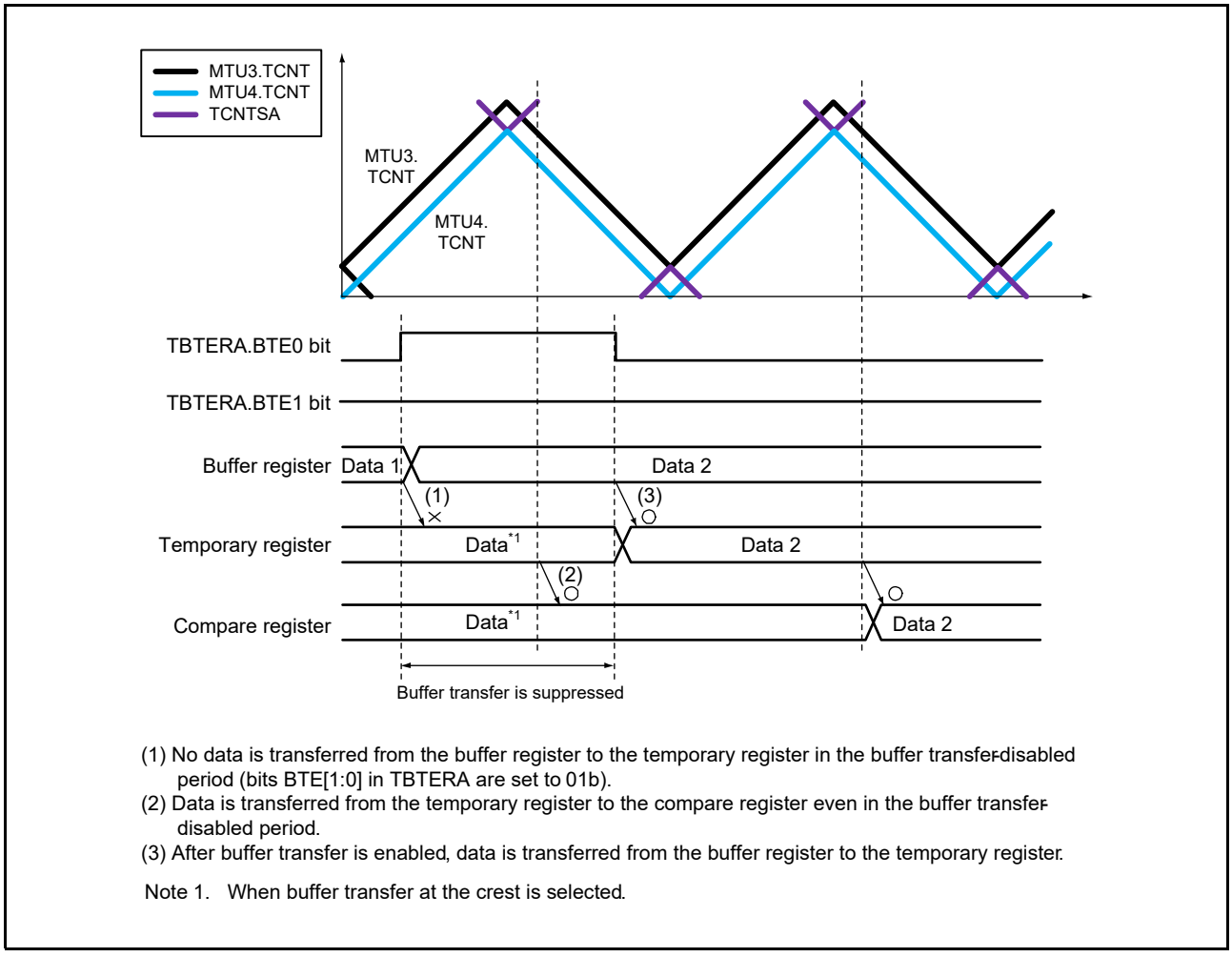


Figure 10.88 Example of Interrupt Skipping Function 1

(c) Buffer Transfer Control Linked with Interrupt Skipping

In complementary PWM mode, whether to transfer data from a buffer register to a temporary register and whether to link the transfer with interrupt skipping can be specified with the BTE[1:0] bits in the TBTERA (TBTERB) register.
Figure 10.89 shows an example of operation when buffer transfer is disabled (BTE[1:0] = 01b). While this setting is valid, data is not transferred from the buffer register to the temporary register.
Figure 10.90 shows an example of operation when buffer transfer is linked with interrupt skipping (BTE[1:0] = 10b). While this setting is valid, data is not transferred from the buffer register outside the buffer transfer-enabled period. Note that the buffer transfer-enabled period differs depending on whether only the T3AEN (T6AEN) bit in the TITCR1A (TITCR1B) register is set to 1, only the T4VEN (T7VEN) bit in the TITCR1A (TITCR1B) register is set to 1, or both the T3AEN and T4VEN (T6AEN and T7VEN) bits are set to 1. Figure 10.91 shows the relationship between the T3AEN (T6AEN) and T4VEN (T7VEN) bit settings in TITCR1A (TITCR1B) and buffer transfer-enabled period.

Note: This function must be used in combination with interrupt skipping function 1.
When interrupt skipping is disabled (the T3AEN and T4VEN (T6AEN and T7VEN) bits in the timer interrupt skipping set register 1 (TITCR1A or TITCR1B) are cleared to 0 or the skipping count set bits (T3ACOR and T4VCOR (T6ACOR and T7VCOR)) in TITCR1A (TITCR1B) are cleared to 0), make sure that buffer transfer is not linked with interrupt skipping (clear the BTE1 bit in TBTERA or TBTERB to 0).
If buffer transfer is linked with interrupt skipping while interrupt skipping is disabled, buffer transfer is never performed.



(1) No data is transferred from the buffer register to the temporary register in the buffer transfer-disabled period (bits BTE[1:0] in TBTERA are set to 01b).
(2) Data is transferred from the temporary register to the compare register even in the buffer transfer-disabled period.
(3) After buffer transfer is enabled, data is transferred from the buffer register to the temporary register.

Note 1. When buffer transfer at the crest is selected.

Figure 10.89 Example of Operation When Buffer Transfer is Disabled (BTE[1:0] = 01b)

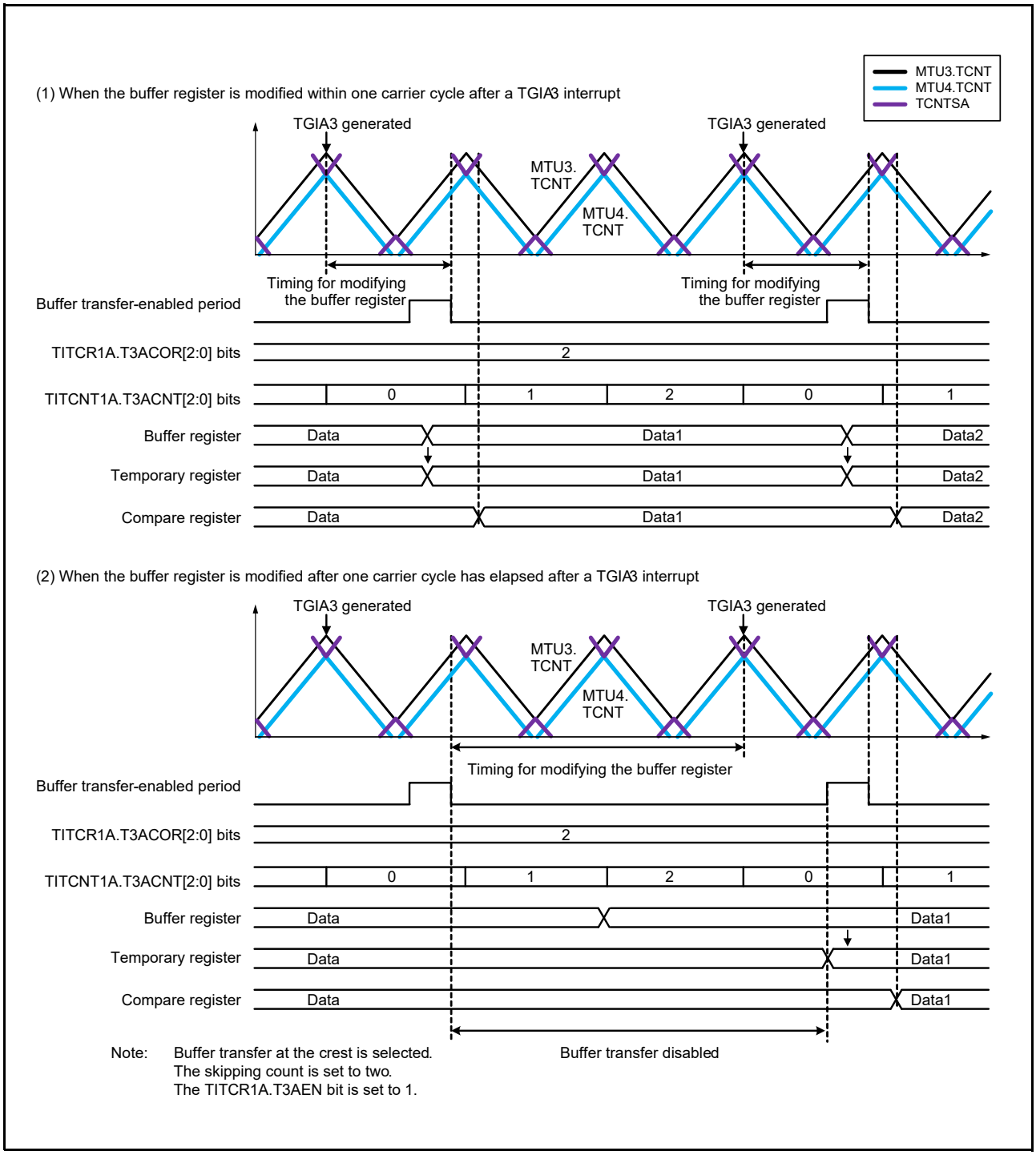


Figure 10.90 Example of Operation When Buffer Transfer is Linked with Interrupt Skipping (BTE[1:0] = 10b)

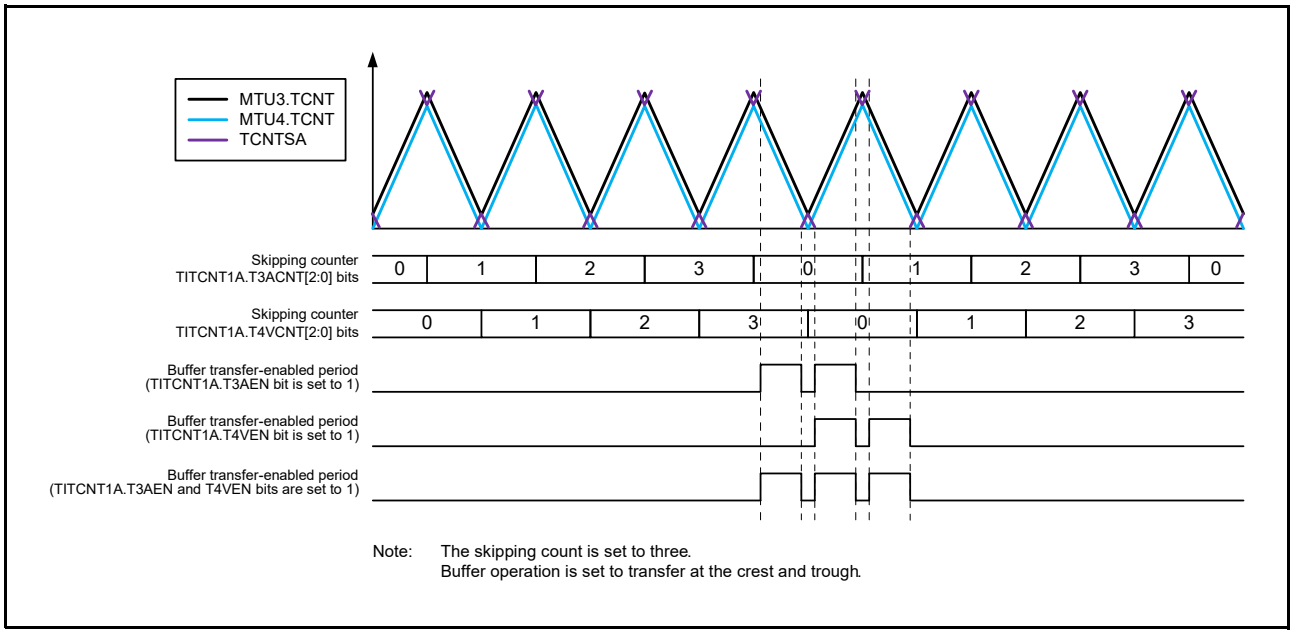


Figure 10.91 Relationship between Bits T3AEN and T4VEN in TITCR1A and Buffer Transfer-Enabled Period

(4) Complementary PWM Mode Output Protection Functions

The following output protection functions are provided for complementary PWM mode.

(a) Register and Counter Miswrite Prevention Function

Access from the CPU to the mode registers, control registers, compare registers, and counters can be enabled or disabled by setting the RWE bit in the TRWERA (TRWERB) register. The applicable registers are some of the registers in MTU3, MTU4, MTU6, and MTU7 shown below:
MTU3.TCR, MTU4.TCR, MTU3.TCR2, MTU4.TCR2, MTU3.TMDR1, MTU4.TMDR1, MTU3.TIORH, MTU4.TIORH, MTU3.TIORL, MTU4.TIORL, MTU3.TIER, MTU4.TIER,
MTU3.TCNT, MTU4.TCNT, MTU3.TGRA, MTU4.TGRA, MTU3.TGRB, MTU4.TGRB,
MTU.TOERA, MTU.TOCR1A, MTU.TOCR2A, MTU.TGCRA, MTU.TCDRA, MTU.TDDRA
MTU6.TCR, MTU7.TCR, MTU6.TCR2, MTU7.TCR2, MTU6.TMDR1, MTU7.TMDR1, MTU6.TIORH, MTU7.TIORH,
MTU6.TIORL, MTU7.TIORL, MTU6.TIER, MTU7.TIER, MTU6.TCNT, MTU7.TCNT,
MTU6.TGRA, MTU7.TGRA, MTU6.TGRB, MTU7.TGRB,
MTU.TOERB, MTU.TOCR1B, MTU.TOCR2B, MTU.TCDRB, and MTU.TDDRB
47 registers in total
This function can disable CPU access to the mode registers, control registers, and counters to prevent miswriting due to CPU runaway. In the access-disabled state, the applicable registers are read as undefined and writing to these registers is ignored.

(b) Halting of PWM Output by External Signal

The PWM output pins of MTU0, MTU3, MTU4, MTU6, and MTU7 can be set to the high-impedance state automatically.
See section 11, Port Output Enable 3 (POE3), for details.

10.3.9 A/D Converter Start Request Delaying Function

A/D converter start requests can be issued in MTU4 or MTU7 by making settings in the timer A/D converter start request control register (MTU4.TADCR or MTU7.TADCR), timer A/D converter start request cycle set registers (MTU4.TADCORA and MTU4.TADCORB, or MTU7.TADCORA and MTU7.TADCORB), and timer A/D converter start request cycle set buffer registers (MTU4.TADCOBRA and MTU4.TADCOBRB, or MTU7.TADCOBRA and MTU7.TADCOBRB).
The A/D converter start request delaying function compares MTU4.TCNT with MTU4.TADCORA or MTU4.TADCORB (MTU7.TCNT with MTU7.TADCORA or MTU7.TADCORB), and when their values match, the function issues a respective A/D converter start request (TRG4AN or TRG4BN (TRG7AN or TRG7BN)).
A/D converter start requests (TRG4AN and TRG4BN (TRG7AN and TRG7BN)) can be skipped in coordination with interrupt skipping by making settings in the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the TADCR register in MTU4 (the ITA6AE, ITA7VE, ITB6AE, and ITB7VE bits in the TADCR register in MTU7).

(1) Example of Procedure for Specifying A/D Converter Start Request Delaying Function

Figure 10.92 shows an example of procedure for specifying the A/D converter start request delaying function.

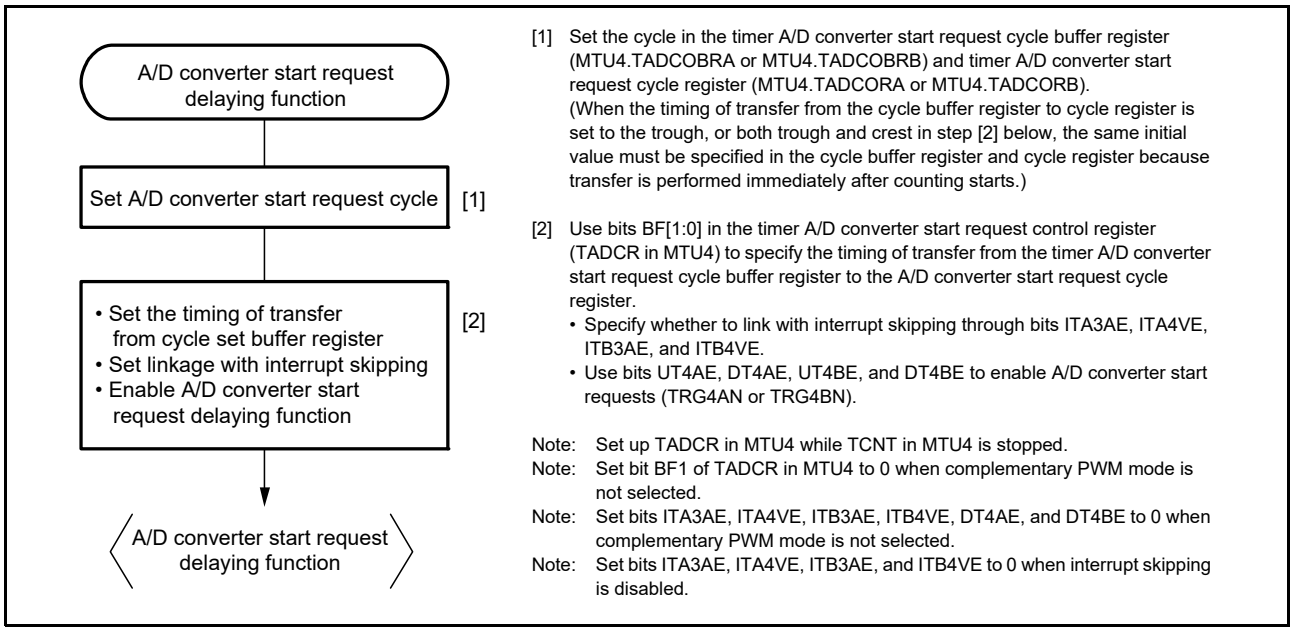


Figure 10.92 Example of Procedure for Specifying A/D Converter Start Request Delaying Function (MTU3 and MTU4)

(2) Basic Example of A/D Converter Start Request Delaying Function Operation

Figure 10.93 shows a basic example of A/D converter start request signal (TRG4AN (TRG7AN)) operation when the trough of MTU4.TCNT (MTU7.TCNT) is specified for the buffer transfer timing and an A/D converter start request signal is output during MTU4.TCNT (MTU7.TCNT) down-counting.

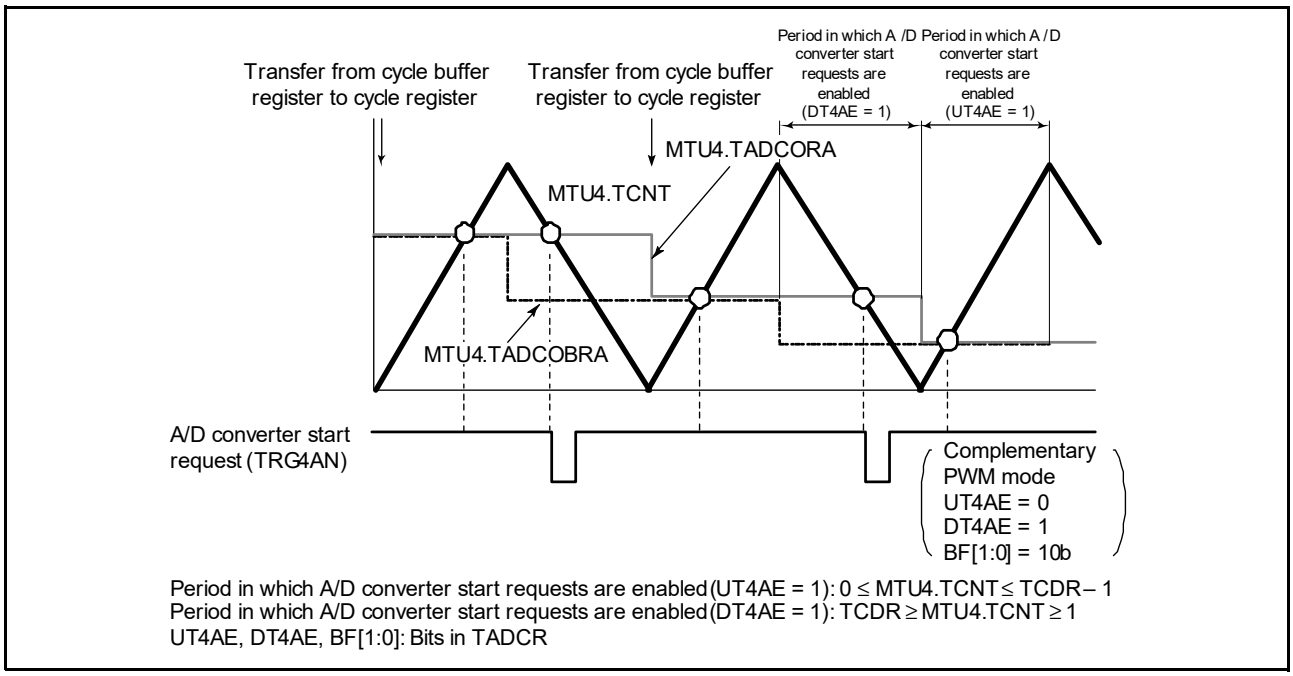


Figure 10.93 Basic Example of A/D Converter Start Request Signal (TRG4AN) Operation

(3) Period in which A/D Converter Start Requests are Enabled

An A/D converter start request (TRG4AN or TRG4BN) can be issued when the TCNT counter value in MTU4 (MTU7) matches the value of the corresponding cycle set register (TADCORA or TADCORB) in MTU4 (MTU7) within the period specified by the UT4AE or UT4BE (UT7AE or UT7BE) bit of TADCR in MTU4 (MTU7).

When the UT4AE or UT4BE (UT7AE or UT7BE) bit is set to 1 in complementary PWM mode, issuing of AD converter start requests is enabled for the period of TCNT up-counting (0 ≤ TCNT ≤ TCDR - 1) in MTU4 (MTU7). When the DT4AE or DT4BE (DT7AE or DT7BE) bit is set to 1, issuing of AD converter start requests is enabled for the period of TCNT down-counting (TCDR ≥ TCNT ≥ 1) in MTU4 (MTU7) (Figure 10.93).

(4) Buffer Transfer

The data in the timer A/D converter start request cycle set registers (MTU4.TADCORA and MTU4.TADCORB, or MTU7.TADCORA and MTU7.TADCORB) is updated by writing data to the timer A/D converter start request cycle set buffer registers (MTU4.TADCOBRA and MTU4.TADCOBRB, or MTU7.TADCOBRA and MTU7.TADCOBRB). Data is transferred from the buffer registers to the respective cycle set registers at the timing selected with the BF[1:0] bits in the MTU4.TADCR (MTU7.TADCR) register.

In complementary PWM mode, data is also transferred from the timer A/D converter start request cycle set buffer registers to the timer A/D converter start request cycle set registers when MTU4.TGRD (MTU7.TGRD) register is updated.

When using buffer transfer in complementary PWM mode, pay attention to the buffer transfer timing. For details, refer to section 10.6.27, Notes on A/D Converter Start Request Delaying Function in Complementary PWM Mode.

When complementary PWM mode is not selected, be sure to clear the BF1 bit of the TADCR register in MTU4 or MTU7.

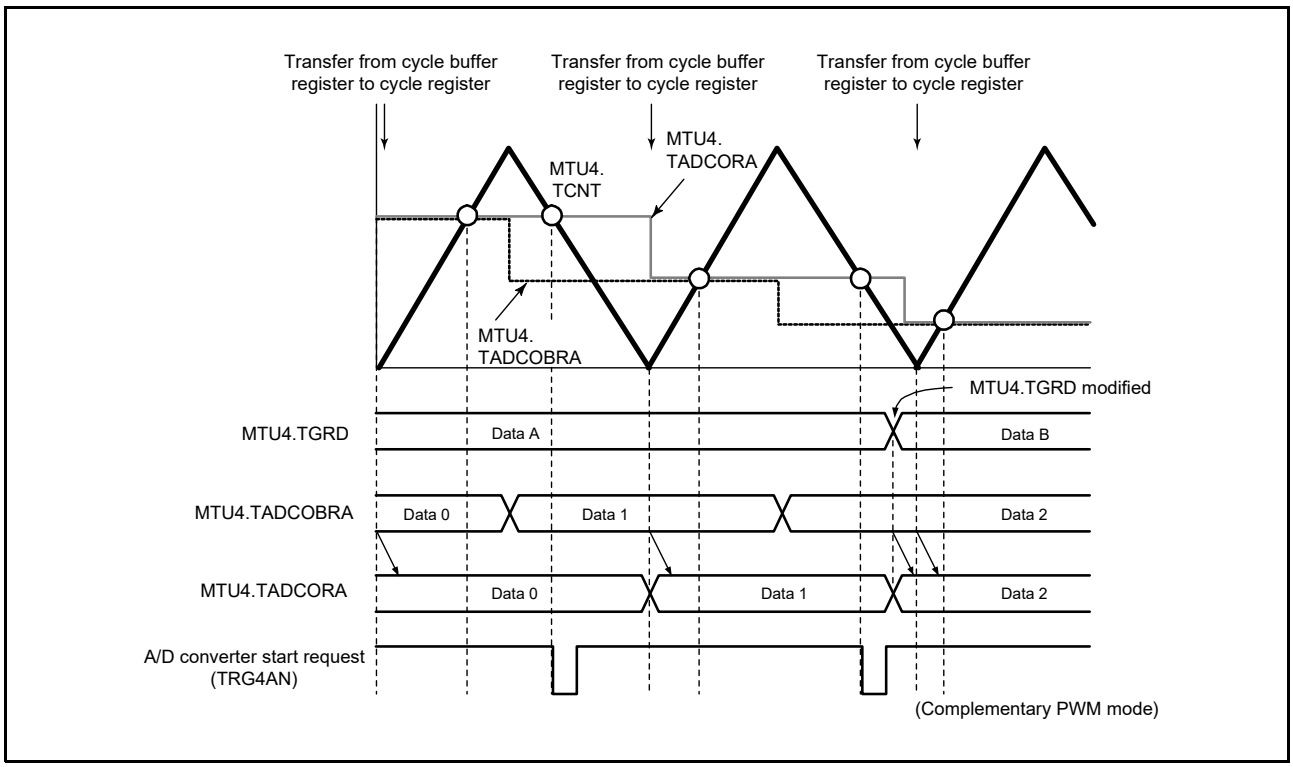


Figure 10.94 Example of A/D Converter Start Request Signal (TRG4AN) and Buffer Transfer Operation

(5) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 1

In complementary PWM mode, A/D converter start requests (TRG4AN and TRG4BN (TRG7AN and TRG7BN)) can be issued in coordination with interrupt skipping 1 by setting the ITA3AE, ITA4VE, ITB3AE, and ITB4VE (ITA6AE, ITA7VE, ITB6AE, and ITB7VE) bits of the TADCR register in MTU4 (MTU7).

Figure 10.95 shows an example of A/D converter start request signal (TRG4AN (TRG7AN)) operation when TRG4AN (TRG7AN) output is enabled during MTU4.TCNT (MTU7.TCNT) up-counting and down-counting and A/D converter start requests are linked with interrupt skipping 1.

Figure 10.96 shows another example of A/D converter start request signal (TRG4AN (TRG7AN)) operation when TRG4AN (TRG7AN) output is enabled during MTU4.TCNT (MTU7.TCNT) up-counting and A/D converter start requests are linked with interrupt skipping 1.

When complementary PWM mode is not selected, A/D converter start request delaying function cannot be used in coordination with interrupt skipping 1; clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE (ITA6AE, ITA7VE, ITB6AE, and ITB7VE) bits of the TADCR register to 0 in MTU4 (MTU7).

Note: This function should be used in combination with interrupt skipping 1.
When interrupt skipping is disabled (the T3AEN and T4VEN (T6AEN and T7VEN) bits in the timer interrupt skipping set register (TITCR1A (TITCR1B)) are cleared to 0 or the skipping count set bits (T3ACOR and T4VCOR (T6ACOR and T7VCOR)) in TITCR1A (TITCR1B) are cleared to 0), make sure that A/D converter start requests are not linked with interrupt skipping 1 (clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE (ITA6AE, ITA7VE, ITB6AE, and ITB7VE) bits in the timer A/D converter start request control register (MTU4.TADCR (MTU7.TADCR)) to 0).
When this function is used, MTU4.TADCORA and MTU4.TADCORB (MTU7.TADCORA and MTU7.TADCORB) should be set with the value ranging 0002h to the value set in TCDRA minus 2 (value set in TCDRB minus 2).

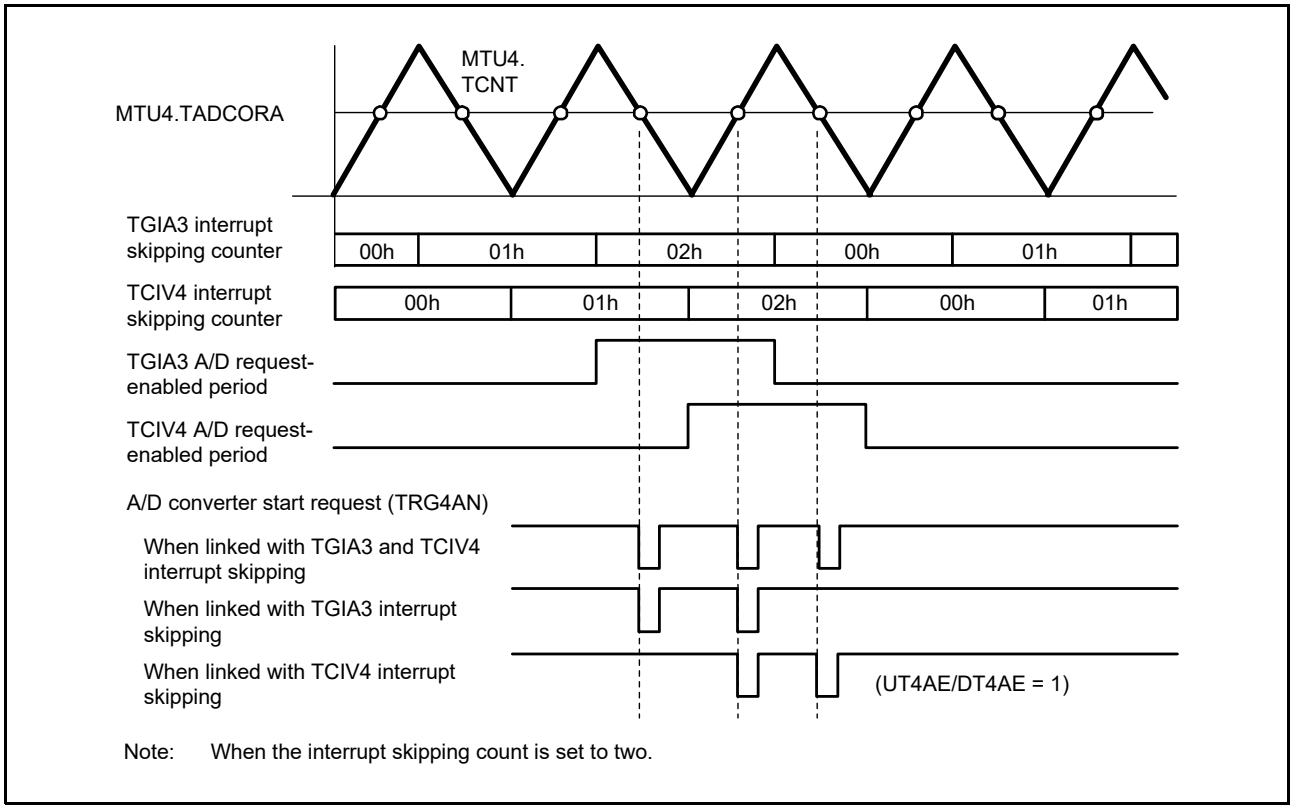


Figure 10.95 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping Function 1 (UT4AE and DT4AE = 1)

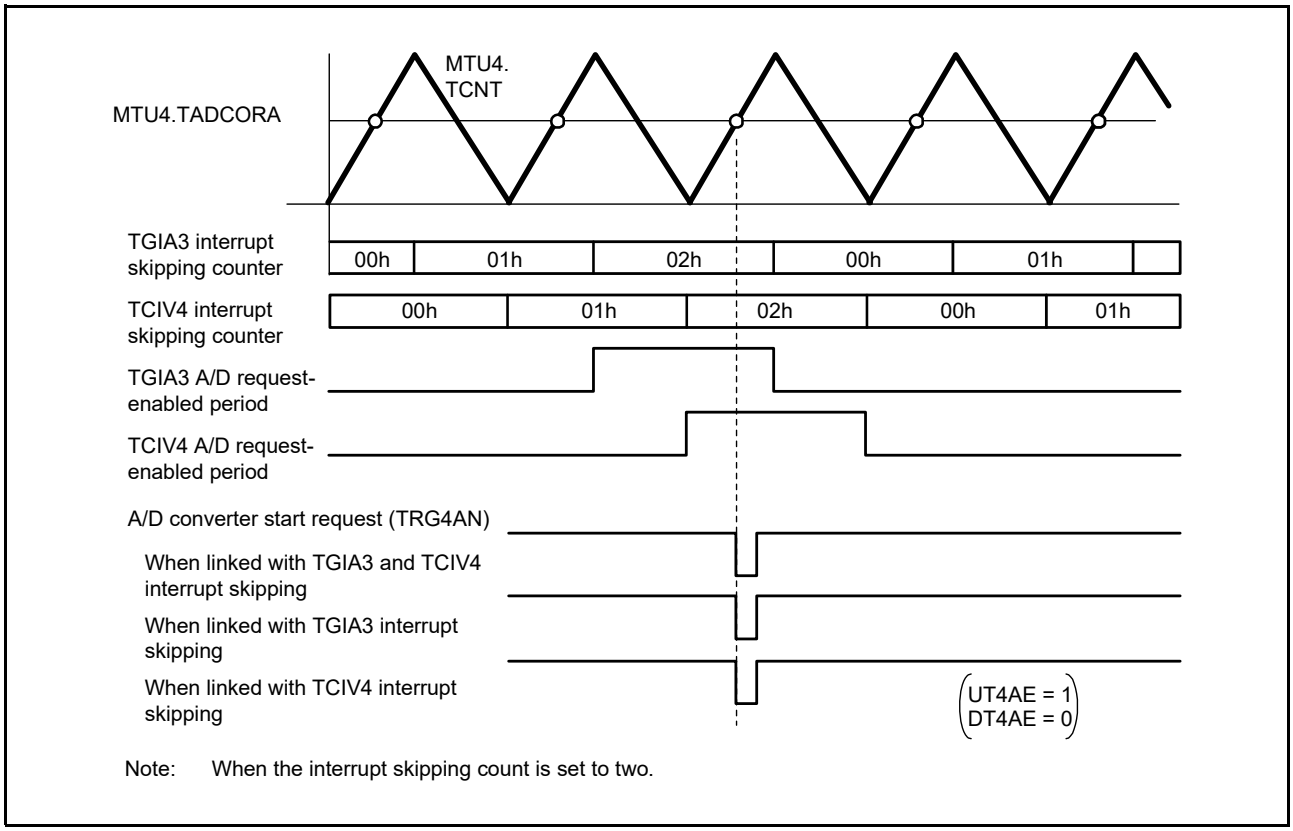


Figure 10.96 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping Function 1 (UT4AE = 1, DT4AE = 0)

(6) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 2

By setting the TITM bit to 1 in the TITMRA (TITMRB) register, the counter starts down-counting from the value (0 to 7) set in the TRG4COR[2:0] (TRG7COR[2:0]) bits in TITCR2A (TITCR2B) register every time an A/D converter start trigger (TRG4AN or TRG4BN (TRG7AN or TRG7BN)) is generated. When the counter value reaches 0 and is reloaded, the TRG4AN and TRG4BN (TRG7AN and TRG7BN) interrupts become valid and an A/D converter start request signal (TRG4ABN (TRG7ABN)) is output.
This function is valid only when the A/D converter request delaying function is enabled.

(a) Example of Procedure for Setting Interrupt Skipping Function 2

Figure 10.97 shows an example of procedure for setting interrupt skipping function 2.

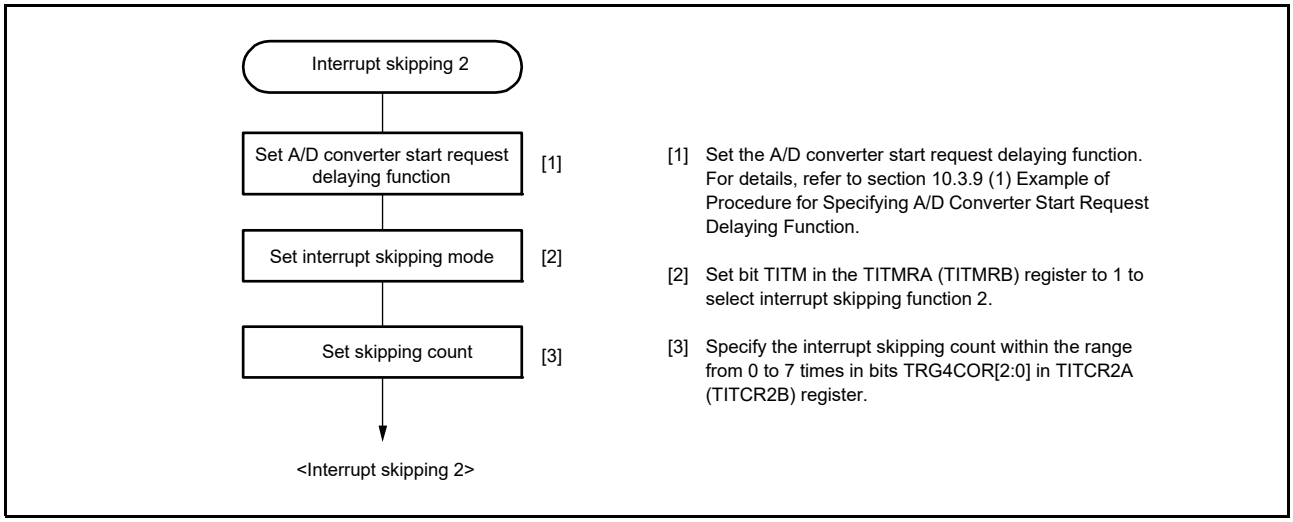


Figure 10.97 Example of Procedure for Setting Interrupt Skipping Function 2

(b) Example of Interrupt Skipping Function 2 Operation

Figure 10.98 shows an example of interrupt skipping 2 operation.

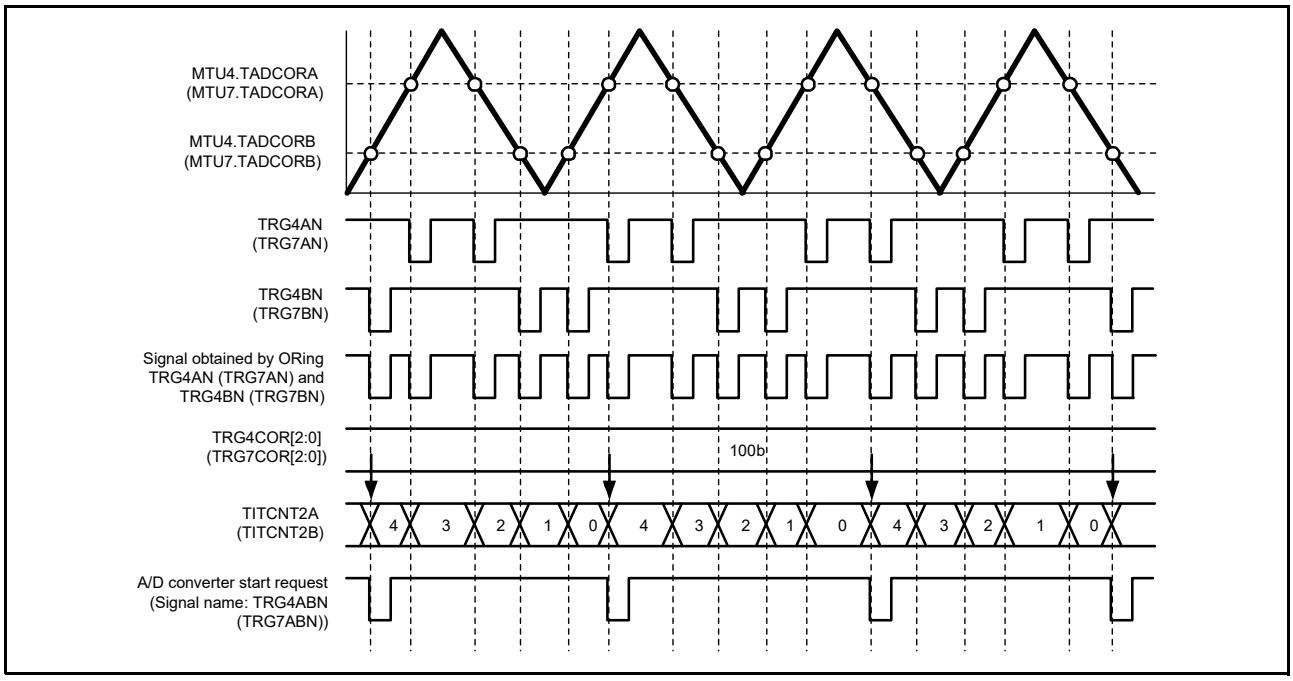


Figure 10.98 Example of Interrupt Skipping 2 Operation (Skipping Count is Set to Four)

10.3.10 Synchronous Operation of MTU0 to MTU4, MTU6, and MTU7

(1) Synchronous Counter Start for MTU0 to MTU4, MTU6, and MTU7

The counters in MTU0 to MTU4, MTU6, and MTU7 can be started synchronously by making the TCSYSTR settings.

(a) Example of Procedure for Setting Synchronous Counter Start for MTU0 to MTU4, MTU6, and MTU7

Figure 10.99 shows an example of the procedure for specifying synchronous counter start in MTU0 to MTU4, MTU6, and MTU7.

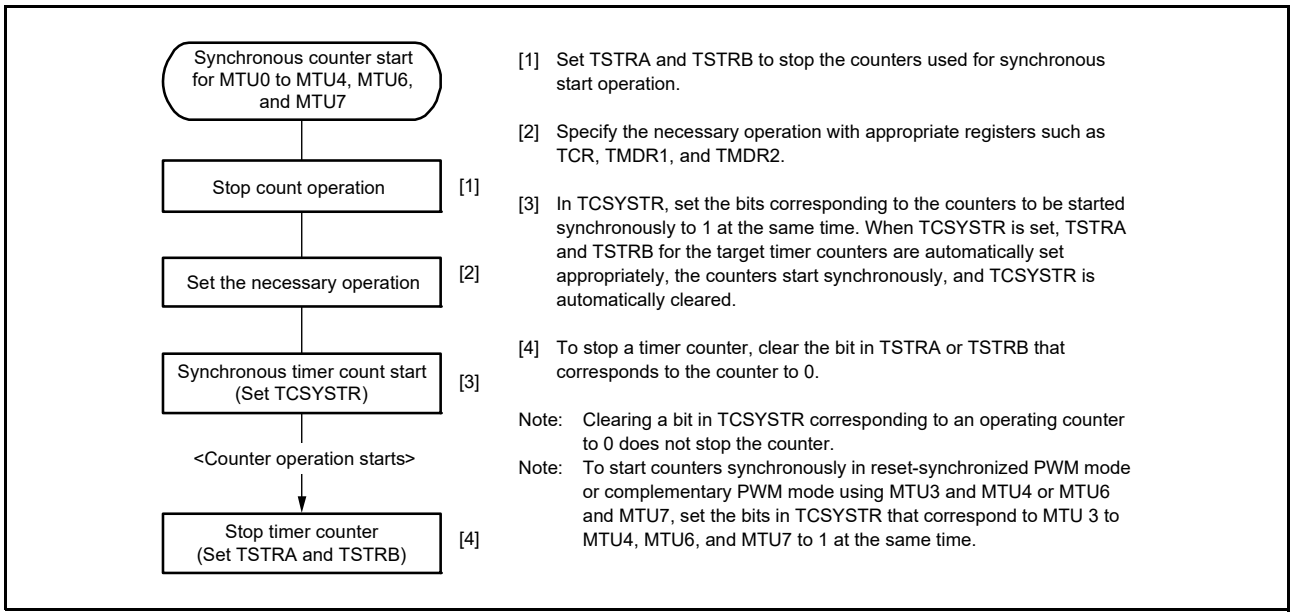


Figure 10.99 Example of Procedure for Specifying Synchronous Counter Start in MTU0 to MTU4, MTU6, and MTU7

(b) Examples of Synchronous Counter Start Operation

Figure 10.100 shows an example of the synchronous counter start operation in MTU0 to MTU4, MTU6, and MTU7.

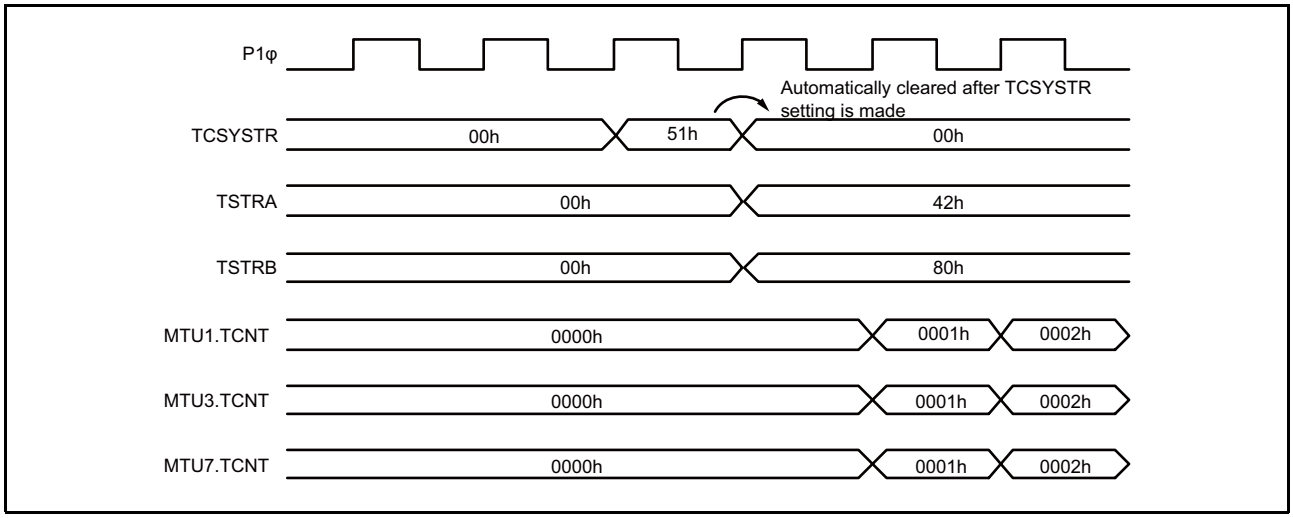


Figure 10.100 Example of Synchronous Counter Start Operation in MTU0 to MTU4, MTU6, and MTU7

(2) Synchronous Counter Clearing for MTU6 and MTU7

The counters in MTU6 and MTU7 can be cleared by the TGImn interrupt generation timing (m = A to D; n = 0 to 2) through the TSYCR setting.

(a) Example of Procedure for Specifying Synchronous Counter Clearing for MTU6 and MTU7

Figure 10.101 shows an example of the procedure for specifying synchronous counter clearing in MTU6 and MTU7 by using the interrupt generation timing.

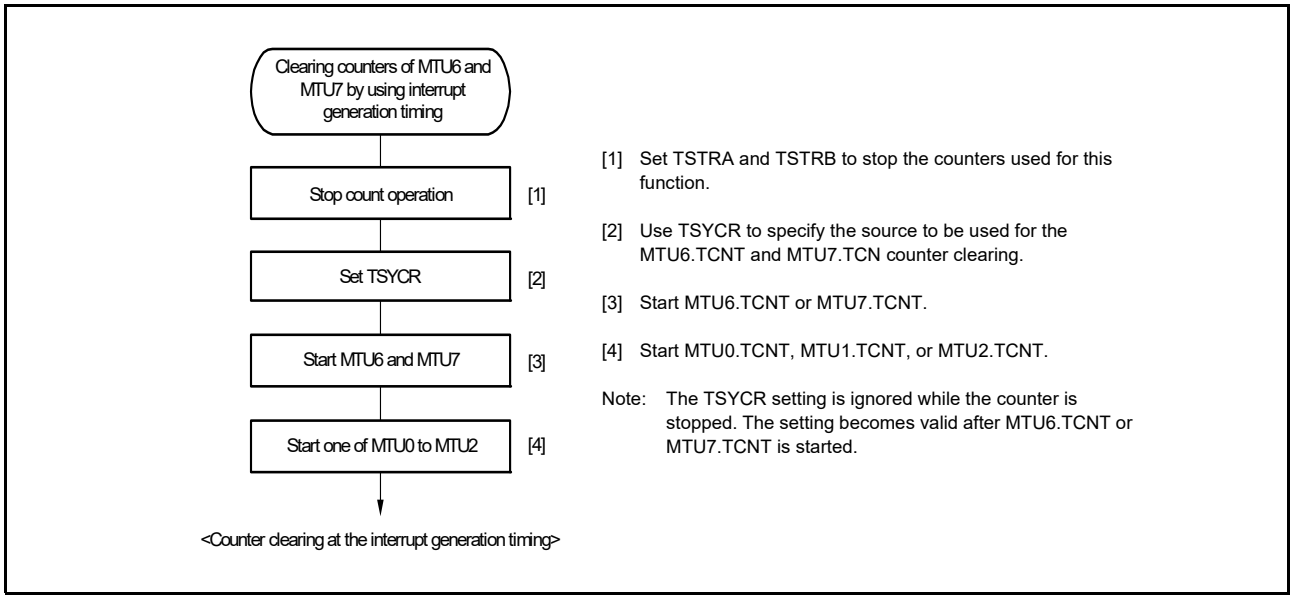


Figure 10.101 Example of Procedure for Specifying Synchronous Counter Clearing in MTU6 and MTU7

(b) Examples of Synchronous Counter Clearing in MTU6 and MTU7

Figure 10.102 and Figure 10.103 show examples of synchronous counter clearing in MTU6 and MTU7 by using the interrupt generation timing.

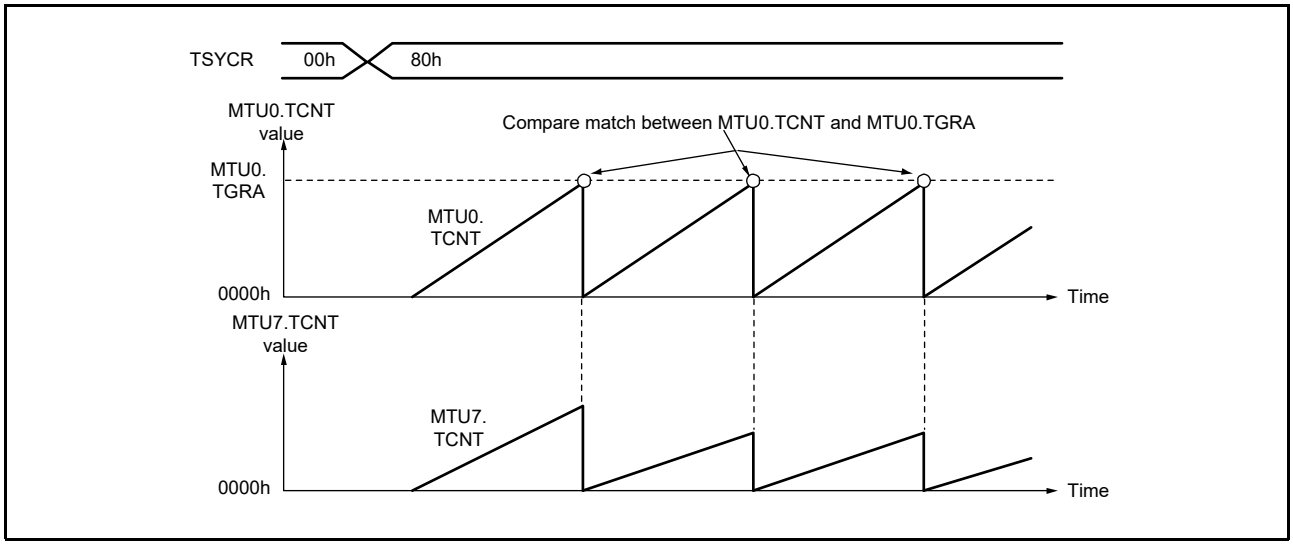


Figure 10.102 Example of Synchronous Counter Clearing in MTU6 and MTU7 (1)

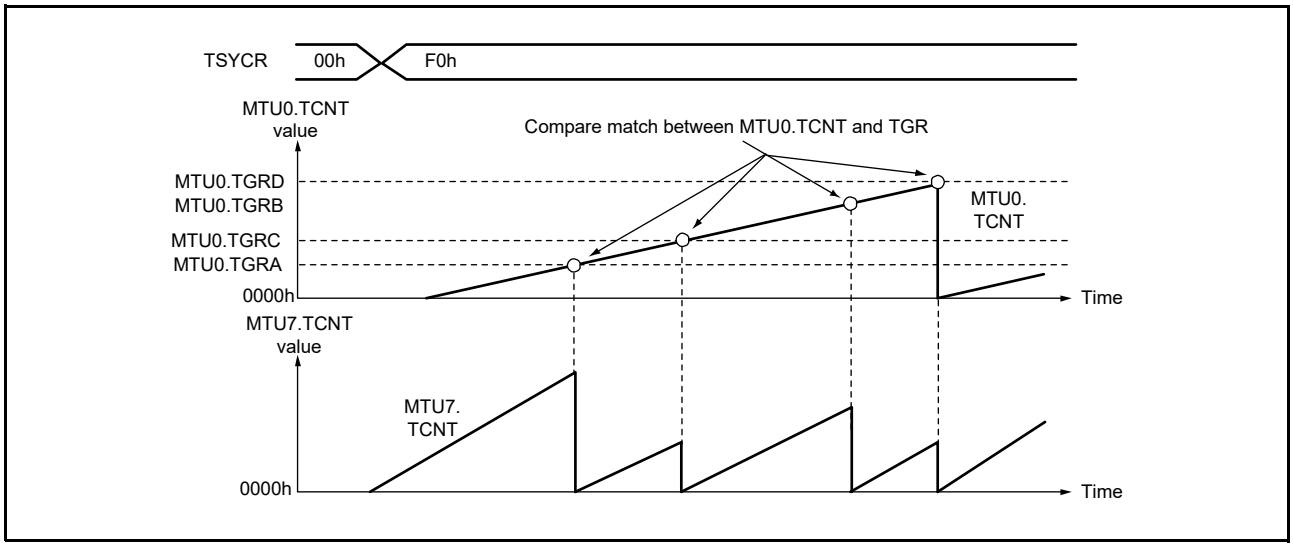


Figure 10.103 Example of Synchronous Counter Clearing in MTU6 and MTU7 (2)

10.3.11 External Pulse Width Measurement

The pulse widths of up to three external input lines can be measured in MTU5.
When the IOC[4:0] bits in MTU5.TIORU, MTU5.TIORV, MTU5.TIORW are set for pulse width measurement, the pulse width of the signal input to the MTIC5U, MTIC5V, and MTIC5W pins are measured. TCNTU, TCNTV, and TCNTW count up while the level specified by the IOC[4:0] bits is input.
Figure 10.104 shows an example of setting external pulse width measurement, and Figure 10.105 an example of external pulse width measurement.

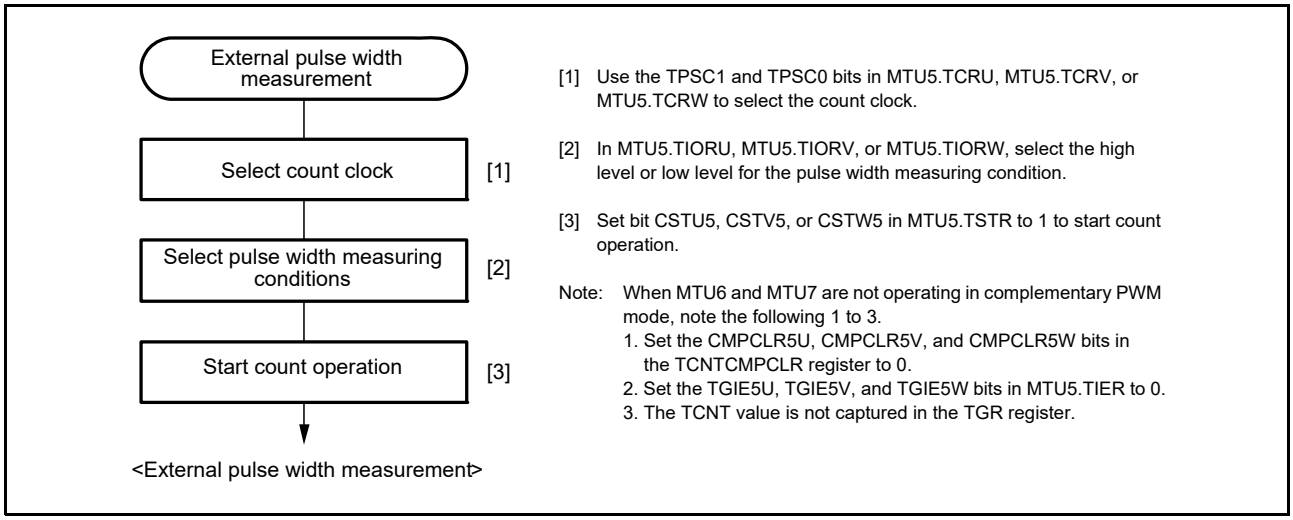


Figure 10.104 Example of External Pulse Width Measurement Setting Procedure

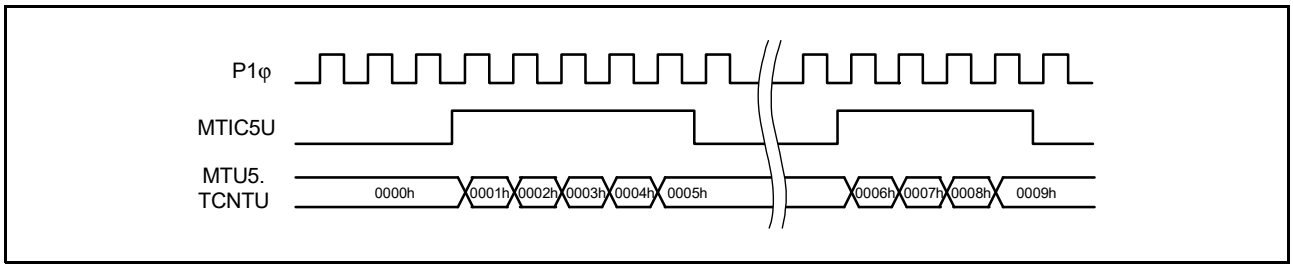


Figure 10.105 Example of External Pulse Width Measurement (Measuring High Pulse Width)

10.3.12 Dead Time Compensation

MTU5 to MTU7 can be used in combination to compensate for the delay in the dead time (the delay between complementary PWM output and inverter output).

Figure 10.106 shows an example of a motor control circuit in which the combination of MTU5 to MTU7 is used to compensate for the dead time delay.

The external pulse measurement function of MTU5 allows the specification of a correction to the duty cycle specified in the PWM output compare registers by measuring the delay between the complementary PWM output and inverter output. This can be used for dead time compensation for the PWM output waveform during complementary PWM operation of MTU6 and MTU7 (Figure 10.107).

Figure 10.108 shows the procedure for setting dead time compensation using MTU5 to MTU7. For details on MTU5 operation at this time, refer to section 10.3.13, TCNTU, TCNTV, and TCNTW Capture at Crest and/or Trough in Complementary PWM Mode.

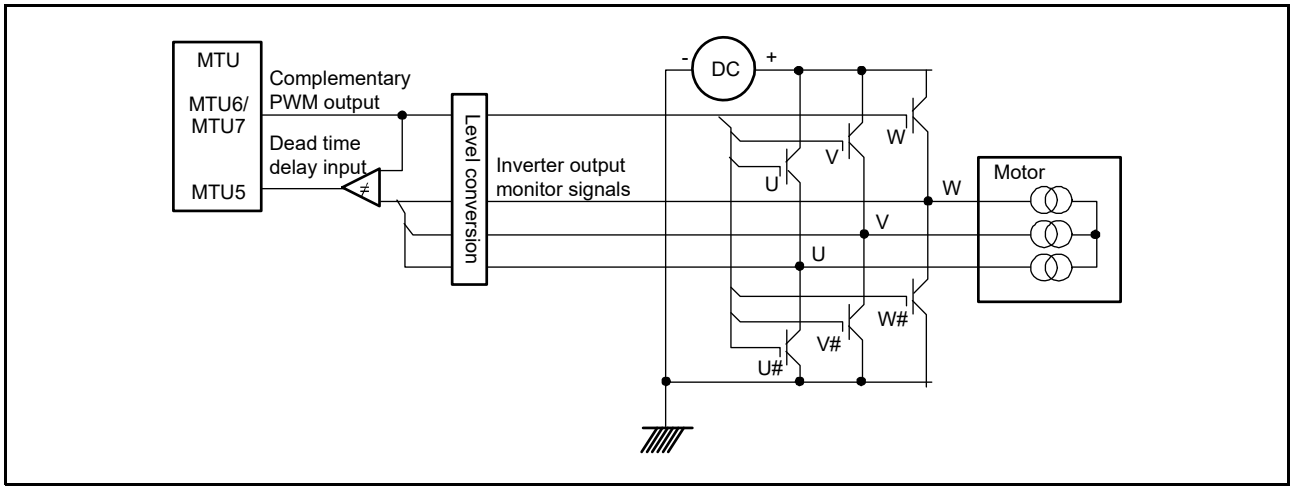


Figure 10.106 Motor Control Circuit Example

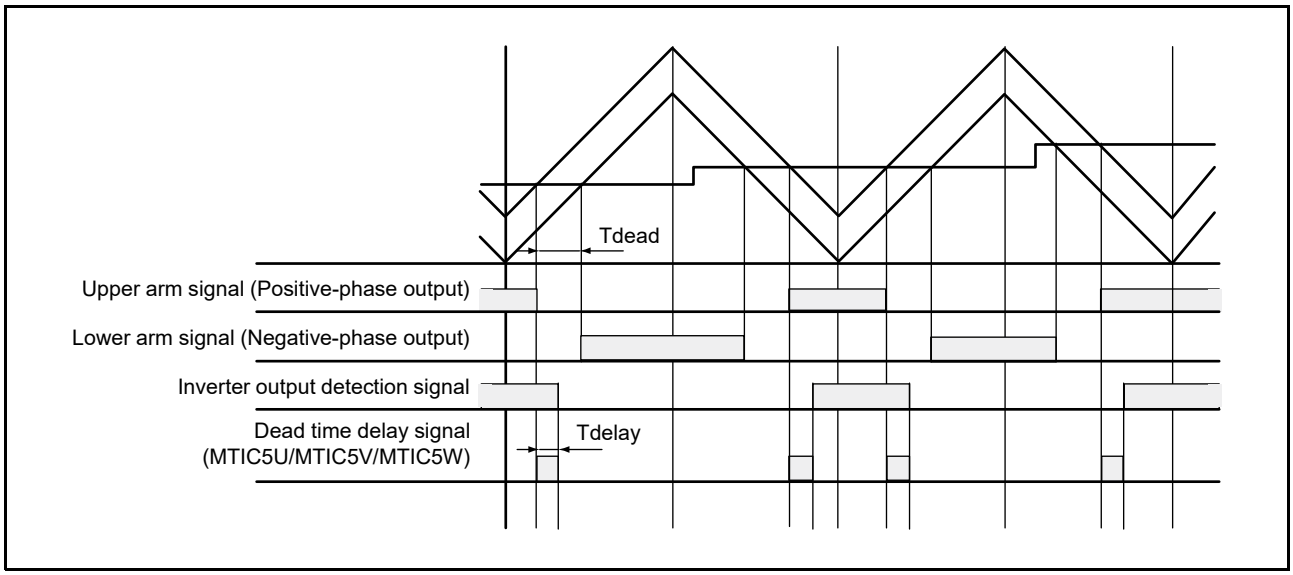
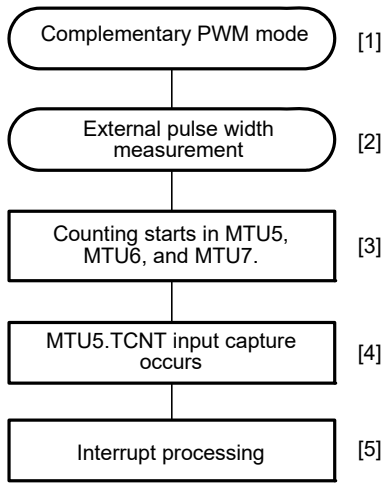


Figure 10.107 Delay in Dead Time in Complementary PWM Operation

(1) Example of Dead Time Compensation Setting Procedure

Figure 10.108 shows an example of dead time compensation setting procedure by using three counters in MTU5.



[1] Set MTU6 and MTU7 to complementary PWM mode. For details, refer to section 10.3.8, Complementary PWM Mode.

[2] Specify the external pulse width measurement function for the target TIOR in MTU5. For details, refer to section 10.3.11, External Pulse Width Measurement.

[3] Set bits CST6 and CST7 in TSTRB and bits CSTU5, CSTV5, and CSTW5 in MTU5.TSTR to 1 to start count operation.

[4] When the capture condition specified in TIOR is satisfied, the MTU5.TCNT value is captured in MTU5.TGR.
To prevent the generation of the MTU5.TGIm5 interrupt (m = U, V, W) request under the capture condition specified in TIOR, do not enable interrupt requests in MTU5.TIER.

[5] For U-phase dead time compensation, when an interrupt is generated at the crest (TGIA6) or trough (TCIV7) in complementary PWM mode, read the MTU5.TGRU value, calculate the difference in time between MTU6.TGRB and MTU5.TGRU, and write the corrected value to MTU6.TGRD in the interrupt processing.
For the V phase and W phase, read the MTU5.TGRV and MTU5.TGRW values and write the corrected values to MTU7.TGRC and TGRD, respectively, in the same way as for U-phase compensation.
The MTU5.TCNT value should be cleared through the TCNTCMPCLR setting or by software.

Note: The port mode registers (PMR) and Pmn pin function control register (PmnPFS) of the GPIO ports should be set up in advance.
As the interrupt flag is set under the capture condition specified in TIOR, do not enable interrupt requests in MTU5.TIER.

Figure 10.108 Example of Dead Time Compensation Setting Procedure

10.3.13 TCNTU, TCNTV, and TCNTW Capture at Crest and/or Trough in Complementary PWM Mode

The external pulse width measurement function of MTU5 can be used to transfer the values of TCNTU, TCNTV, and TCNTW to TGRU, TGRV, and TGRW at the crests, troughs, or both crests and troughs of the complementary PWM output when MTU6 and MTU7 are being used in complementary PWM mode. The type of transfer timing is specified in TIORU, TIORV, and TIORW. When the CMPCLR5U, CMPCLR5V, and CMPCLR5W bits in the TCNTCNPCLR register are set to 1, TCNTU, TCNTV, and TCNTW are cleared to 0000h at the timing for transfer to TGRU, TGRV, and TGRW.
Note that the TCNTU, TCNTV, and TCNTW capture operation at the crest and/or trough of complementary PWM output in MTU5 is not available when MTU3 and MTU4 are used in complementary PWM mode.

Figure 10.109 shows an operation example in which TCNTU is used as a free-running counter without being cleared and the value is captured in TGRU at the crest and trough in complementary PWM mode.

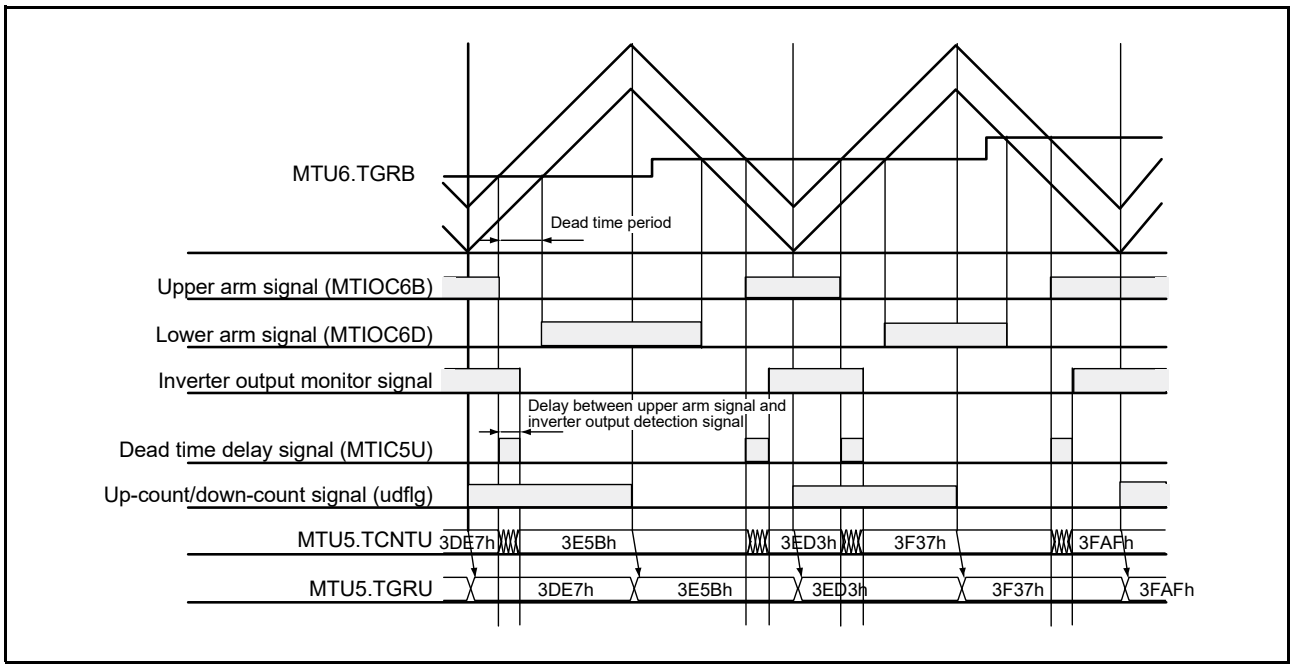


Figure 10.109 TCNTU Capture at Crest and Trough in Complementary PWM Operation

10.3.14 Noise Filter Function

The input capture input pins and external pulse input pins have a noise filter function.
Set the NFCRn register (n = 0 to 7, C) to enable or disable the noise filter function and set the sampling clock. The noise filter for each pin can be enabled or disabled individually, and the sampling clock can be set for each channel.
Figure 10.110 shows the timing of noise filtering.

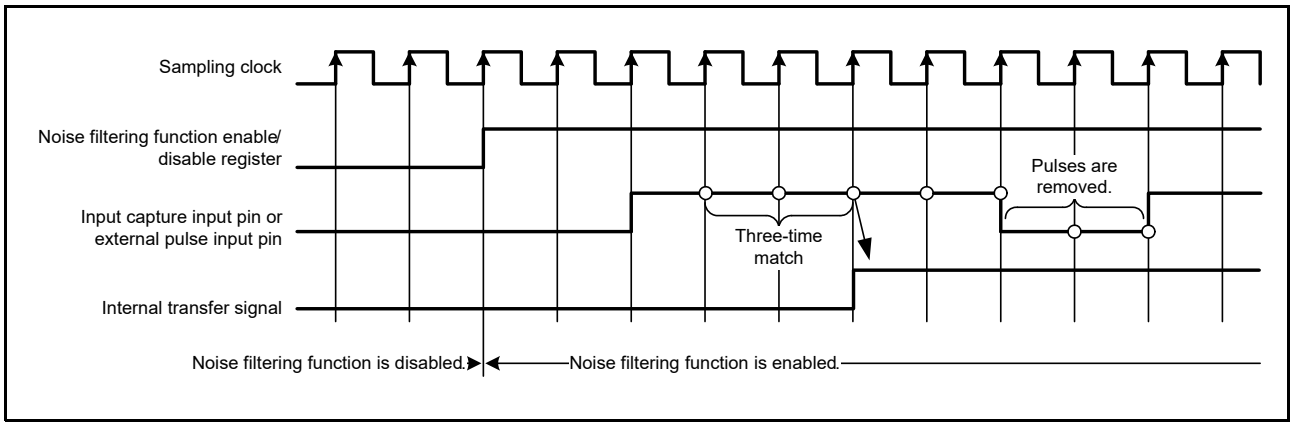


Figure 10.110 Timing of Noise Filtering

10.4 Interrupt Sources

10.4.1 Interrupt Sources and Priorities

There are three kinds of interrupt source; TGR input capture/compare match, TCNT overflow, and TCNT underflow. Each interrupt source has its own enable/disable bit, allowing the generation of interrupt request signals to be enabled or disabled individually.

When an interrupt source is generated, if the corresponding enable/disable bit in TIER is set to 1, an interrupt is requested.

Relative channel priorities can be changed by the interrupt controller; however the priority within a channel is fixed. For details, refer to section 7, Interrupt Controller. Table 10.78 lists the MTU interrupt sources.

Table 10.78 MTU Interrupt Sources

Channel	Name	Interrupt Source	DMAC Activation
MTU0	TGIA0	MTU0.TGRA input capture/compare match	Possible
	TGIB0	MTU0.TGRB input capture/compare match	Possible
	TGIC0	MTU0.TGRC input capture/compare match	Possible
	TGID0	MTU0.TGRD input capture/compare match	Possible
	TCIV0	MTU0.TCNT overflow	Not possible
	TGIE0	MTU0.TGRE compare match	Not possible
	TGIF0	MTU0.TGRF compare match	Not possible
MTU1	TGIA1	MTU1.TGRA input capture/compare match	Possible
	TGIB1	MTU1.TGRB input capture/compare match	Possible
	TCIV1	MTU1.TCNT overflow	Not possible
	TCIU1	MTU1.TCNT underflow	Not possible
MTU2	TGIA2	MTU2.TGRA input capture/compare match	Possible
	TGIB2	MTU2.TGRB input capture/compare match	Possible
	TCIV2	MTU2.TCNT overflow	Not possible
	TCIU2	MTU2.TCNT underflow	Not possible
MTU3	TGIA3	MTU3.TGRA input capture/compare match	Possible
	TGIB3	MTU3.TGRB input capture/compare match	Possible
	TGIC3	MTU3.TGRC input capture/compare match	Possible
	TGID3	MTU3.TGRD input capture/compare match	Possible
	TCIV3	MTU3.TCNT overflow	Not possible
MTU4	TGIA4	MTU4.TGRA input capture/compare match	Possible
	TGIB4	MTU4.TGRB input capture/compare match	Possible
	TGIC4	MTU4.TGRC input capture/compare match	Possible
	TGID4	MTU4.TGRD input capture/compare match	Possible
	TCIV4	MTU4.TCNT overflow/underflow*1	Possible
MTU5	TGIU5	MTU5.TGRU input capture/compare match	Possible
	TGIV5	MTU5.TGRV input capture/compare match	Possible
	TGIW5	MTU5.TGRW input capture/compare match	Possible
MTU6	TGIA6	MTU6.TGRA input capture/compare match	Possible
	TGIB6	MTU6.TGRB input capture/compare match	Possible
	TGIC6	MTU6.TGRC input capture/compare match	Possible
	TGID6	MTU6.TGRD input capture/compare match	Possible
	TCIV6	MTU6.TCNT overflow	Not possible

Table 10.78 MTU Interrupt Sources

Channel	Name	Interrupt Source	DMAC Activation
MTU7	TGIA7	MTU7.TGRA input capture/compare match	Possible
	TGIB7	MTU7.TGRB input capture/compare match	Possible
	TGIC7	MTU7.TGRC input capture/compare match	Possible
	TGID7	MTU7.TGRD input capture/compare match	Possible
	TCIV7	MTU7.TCNT overflow/underflow*1	Possible
MTU8	TGIA8	MTU8.TGRA input capture/compare match	Possible
	TGIB8	MTU8.TGRB input capture/compare match	Possible
	TGIC8	MTU8.TGRC input capture/compare match	Possible
	TGID8	MTU8.TGRD input capture/compare match	Possible
	TCIV8	MTU8.TCNT overflow	Possible

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

Note 1. The underflow interrupt source is valid only in complementary PWM mode.

(1) Input Capture/Compare Match Interrupt

If the TIER.TGIE bit is set to 1 when a TGR input capture/compare match occurs on a channel, an interrupt is requested. The MTU has 33 input capture/compare match interrupts (six for MTU0, four each for MTU3, MTU4, MTU6, MTU7, and MTU8, two each for MTU1 and MTU2, and three for MTU5).

(2) Overflow Interrupt

If the TIER.TCIEV bit is set to 1 when a TCNT overflow occurs on a channel, clearing an interrupt is requested. The MTU provides a total of eight overflow interrupts (one for each channel except MTU5).
In complementary PWM mode, an overflow interrupt is generated even when an underflow occurs in TCNT in MTU4 or MTU7.

(3) Underflow Interrupt

If the TIER.TCIEU bit is set to 1 when a TCNT underflow occurs on a channel, an interrupt is requested. The MTU has two underflow interrupts (one each for MTU1, and MTU2).

10.4.2 DMAC Activation

(1) DMAC Activation

The DMAC can be activated by the TGR input capture/compare match interrupt and the overflow interrupt in MTU4 and MTU7. For details, see section 9, Direct Memory Access Controller.
The MTU provides a total of 33 interrupts (input capture/compare match and overflow interrupts) that can be used as DMAC activation sources: four each for MTU0, MTU3, MTU6, and MTU8, two each for MTU1 and MTU2, five each for MTU4 and MTU7, and three for MTU5.

10.4.3 A/D Converter Activation

The A/D converter can be activated by one of the following three methods in the MTU. Table 10.79 shows the relationship between interrupt sources and A/D converter start request signals.

(1) A/D Converter Activation by TGRA Input Capture/Compare Match or at MTU4.TCNT (MTU7.TCNT) Trough in Complementary PWM Mode

The A/D converter can be activated by the occurrence of a TGRA input capture/compare match in each channel. In addition, if complementary PWM operation is performed while the TTGE2 bit in MTU4.TIER (MTU7.TIER) is set to 1, the A/D converter can be activated at the trough of MTU4.TCNT (MTU7.TCNT) count (MTU4.TCNT (MTU7.TCNT) = 0000h).
A/D converter start request signal TRGAnN is issued to the A/D converter under either of the following conditions (n = 0 to 4, 6, or 7).

- When a TGRA input capture/compare match occurs on a channel while the TIER.TTG bit is set to 1
- When the MTU4.TCNT (MTU7.TCNT) count reaches the trough (MTU4.TCNT (MTU7.TCNT) = 0000h) during complementary PWM operation while the TTGE2 bit in MTU4.TIER (MTU7.TIER) is set to 1

When either condition is satisfied, if A/D converter start signal TRGAnN from the MTU is selected as the trigger in the A/D converter, A/D conversion will start.

(2) A/D Converter Activation by Compare Match between MTU0.TCNT and MTU0.TGRE

A/D converter start request signal TRG0N is issued to the A/D converter when a compare match occurs between MTU0.TCNT and MTU0.TGRE.
When a compare match occurs between MTU0.TCNT and MTU0.TGRE in MTU0 while the TTGE2 bit.TIER2 is set to 1, A/D converter start request TGR0N is issued to the A/D converter. If A/D converter start signal TRG0N from the MTU is selected as the trigger in the A/D converter, A/D conversion will start.

(3) A/D Converter Activation by A/D Converter Start Request Delaying Function

The A/D converter can be activated by generating A/D converter start request signal TRG4AN or TRG4BN (TRG7AN or TRG7BN) when the MTU4.TCNT (MTU7.TCNT) count matches the MTU4.TADCORA or MTU4.TADCORB (MTU7.TADCORA or MTU7.TADCORB) value if the UT4AE, DT4AE, UT4BE, or DT4BE (UT7AE, DT7AE, UT7BE, or DT7BE) bit in the A/D converter start request control register (MTU4.TADCR (MTU7.TADCR)) is set to 1. For details, refer to section 10.3.9, A/D Converter Start Request Delaying Function.
A/D conversion will start when TRG4AN (TRG7AN) is generated if A/D converter start signal TRG4AN (TRG7AN) from the MTU is selected as the trigger in the A/D converter, when TRG4BN (TRG7BN) is generated if TRG4BN (TRG7BN) from the MTU is selected as the trigger in the A/D converter, or when TRG4ABN (TRG7ABN) is generated if TRG4ABN (TRG7ABN) from the MTU is selected as the trigger in the A/D converter.

Table 10.79 Interrupt Sources and A/D Converter Start Request Signals

Target Registers	Interrupt Source	A/D Converter Start Request Signal
MTU0.TGRA and MTU0.TCNT	Input capture/compare match	TRGA0N
MTU1.TGRA and MTU1.TCNT		TRGA1N
MTU2.TGRA and MTU2.TCNT		TRGA2N
MTU3.TGRA and MTU3.TCNT		TRGA3N
MTU4.TGRA and MTU4.TCNT*1		TRGA4N
MTU4.TCNT	MTU4.TCNT trough in complementary PWM mode	
MTU6.TGRA and MTU6.TCNT	Input capture/compare match	TRGA6N
MTU7.TGRA and MTU7.TCNT*1		TRGA7N
MTU7.TCNT	MTU7.TCNT trough in complementary PWM mode	
MTU0.TGRE and MTU0.TCNT	Compare match	TRG0N
MTU4.TADCORA and MTU4.TCNT		TRG4AN
MTU4.TADCORB and MTU4.TCNT		TRG4BN
MTU7.TADCORA and MTU7.TCNT		TRG7AN
MTU7.TADCORB and MTU7.TCNT		TRG7BN
MTU4.TADCORA and MTU4.TCNT, MTU4.TADCORB and MTU4.TCNT	Compare match (interrupt skipping function 2)	TRG4ABN
MTU7.TADCORA and MTU7.TCNT, MTU7.TADCORB and MTU7.TCNT		TRG7ABN

Note 1. Since PWM waveforms are generated in complementary PWM mode, MTU4.TGRA (MTU7.TGRA) compare match not only with MTU4.TCNT (MTU7.TCNT) but also with MTU3.TCNT (MTU6.TCNT) and TCNTSA (TCNTSB) is detected. Accordingly, when compare match with MTU3.TCNT (MTU6.TCNT) and TCNTSA (TCNTSB) occurs, TRGA4N (TRGA7N) is also generated. When MTU3 and MTU 4 (MTU 6 and MTU7) are made to operate in complementary PWM mode for generating an A/D converter start request, use the A/D converter start request by compare match between MTU4.TCNT (MTU7.TCNT) and MTU4.TADCORA or TADCORB (MTU7.TADCORA or TADCORB).

10.5 Operation Timing

10.5.1 Input/Output Timing

(1) TCNT Count Timing

Figure 10.111 and Figure 10.112 show the TCNT count timing in internal clock operation, Figure 10.113 shows the TCNT count timing in external clock operation (normal mode), and Figure 10.114 shows the TCNT count timing in external clock operation (phase counting mode).

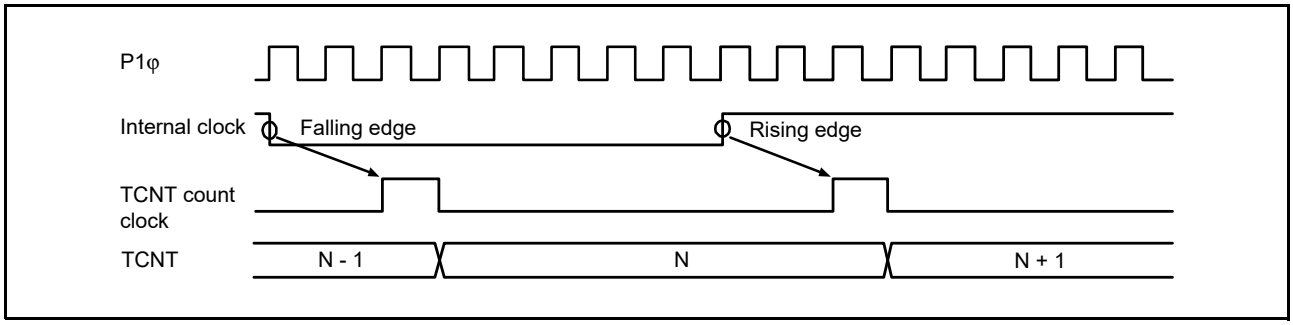


Figure 10.111 Count Timing in Internal Clock Operation (MTU0 to MTU4 and MTU6 to MTU8)

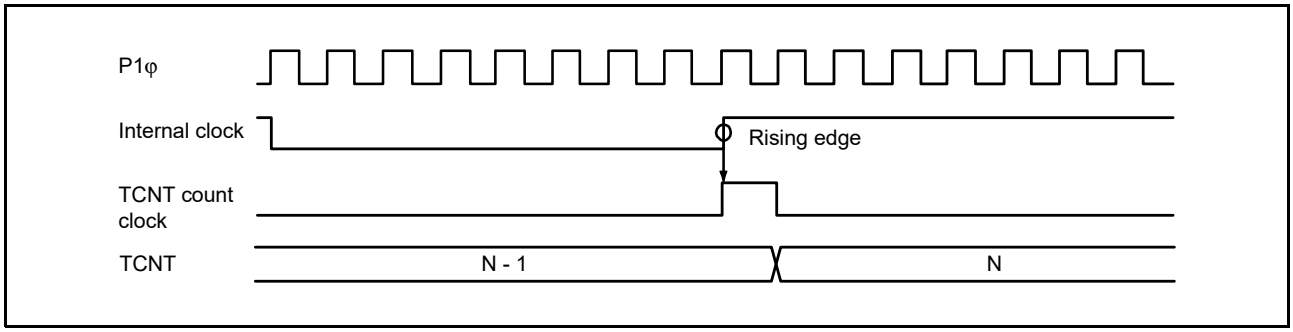


Figure 10.112 Count Timing in Internal Clock Operation (MTU5)

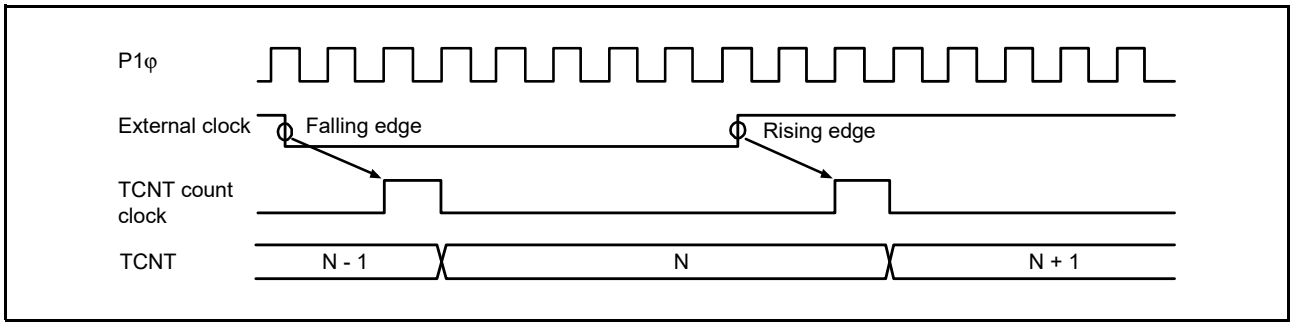


Figure 10.113 Count Timing in External Clock Operation (MTU0 to MTU4, MTU6 to MTU8)

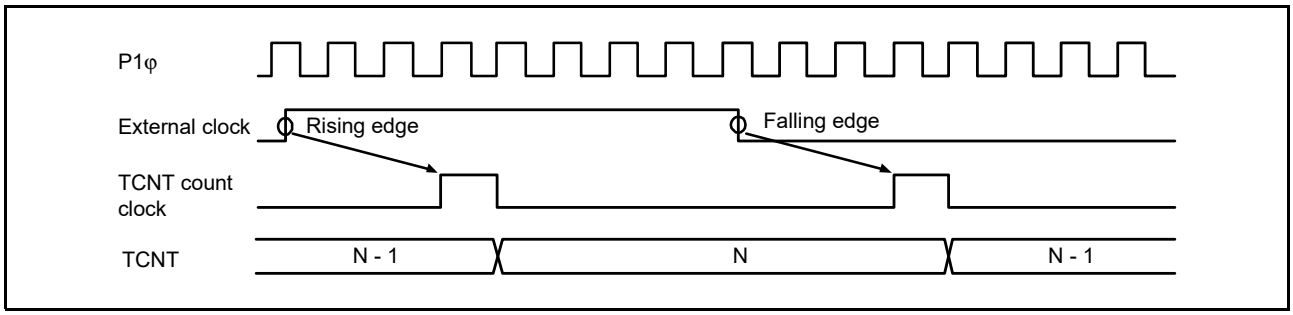


Figure 10.114 Count Timing in External Clock Operation (Phase Counting Mode)

(2) Output Compare Output Timing

A compare match signal is generated in the final state in which TCNT and TGR match (the point at which the count value matched by TCNT is updated). When a compare match signal is generated, the value set in TIOR is output from MTIOCnm pin (n = 0 to 4, 6, 7, 8; m = A to D). After a match between TCNT and TGR, the compare match signal is not generated until the TCNT count clock is generated.

Figure 10.115 shows the output compare output timing (normal mode or PWM mode) and Figure 10.116 shows the output compare output timing (complementary PWM mode or reset-synchronized PWM mode).

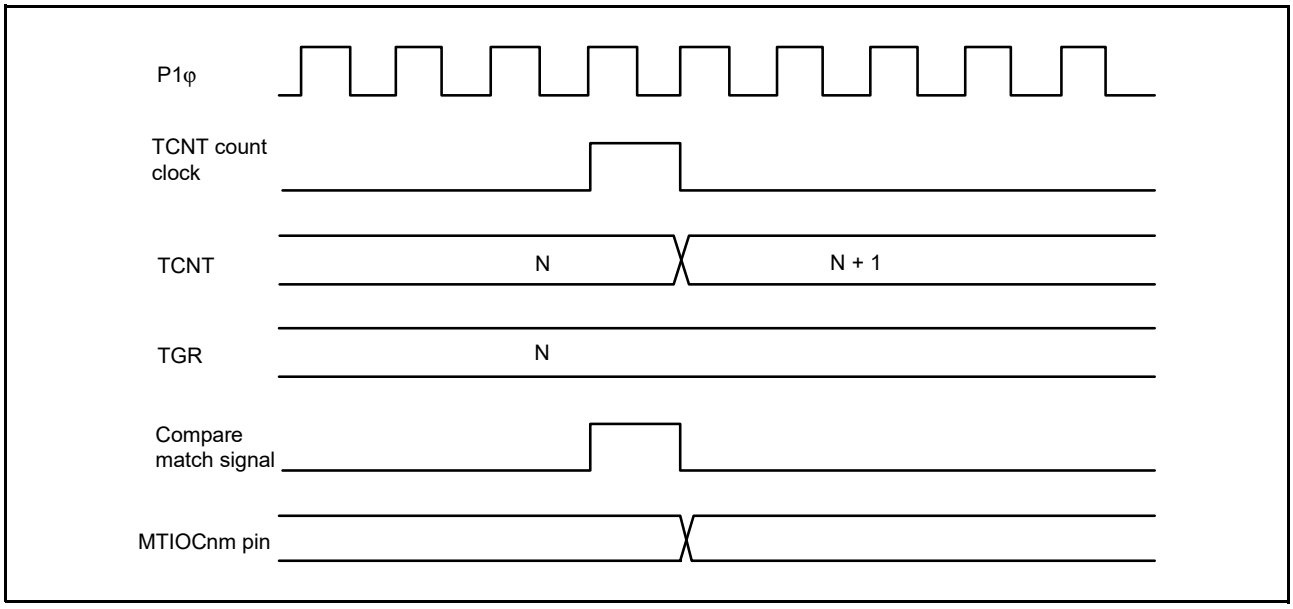


Figure 10.115 Output Compare Output Timing (Normal Mode or PWM Mode) (n = 0 to 4, 6, 7, 8; m = A to D)

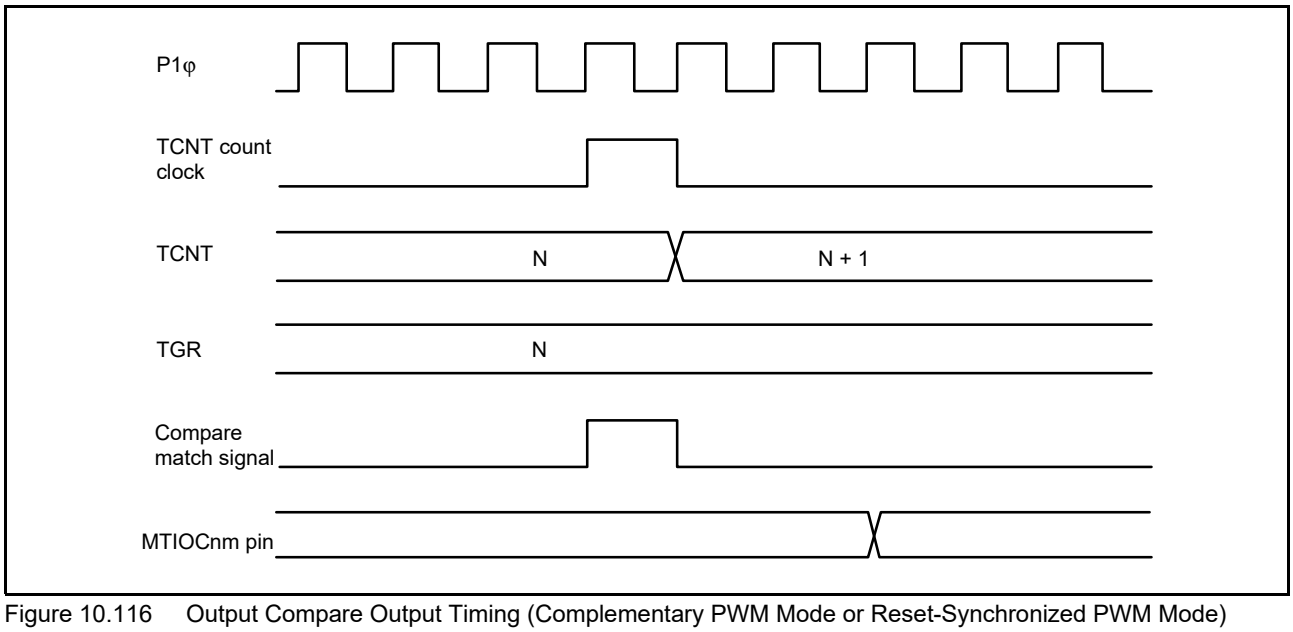


Figure 10.116 Output Compare Output Timing (Complementary PWM Mode or Reset-Synchronized PWM Mode) (n = 0 to 4, 6, 7, 8; m = A to D)

(3) Input Capture Signal Timing

Figure 10.117 shows the input capture signal timing.

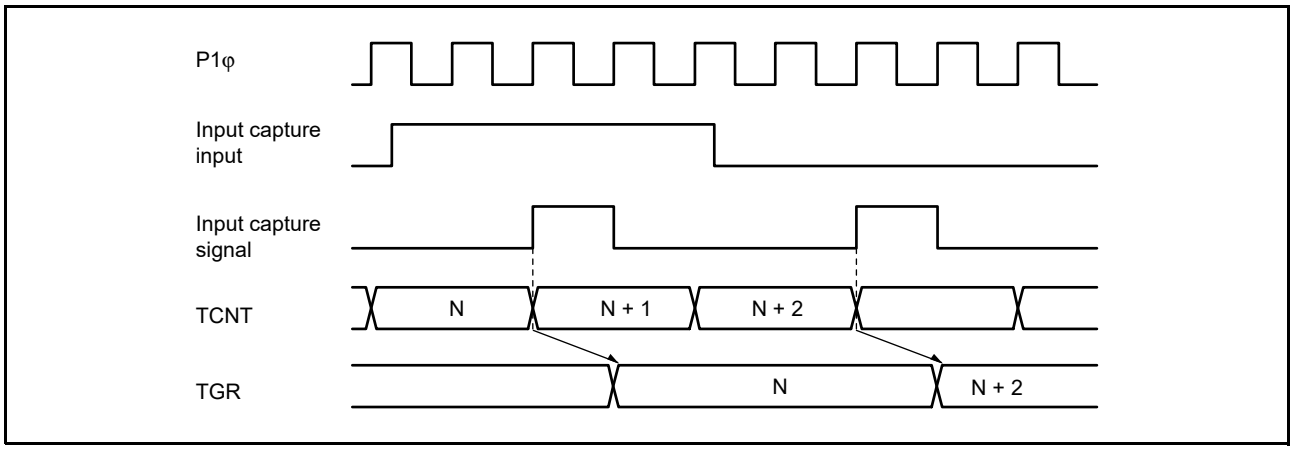


Figure 10.117 Input Capture Input Signal Timing

(4) Timing for Counter Clearing by Compare Match/Input Capture

Figure 10.118 and Figure 10.119 show the timing when counter clearing on compare match is specified, and Figure 10.120 shows the timing when counter clearing on input capture is specified.

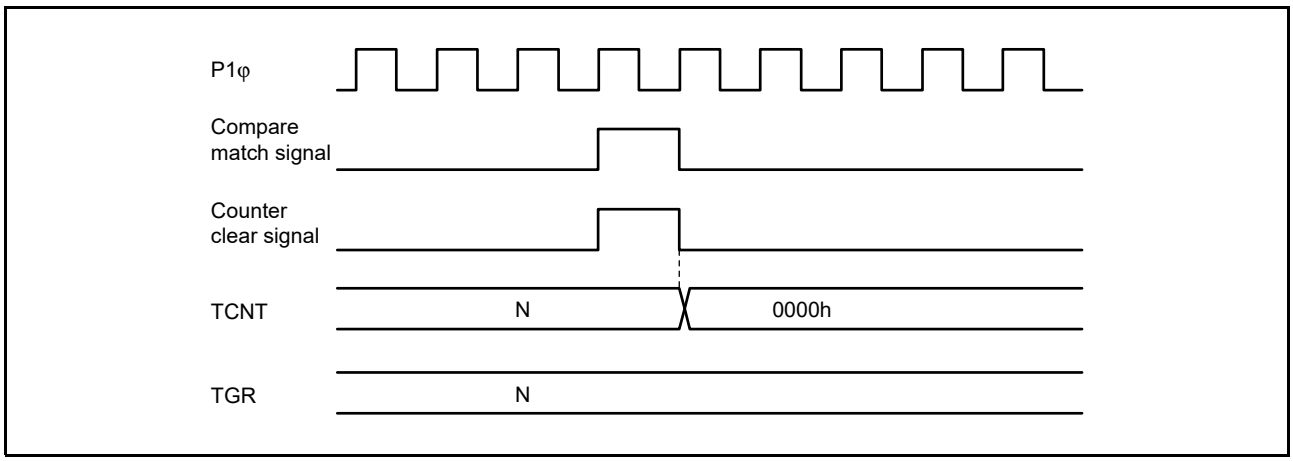


Figure 10.118 Counter Clear Timing (Compare Match) (MTU0 to MTU4 and MTU6 to MTU8)

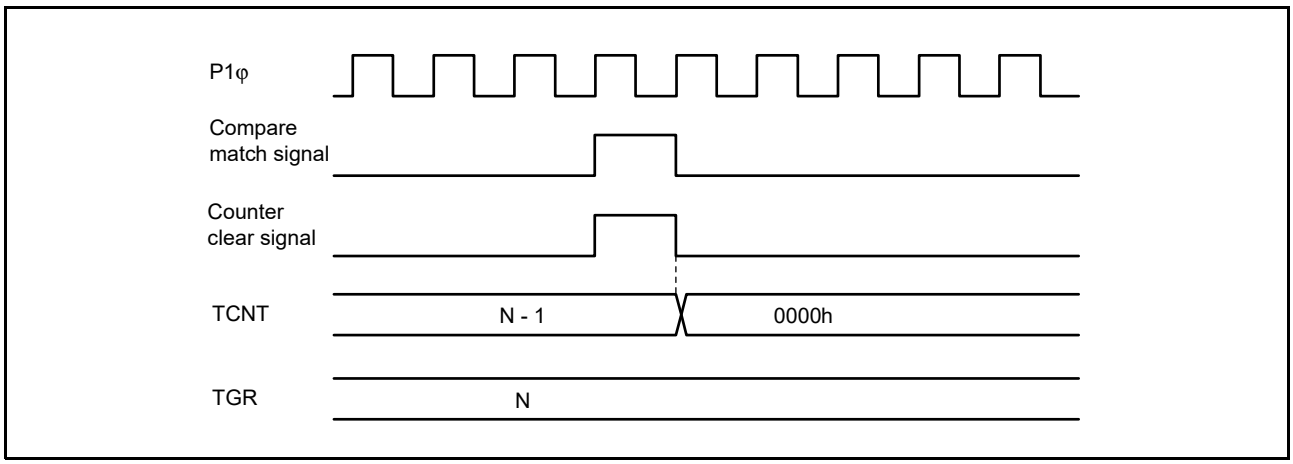


Figure 10.119 Counter Clear Timing (Compare Match) (MTU5)

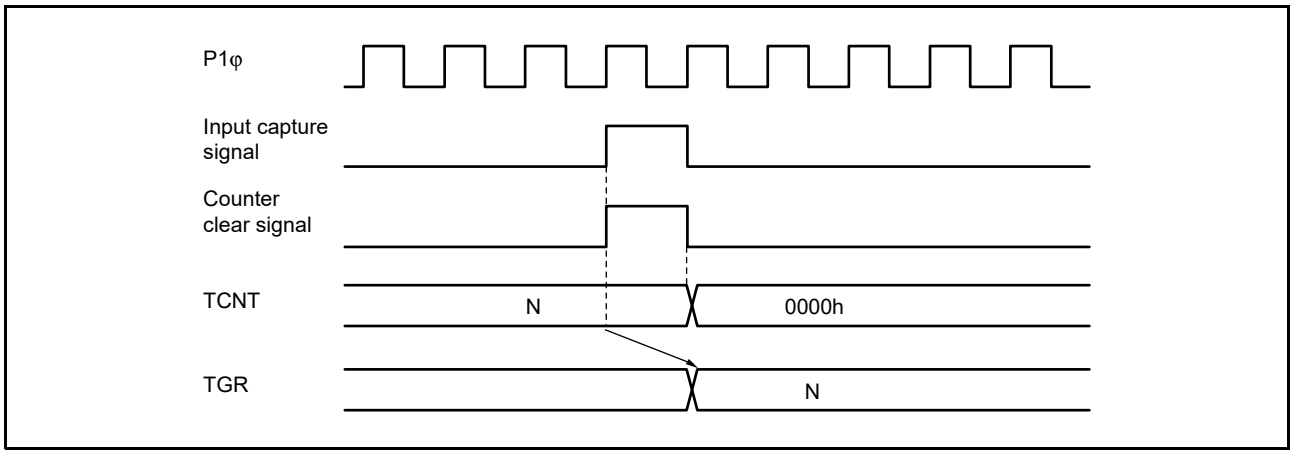


Figure 10.120 Counter Clear Timing (Input Capture) (MTU0 to MTU8)

(5) Buffer Operation Timing

Figure 10.121 to Figure 10.123 show the timing in buffer operation.

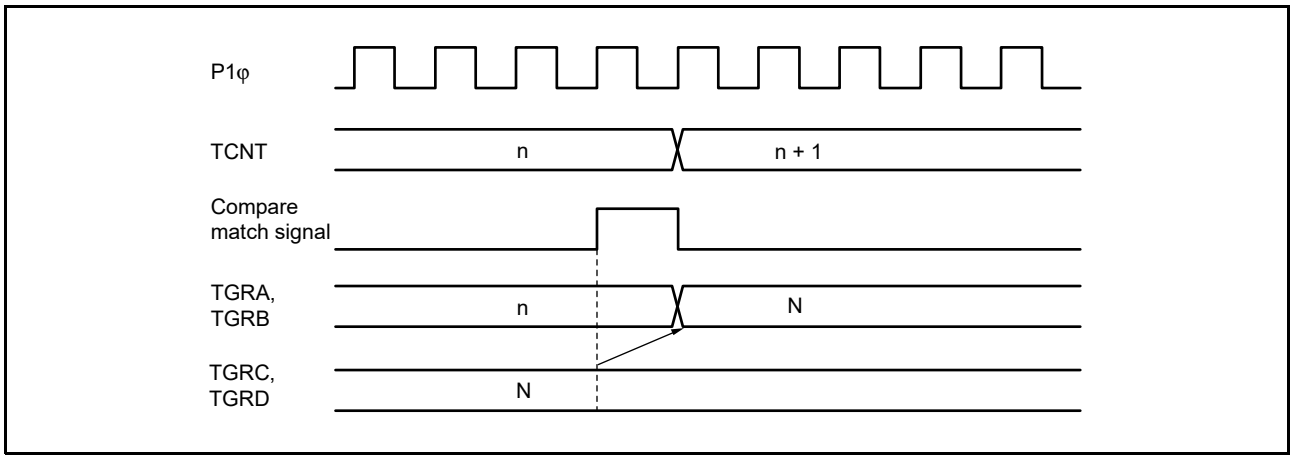


Figure 10.121 Buffer Operation Timing (Compare Match)

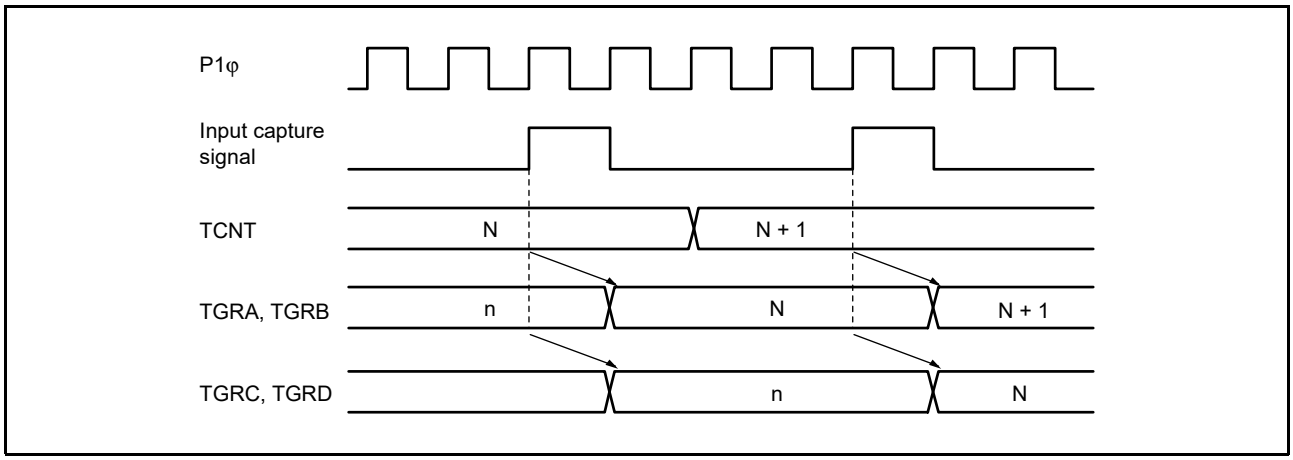


Figure 10.122 Buffer Operation Timing (Input Capture)

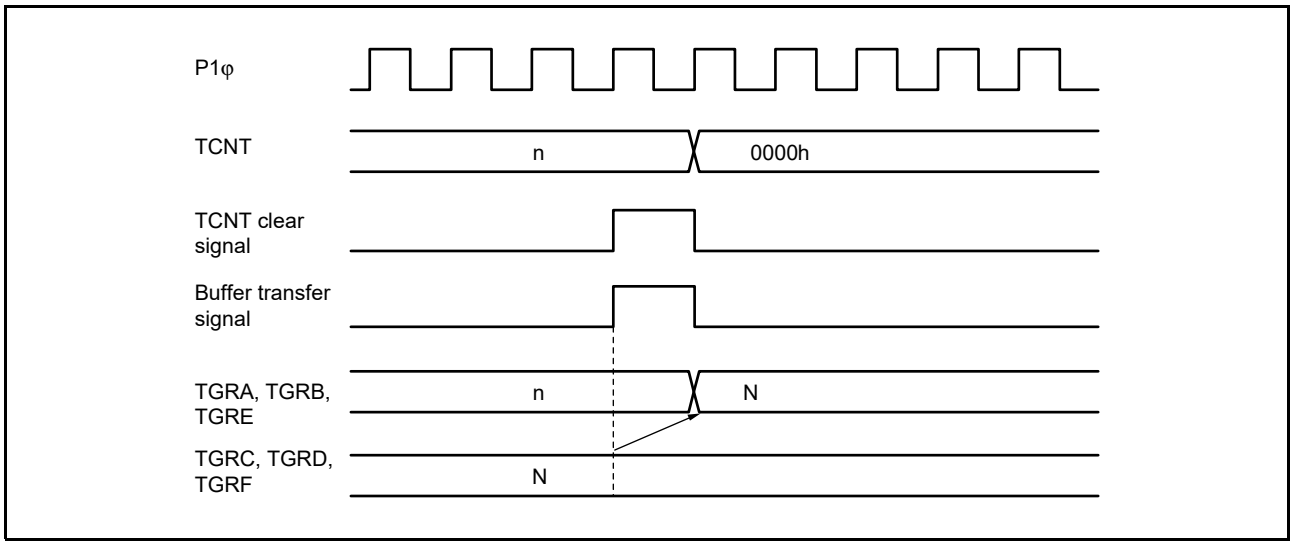


Figure 10.123 Buffer Operation Timing (When TCNT Cleared)

(6) Buffer Transfer Timing (Complementary PWM Mode)

Figure 10.124 to Figure 10.126 show the buffer transfer timing in complementary PWM mode.

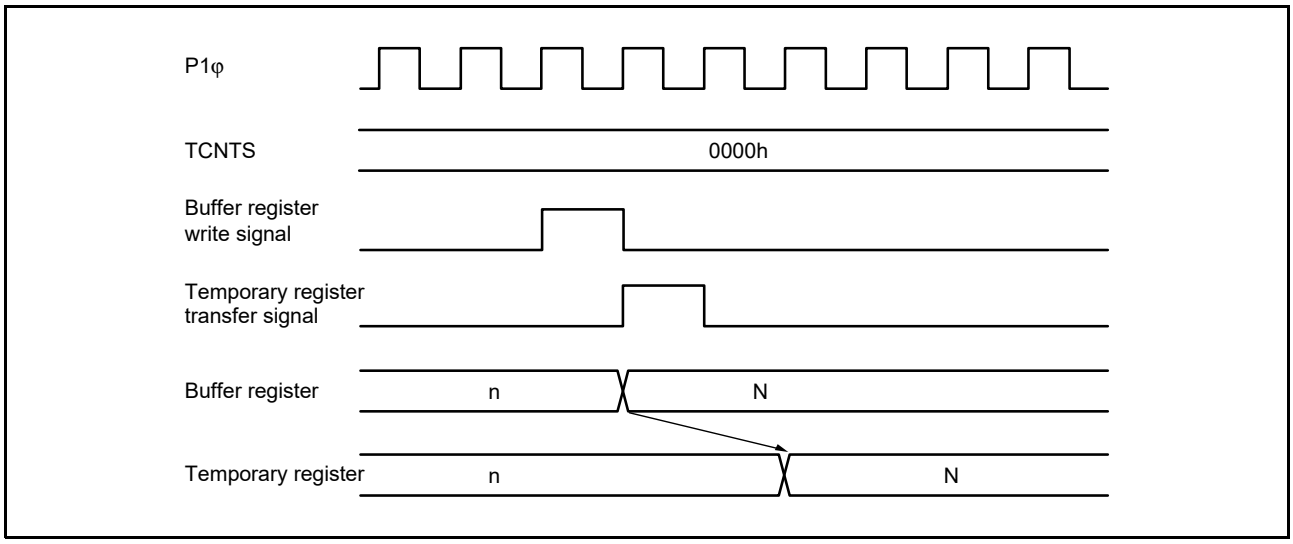


Figure 10.124 Transfer Timing from Buffer Register to Temporary Register (TCNTSA Stopped)

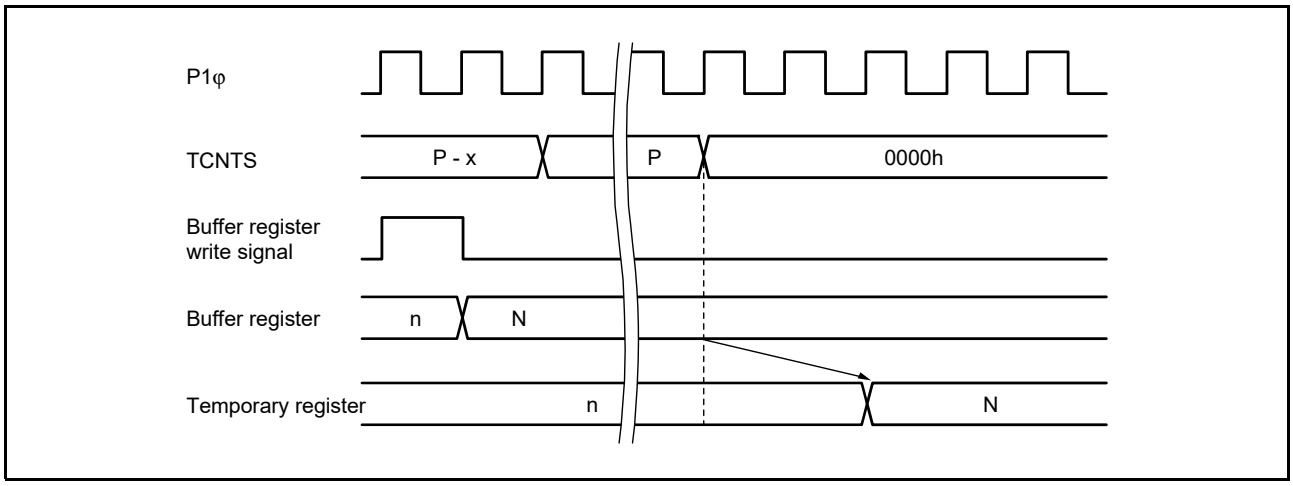


Figure 10.125 Transfer Timing from Buffer Register to Temporary Register (TCNTSA Operating)

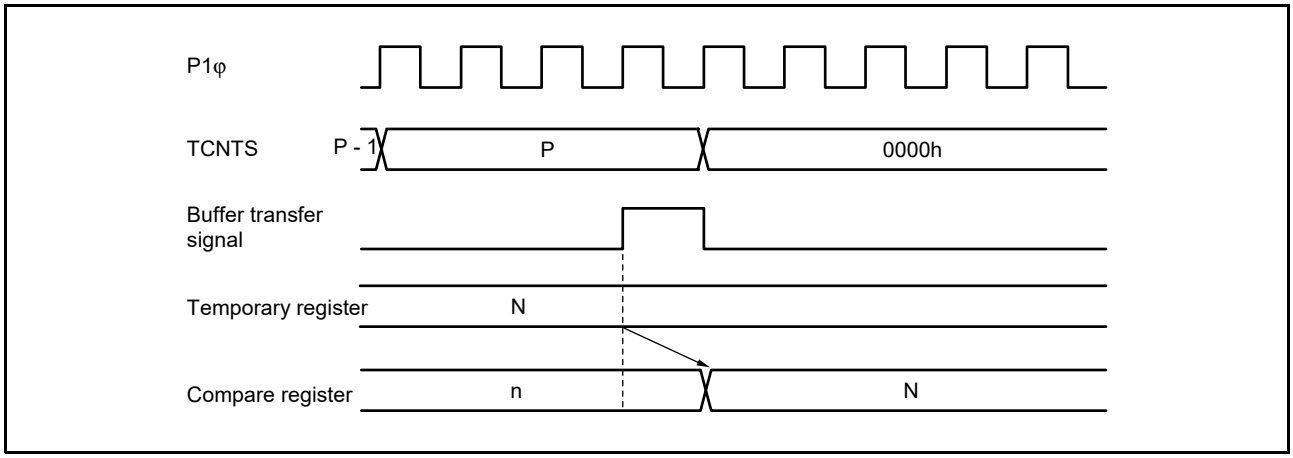


Figure 10.126 Transfer Timing from Temporary Register to Compare Register

10.5.2 Interrupt Signal Timing

(1) TGI Interrupt Timing by Compare Match

Figure 10.127 and Figure 10.128 show the TGI interrupt request signal timing when a compare match occurs.

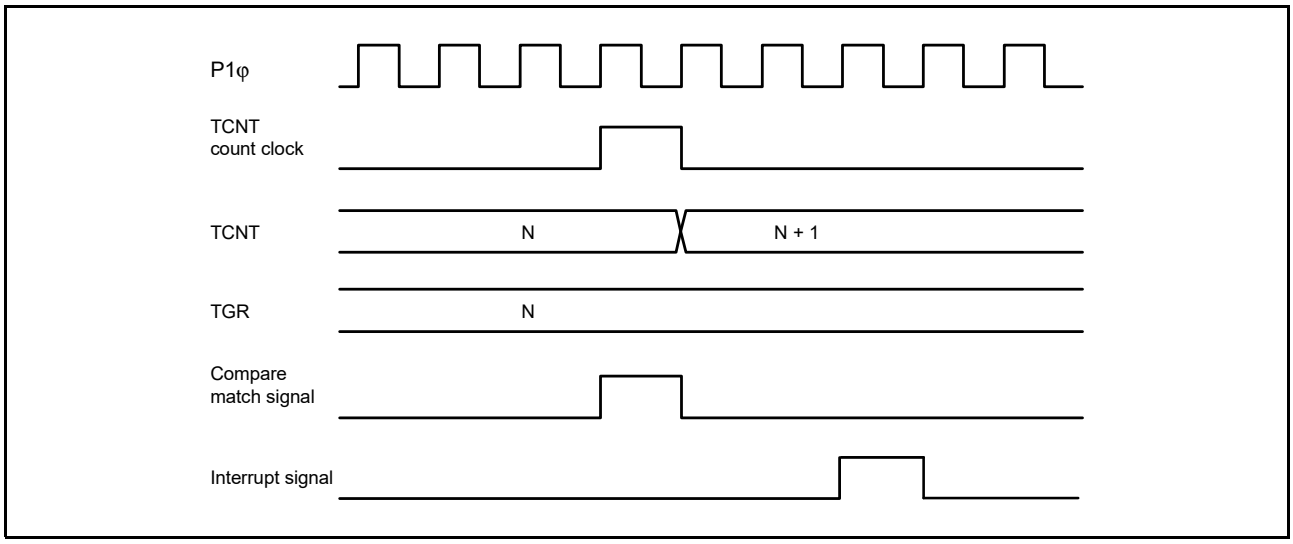


Figure 10.127 TGI Interrupt Timing (Compare Match) (MTU0 to MTU4 and MTU6 to MTU8)

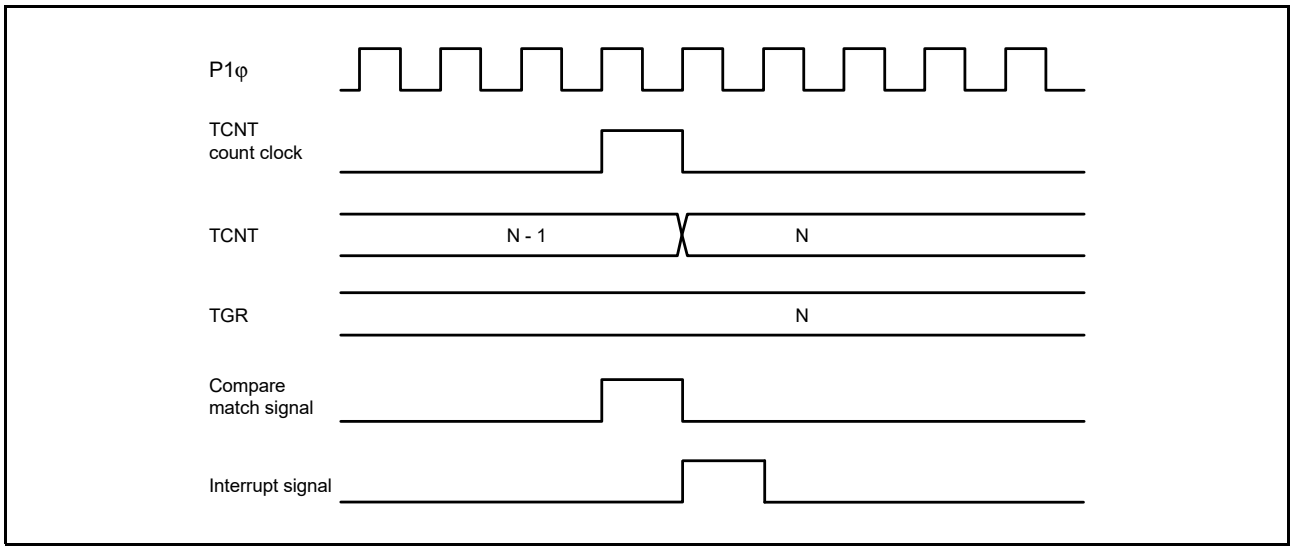


Figure 10.128 TGI Interrupt Timing (Compare Match) (MTU5)

(2) TGI Interrupt Timing by Input Capture

Figure 10.129 and Figure 10.130 show the TGI interrupt request signal timing when an input capture occurs.

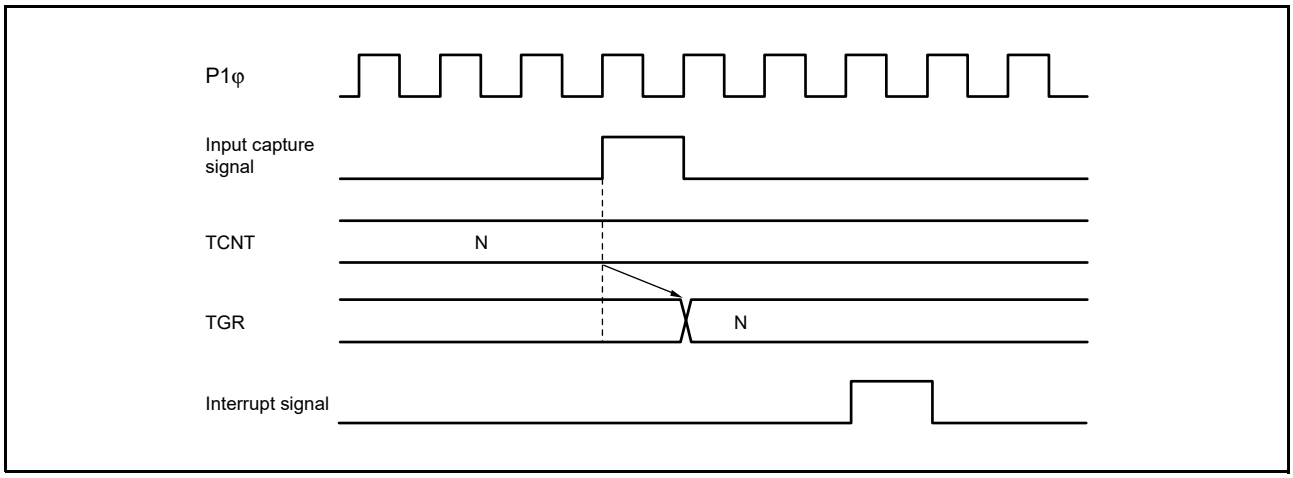


Figure 10.129 TGI Interrupt Timing (Input Capture) (MTU0 to MTU4 and MTU6 to MTU8)

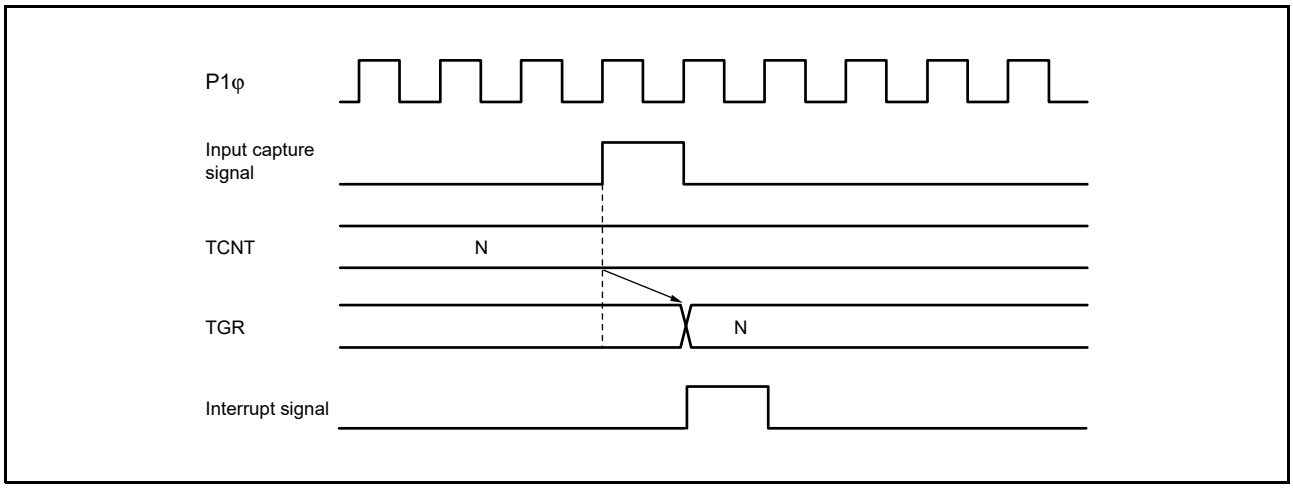


Figure 10.130 TGI Interrupt Timing (Input Capture) (MTU5)

(3) TCFV and TCFU Interrupt Timing

Figure 10.131 shows the TCIV interrupt request signal timing when an overflow is generated.
Figure 10.132 shows the TCIU interrupt request signal timing when an underflow is generated.

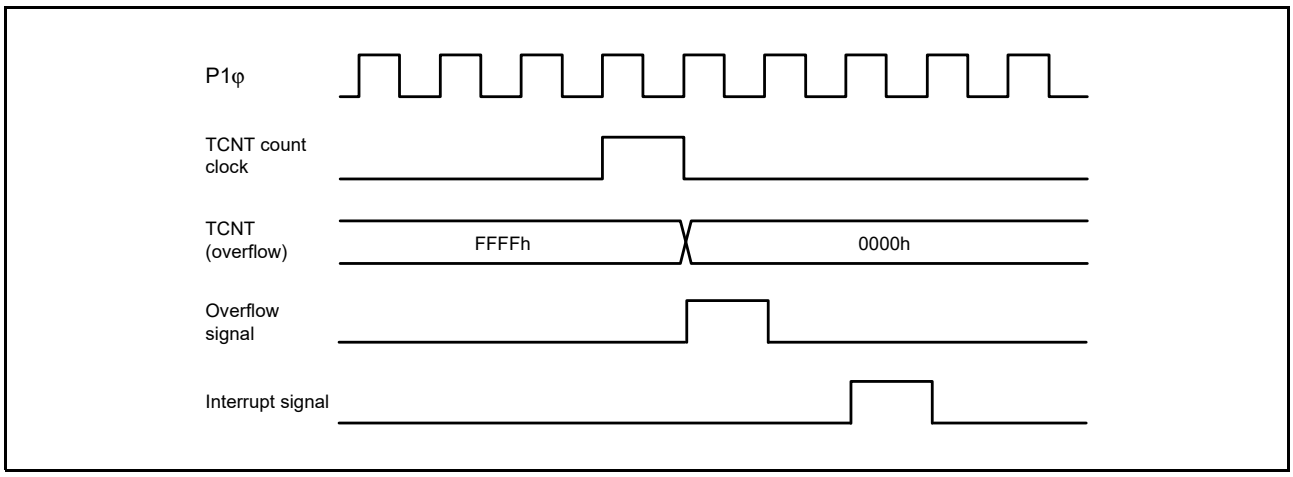


Figure 10.131 TCIV Interrupt Timing

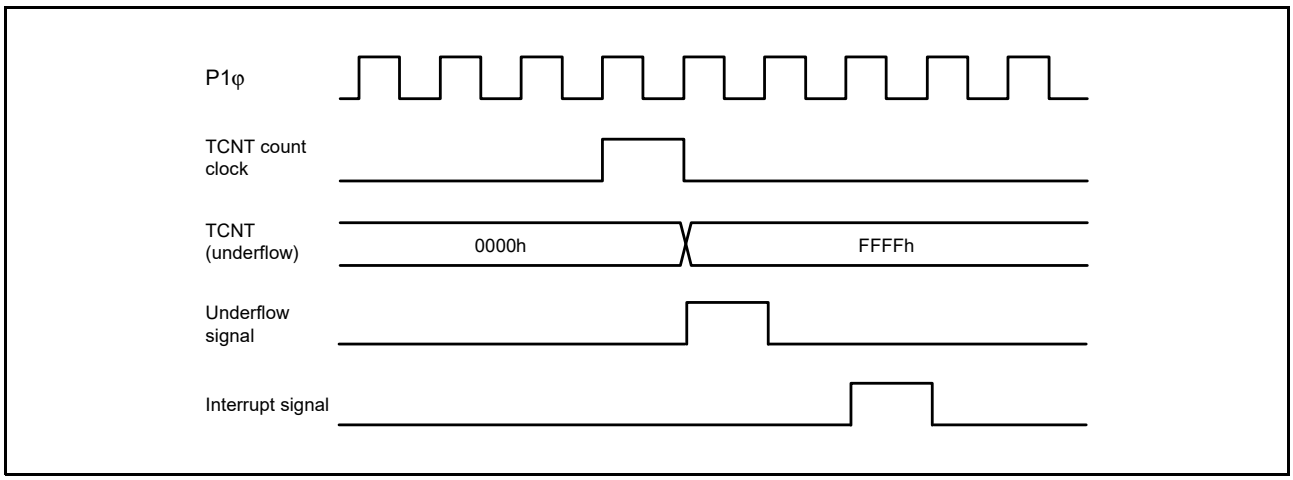


Figure 10.132 TCIU Interrupt Timing

10.6 Usage Notes

10.6.1 Module Stop Function Setting

MTU operation can be disabled or enabled using the module stop control register. MTU operation is stopped with the initial setting. Register access is enabled by releasing the module clock stop state. For details, refer to section 52, Power-Down Modes.

10.6.2 Count Clock Restrictions

The count clock source pulse width must be at least three P1φ clock cycles for single-edge detection, and at least five P1φ clock cycles for both-edge detection. The MTU will not operate properly at narrower pulse widths.
In phase counting mode, the phase difference and overlap between the two input clocks must be at least three P1φ clock cycles, and the pulse width must be at least five P1φ clock cycles. Figure 10.133 shows the input clock conditions in phase counting mode.

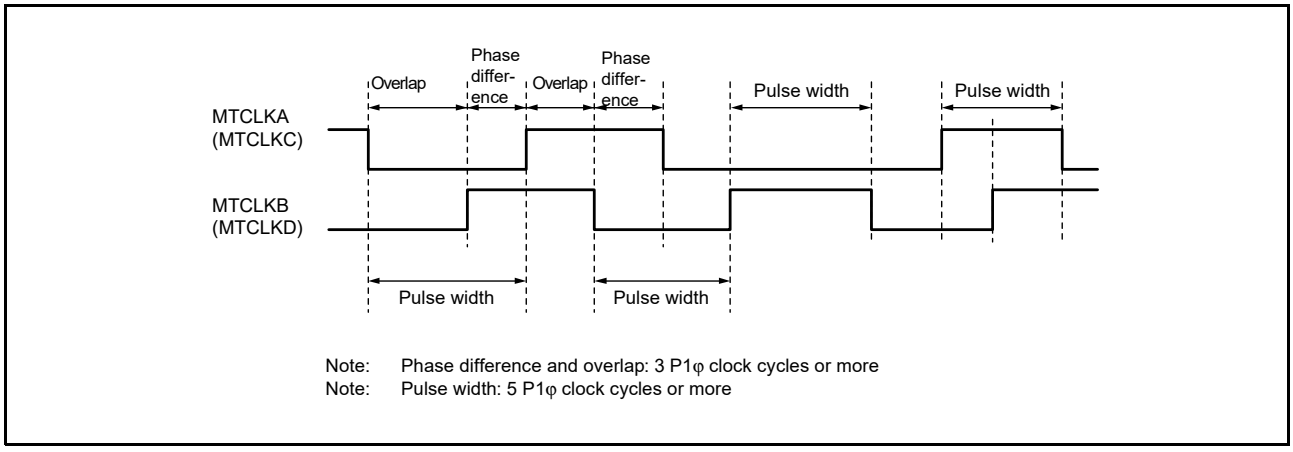


Figure 10.133 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode

10.6.3 Note on Cycle Setting

When counter clearing on compare match is set, TCNT is cleared in the final state in which it matches the TGR value (the point at which TCNU updates the matched count value). Consequently, the actual counter frequency is given by the following formula:

- MTU0 to MTU4 and MTU6 to MTU8

$$f = \frac{CNTCLK}{N + 1}$$

- MTU5

$$f = \frac{CNTCLK}{N}$$

f: Counter frequency
CNTCLK: The count clock frequency set by TCR.TPSC[2:0] and TCR2.TPSC2[2:0]
N: TGR setting

10.6.4 Contention between TCNT Write and Clear Operations

If the counter clear signal is generated in the TCNT write cycle, TCNT clearing takes precedence and TCNT write operation is not performed.
Figure 10.134 shows the timing in this case.

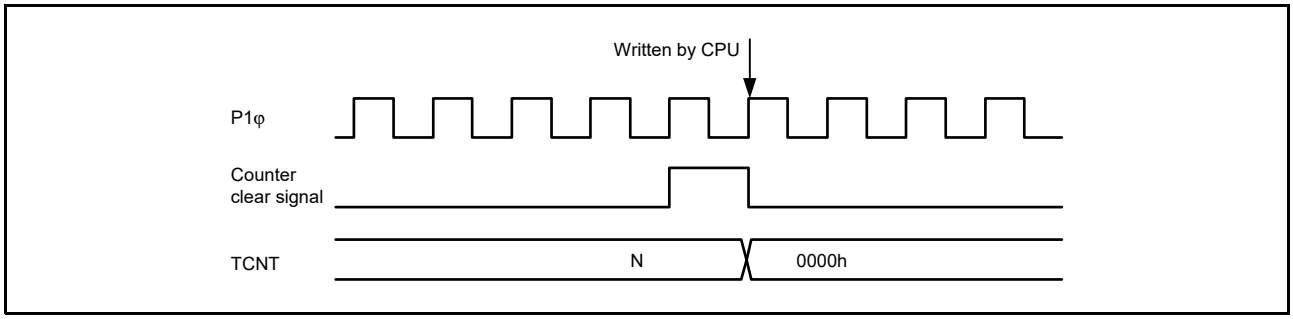


Figure 10.134 Contention between TCNT Write and Clear Operations

10.6.5 Contention between TCNT Write and Increment Operations

If incrementing occurs in a TCNT write cycle, TCNT write operation takes precedence and TCNT is not incremented.
Figure 10.135 shows the timing in this case.

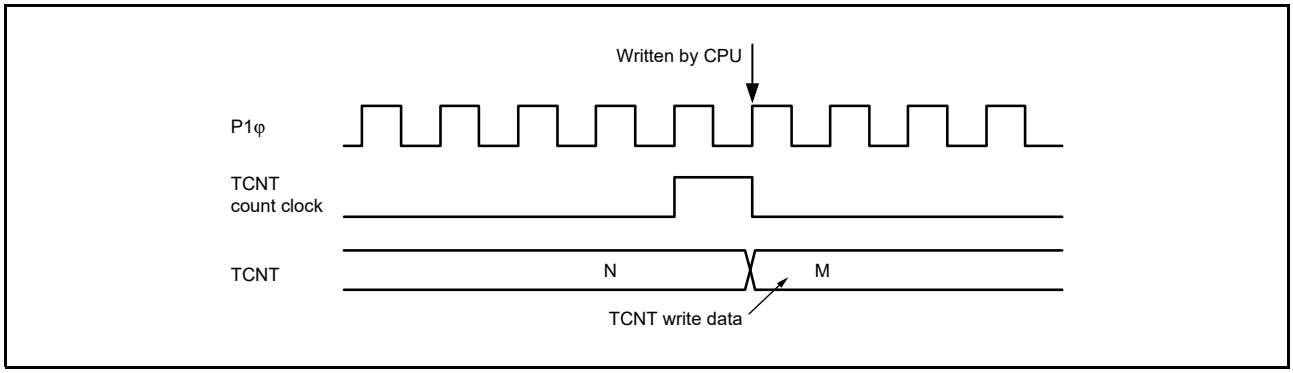


Figure 10.135 Contention between TCNT Write and Increment Operations

10.6.6 Contention between TGR Write Operation and Compare Match

If a compare match occurs in a TGR write cycle, TGR write operation is executed and the compare match signal is also generated.

Figure 10.136 shows the timing in this case.

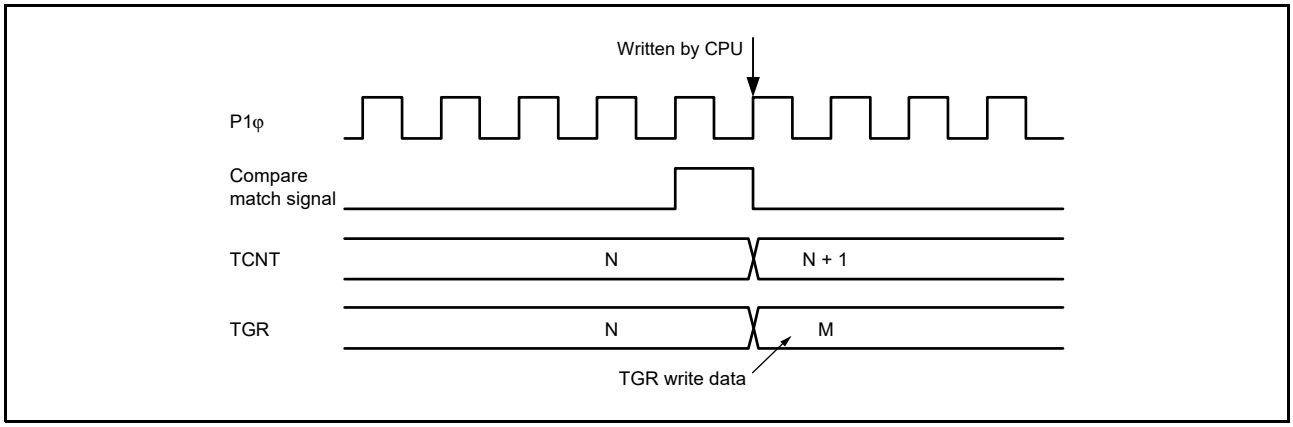


Figure 10.136 Contention between TGR Write Operation and Compare Match

10.6.7 Contention between Buffer Register Write Operation and Compare Match

If a compare match occurs in the T2 state in a TGR write cycle, the data before write operation is transferred to TGR by the buffer operation.

Figure 10.137 shows the timing in this case.

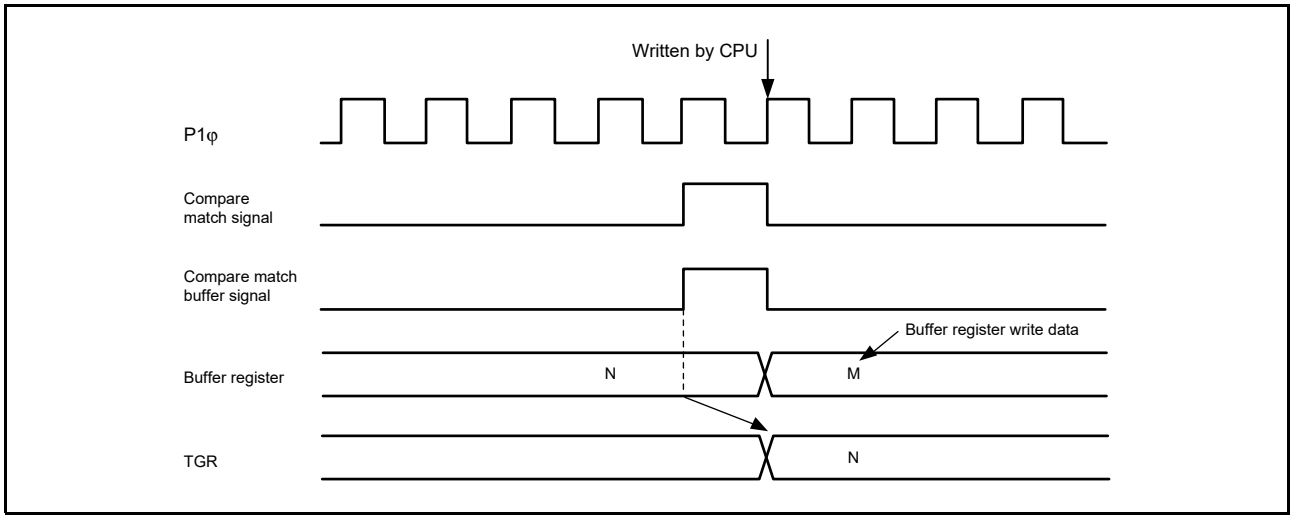


Figure 10.137 Contention between Buffer Register Write Operation and Compare Match

10.6.8 Contention between Buffer Register Write and TCNT Clear Operations

When the buffer transfer timing is set to the TCNT clear timing by the timer buffer transfer mode register (TBTM), if TCNT is cleared during the TGR write cycle, the data before write operation is transferred to TGR by the buffer operation.
Figure 10.138 shows the timing in this case.

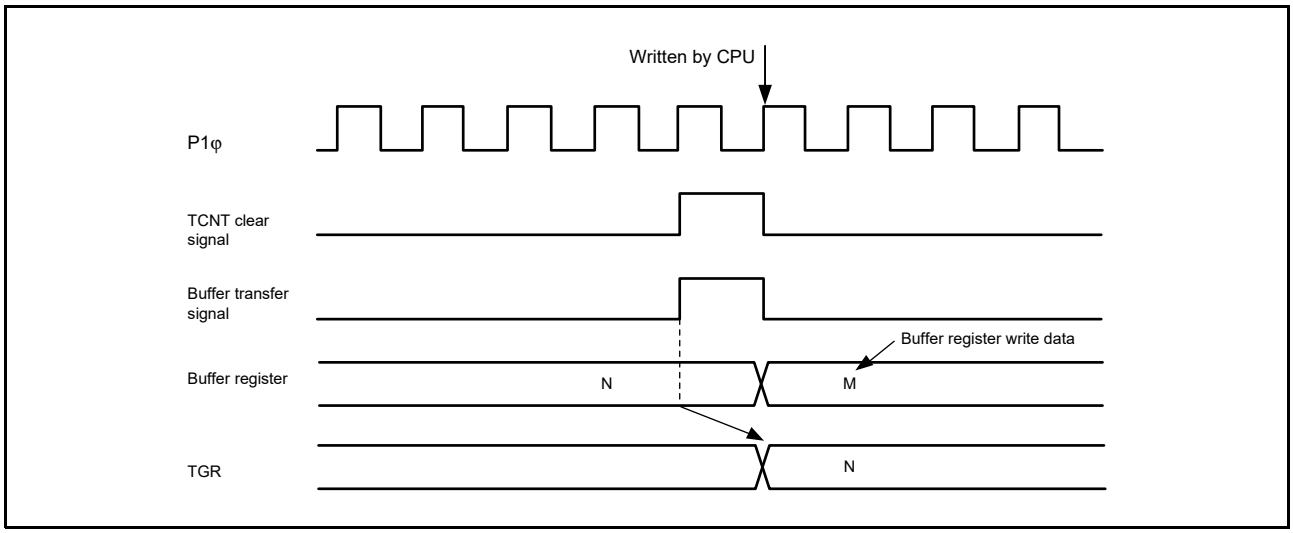


Figure 10.138 Contention between Buffer Register Write and TCNT Clear Operations

10.6.9 Contention between TGR Read Operation and Input Capture

If an input capture signal is generated in a TGR read cycle, the data before input capture transfer is read.
Figure 10.139 shows the timing in this case.

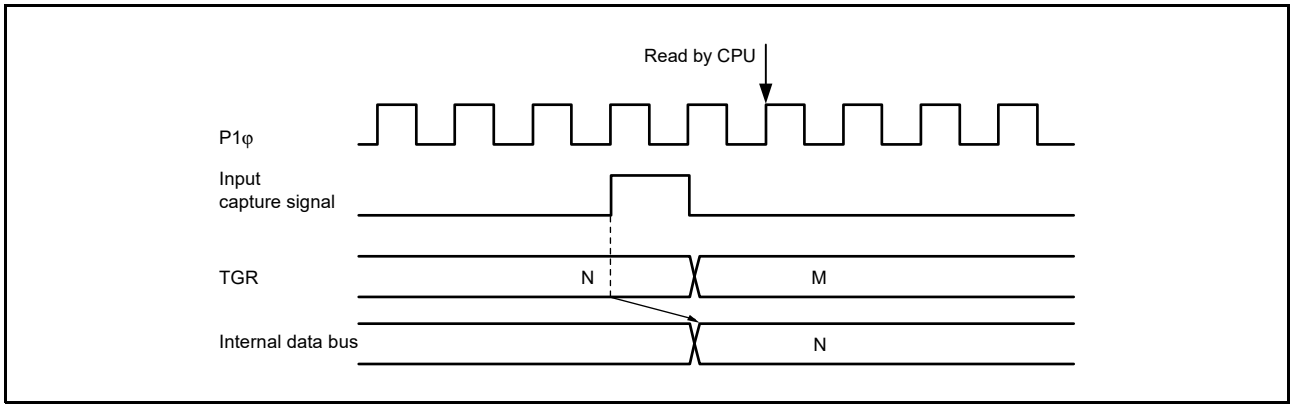


Figure 10.139 Contention between TGR Read Operation and Input Capture (MTU0 to MTU8)

10.6.10 Contention between TGR Write Operation and Input Capture

If an input capture signal is generated in the TGR write cycle, the input capture operation takes precedence and the TGR write operation is not performed in MTU0 to MTU4 and MTU6 to MTU8. In MTU5, the TGR write operation is performed and the input capture signal is generated.
Figure 10.140 and Figure 10.141 show the timing in this case.

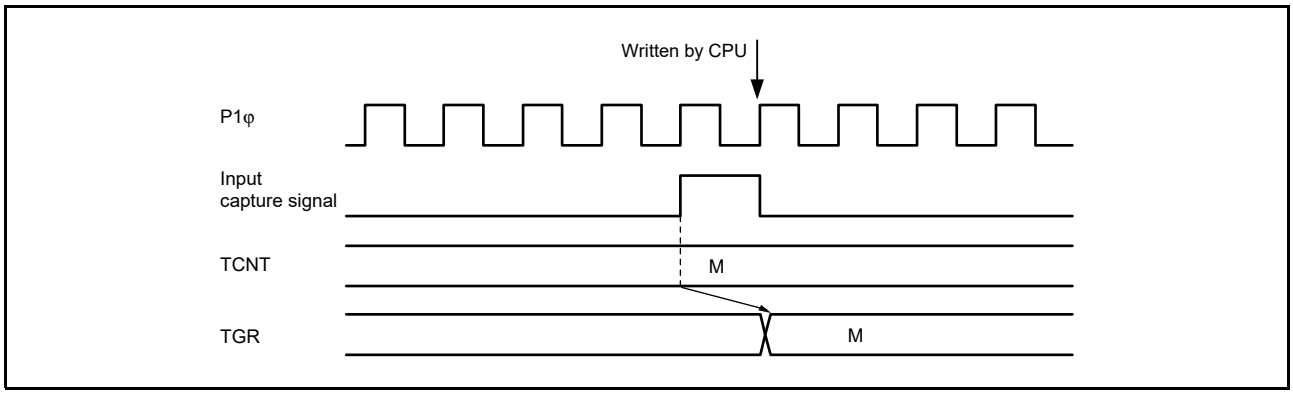


Figure 10.140 Contention between TGR Write Operation and Input Capture (MTU0 to MTU4 and MTU6 to MTU8)

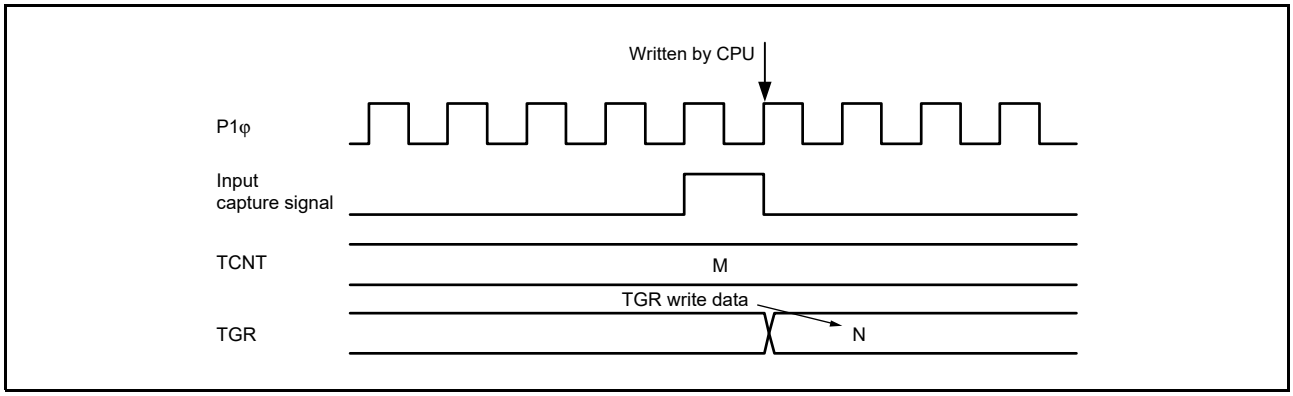


Figure 10.141 Contention between TGR Write Operation and Input Capture (MTU5)

10.6.11 Contention between Buffer Register Write Operation and Input Capture

If an input capture signal is generated in the buffer register write cycle, the buffer operation takes precedence and the buffer register write operation is not performed.
Figure 10.142 shows the timing in this case.

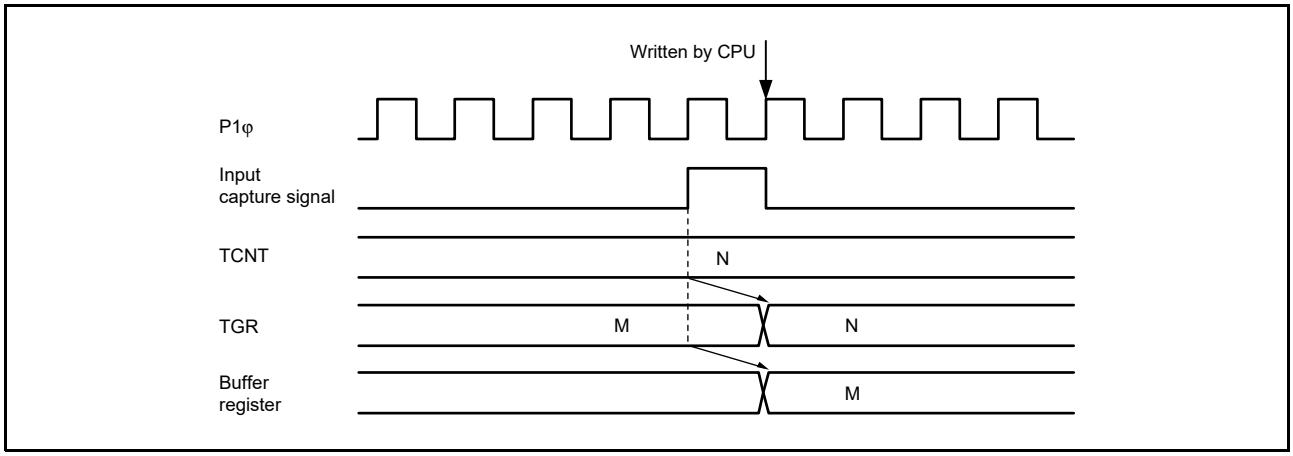


Figure 10.142 Contention between Buffer Register Write Operation and Input Capture

10.6.12 Contention between MTU2.TCNT Write Operation and Overflow/Underflow in Cascaded Operation

With timer counters TCNT in MTU1 and MTU2 cascaded, when a contention occurs between the counting by TCNT in MTU1 (a TCNT overflow/underflow in MTU2) and the writing to TCNT in MTU2, the TCNT write operation is performed in MTU2 and the TCNT count signal in MTU1 is disabled. In this case, if TGRA in MTU1 works as a compare match register and there is a match between the values of TGRA and TCNT in MTU1, a compare match signal is issued.

Furthermore, when the TCNT count clock in MTU1 is selected as the input capture source of MTU0, TGRA to TGRD in MTU0 work in input capture mode. In addition, when the TGRC compare match/input capture in MTU0 is selected as the input capture source of TGRB in MTU1, TGRB in MTU1 works in input capture mode.

Figure 10.143 shows the timing in this case.

When setting the TCNT clearing function in cascaded operation, be sure to synchronize MTU1 and MTU2.

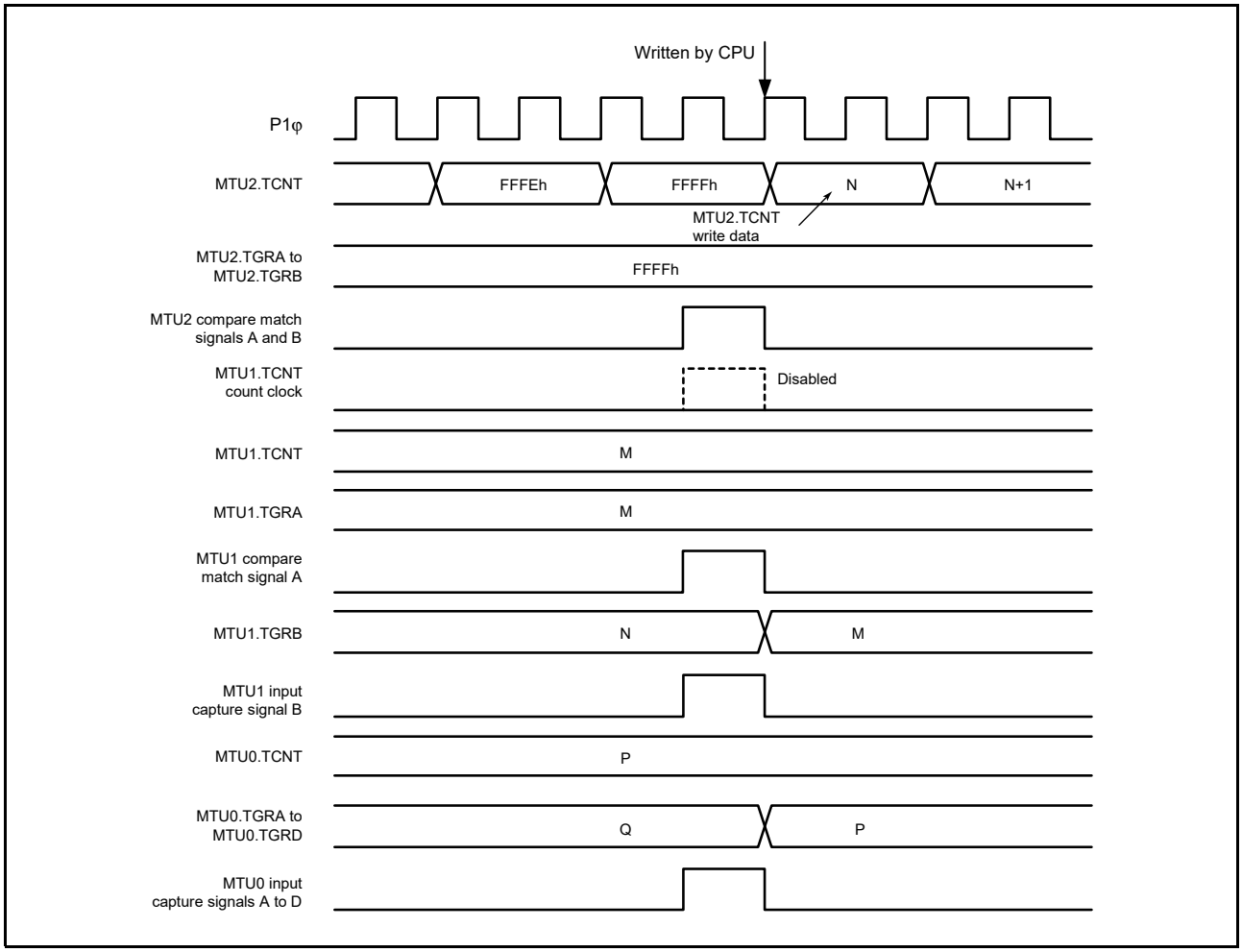


Figure 10.143 Contention between MTU2.TCNT Write Operation and Overflow/Underflow in Cascaded Operation

10.6.13 Counter Value When Count Operation is Stopped in Complementary PWM Mode

When counting operation in MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT) is stopped in complementary PWM mode, the MTU3.TCNT (MTU6.TCNT) value is set to the timer dead time register (TDDRA (TDDRB)) value and MTU4.TCNT (MTU7.TCNT) is set to 0000h.
When operation is restarted in complementary PWM mode, counting begins automatically from the initial setting state. Figure 10.144 shows this operation.
When counting begins in another operating mode, be sure to make initial settings in MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT).

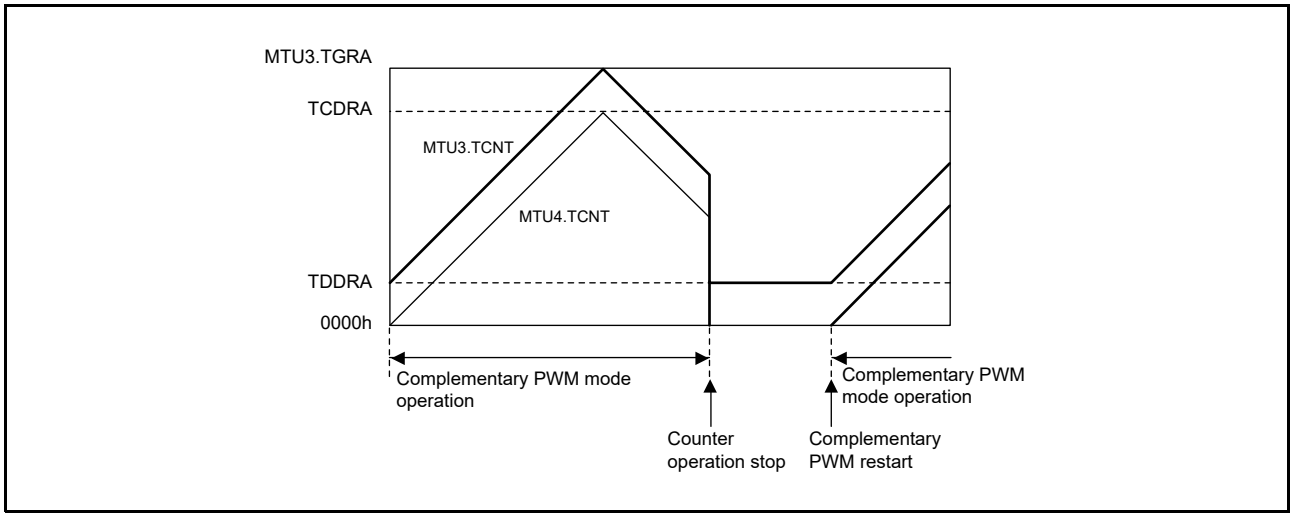


Figure 10.144 Counter Value When Stopped in Complementary PWM Mode

10.6.14 Buffer Operation Setting in Complementary PWM Mode

When modifying the PWM cycle set register (MTU3.TGRA or MTU6.TGRA), timer cycle data register (TCDRA or TCDRB), and duty set registers (MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB (MTU6.TGRB, MTU7.TGRA, and MTU7.TGRB)) in complementary PWM mode, be sure to use buffer operation. In addition, set the BFA and BFB bits in MTU4.TMDR1 (MTU7.TMDR1) to 0. The MTIOC4C (MTIOC7C) pin cannot output waveforms if the BFA bit in MTU4.TMDR1 (MTU7.TMDR1) is set to 1. Likewise, the MTIOC4D (MTIOC7D) pin cannot output waveforms if the BFB bit in MTU4.TMDR1 (MTU7.TMDR1) is set to 1.
In complementary PWM mode, buffer operation in MTU3 and MTU4 (or MTU6 and MTU7) depends on the settings in bits BFA and BFB of MTU3.TMDR1 (MTU6.TMDR1). When the BFA bit in MTU3.TMDR1 (MTU6.TMDR1) is set to 1, MTU3.TGRC (MTU6.TGRC) functions as a buffer register for MTU3.TGRA (MTU6.TGRA). At the same time, MTU4.TGRC (MTU7.TGRC) functions as a buffer register for MTU4.TGRA (MTU7.TGRA), and TCBRA (TCBRB) functions as a buffer register for TCDRA (TCDRB).

10.6.15 Buffer Operation and Compare Match in Reset-Synchronized PWM Mode

When setting buffer operation in reset-synchronized PWM mode, set the BFA and BFB bits in MTU4.TMDR1 (MTU7.TMDR1) to 0. The MTIOC4C (MTIOC7C) pin cannot output waveforms if the BFA bit in MTU4.TMDR1 (MTU7.TMDR1) is set to 1. Likewise, the MTIOC4D (MTIOC7D) pin cannot output waveforms if the BFB bit in MTU4.TMDR1 (MTU7.TMDR1) is set to 1.

In reset-synchronized PWM mode, buffer operation in MTU3 and MTU4 (or MTU 6 and MTU7) depends on the settings in the BFA and BFB bits of MTU3.TMDR1 (MTU6.TMDR1). For example, if the BFA bit in MTU3.TMDR1 (MTU6.TMDR1) is set to 1, MTU3.TGRC (MTU6.TGRC) functions as a buffer register for MTU3.TGRA (MTU6.TGRA). At the same time, MTU4.TGRC (MTU7.TGRC) functions as a buffer register for MTU4.TGRA (MTU7.TGRA).

While the MTU3.TGRC (MTU6.TGRC) and MTU3.TGRD (MTU6.TGRD) are operating as buffer registers, a TGImn interrupt (m = C or D; n = 3, 4, 6, or 7) is not generated.

Figure 10.145 shows an example of MTU3.TGR (MTU6.TGR), MTU4.TGR (MTU7.TGR), MTIOC3 (MTIOC6), and MTIOC4 (MTIOC7) operation with the BFA and BFB bits in MTU3.TMDR1 (MTU6.TMDR1) set to 1 and the BFA and BFB bits in MTU4.TMDR1 (MTU7.TMDR1) set to 0.

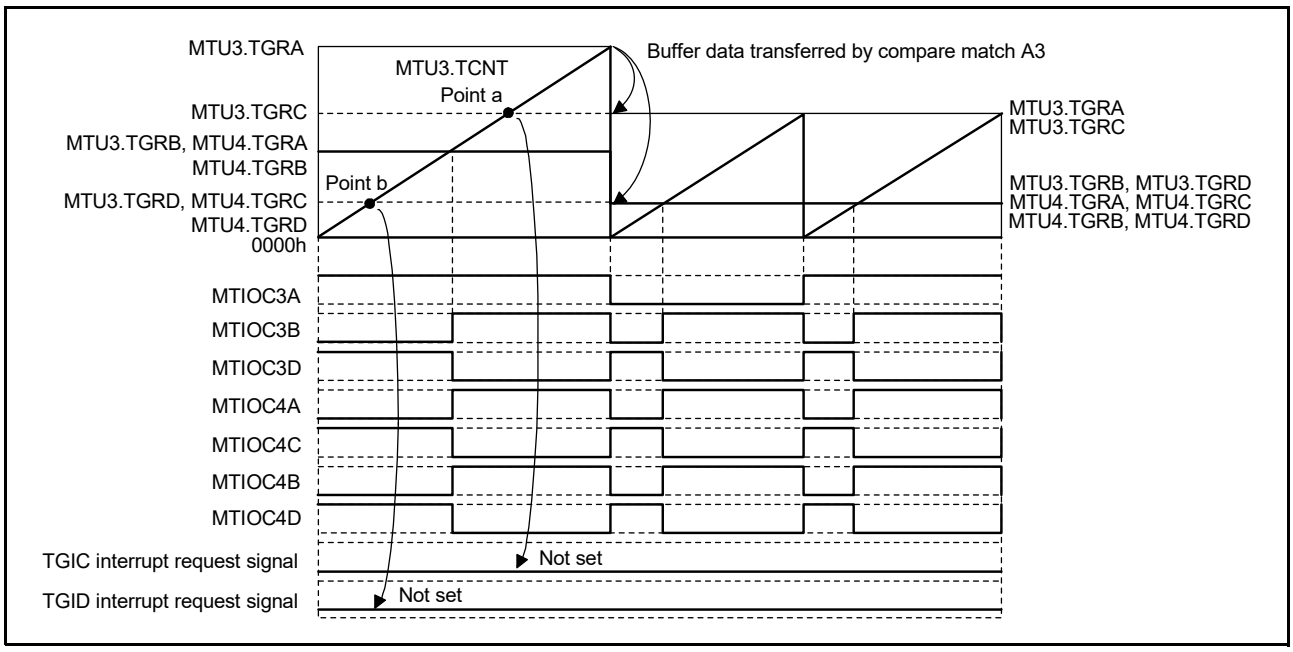


Figure 10.145 Buffer Operation and Compare Match in Reset-Synchronized PWM Mode

10.6.16 Overflow in Reset-Synchronized PWM Mode

After reset-synchronized PWM mode is selected, the TCNT counters in MTU3 and MTU4 (MTU6 and MTU7) start counting when the CST3 (CST6) bit of TSTRA (TSTRB) is set to 1. In this state, the TCNT count clock source and count edge in MTU4 (MTU7) are determined by the setting of TCR in MTU3 (MTU6).

In reset-synchronized PWM mode, with cycle register TGRA in MTU3 (MTU6) set to FFFFh and the TGRA compare match in MTU3 (MTU6) selected as the counter clearing source, the TCNT counters in MTU3 and MTU4 (MTU6 and MTU7) count up to FFFFh, then a compare match occurs with TGRA in MTU3 (MTU6), and the TCNT counters in MTU3 and MTU4 (MTU6 and MTU7) are both cleared. In this case, a TCIVn interrupt (n = 3, 4, 6, or 7) is not generated.

Figure 10.146 shows an example of the operation in reset-synchronized PWM mode with cycle register TGRA in MTU3 (MTU6) set to FFFFh and the TGRA compare match in MTU3 (MTU6) specified for the counter clearing source.

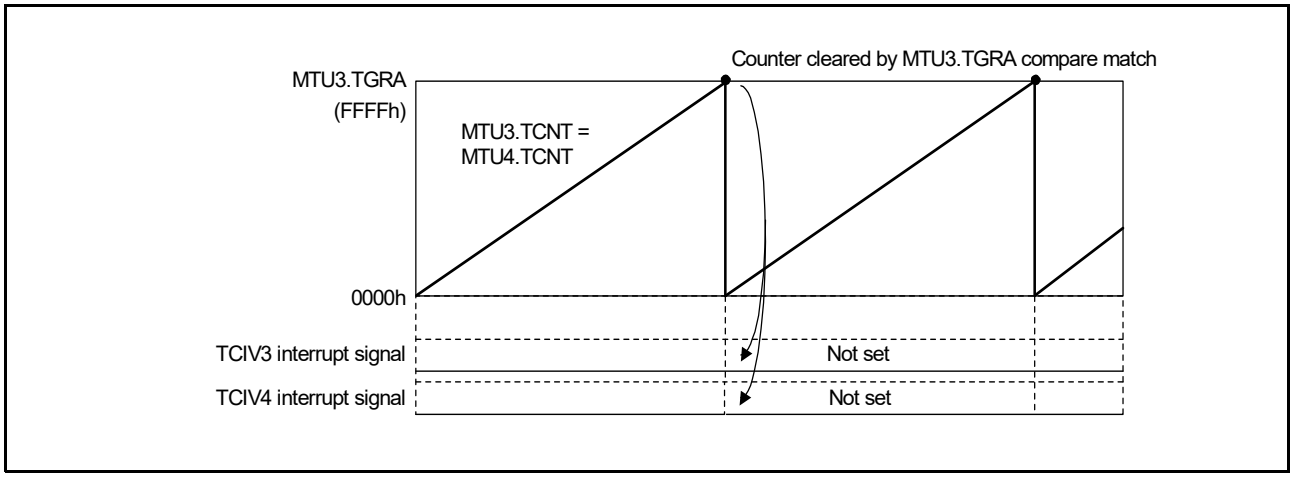


Figure 10.146 Overflow in Reset-Synchronized PWM Mode

10.6.17 Contention between Overflow/Underflow and Counter Clearing

If an overflow/underflow and counter clearing occur simultaneously, a TCIVn interrupt (n = 0 to 4 or 6 to 8) nor a TCIUn interrupt (n = 1 or 2) is not generated and TCNT clearing takes precedence.

Figure 10.147 shows the operation timing when a TGR compare match is specified as the clearing source and TGR is set to FFFFh.

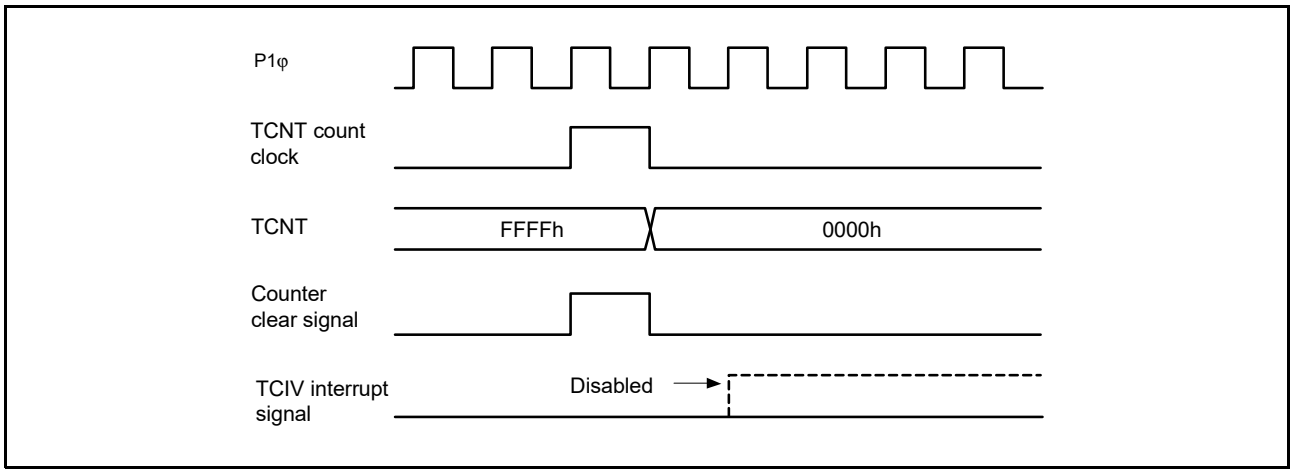


Figure 10.147 Contention between Overflow and Counter Clearing

10.6.18 Contention between TCNT Write Operation and Overflow/Underflow

If TCNT counts up or down in a TCNT write cycle and an overflow or an underflow occurs, the TCNT write operation takes precedence. A TCIVn interrupt (n = 0 to 4 or 6 to 8) nor a TCIUn interrupt (n = 1 or 2) is not generated.
Figure 10.148 shows the operation timing when there is contention between TCNT write operation and overflow.

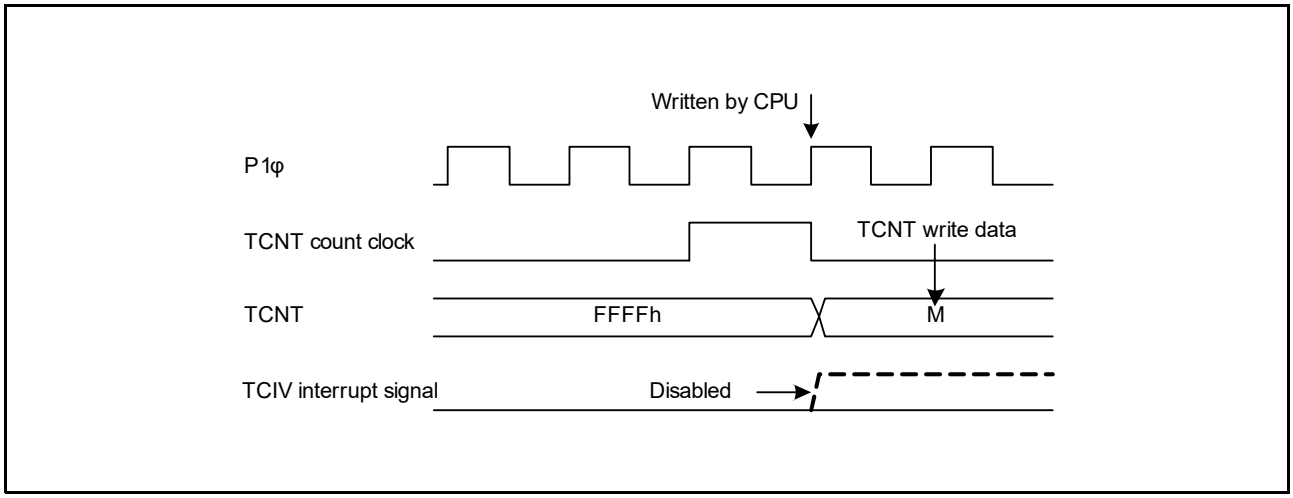


Figure 10.148 Contention between TCNT Write Operation and Overflow

10.6.19 Note on Transition from Normal Mode or PWM Mode 1 to Reset-Synchronized PWM Mode

When making a transition from normal mode or PWM mode 1 to reset-synchronized PWM mode in MTU3 and MTU4 (or MTU6 and MTU7), if the counter is stopped while the output pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, MTIOC4D, MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7C, MTIOC7B, and MTIOC7D) are held at a high level and then operation is started after a transition to reset-synchronized PWM mode, the initial pin output will not be correct.
When making a transition from normal mode to reset-synchronized PWM mode, write 11h to MTU3.TIORH, MTU3.TIORL, MTU4.TIORH, and MTU4.TIORL (MTU6.TIORH, MTU6.TIORL, MTU7.TIORH, and MTU7.TIORL) to initialize the output pin state to a low level, then set the registers to the initial value (00h) before making the mode transition.
When making a transition from PWM mode 1 to reset-synchronized PWM mode, switch to normal mode, initialize the output pin state to a low level, and then set the registers to the initial value (00h) before making the transition to reset-synchronized PWM mode.

10.6.20 Output Level in Complementary PWM Mode and Reset-Synchronized PWM Mode

When MTU3 and MTU4 (or MTU6 and MTU7) are in complementary PWM mode or reset-synchronized PWM mode, the PWM waveform output level is determined by the OLSP and OLSN bits in the timer output control register (TOCR1A or TOCR1B). In complementary PWM mode or reset-synchronized PWM mode, TIOR should be set to 00h. The output level in negative phase when the TDER bit in TDERA (TDERB) is set to 0 in complementary PWM mode (the dead time is not generated) does not depend on the setting of the OLSN bit in TOCR1A (TOCR1B). It is equivalent to the inverted level of positive phase output based on the setting of the OLSP bit in TOCR1A (TOCR1B).

10.6.21 Simultaneous Input Capture in MTU1.TCNT and MTU2.TCNT in Cascade Connection

When timer counters 1 and 2 (MTU1.TCNT and MTU2.TCNT) operate as a 32-bit counter in cascade connection, the cascaded counter value cannot be captured successfully in some cases even if input-capture input is simultaneously done to MTIOC1A and MTIOC2A or to MTIOC1B and MTIOC2B. This is because the input timing of MTIOC1A and MTIOC2A or to MTIOC1B and MTIOC2B may not be the same when external input-capture signals input into MTU1.TCNT and MTU2.TCNT are taken in synchronization with the internal clock.
For example, MTU1.TCNT (the counter for upper 16 bits) does not capture the count-up value by an overflow from MTU2.TCNT (the counter for lower 16 bits) but captures the count value before the up-counting. In this case, the values of MTU1.TCNT = FFF1h and MTU2.TCNT = 0000h should be transferred to MTU1.TGRA and MTU2.TGRA or to MTU1.TGRB and MTU2.TGRB, but the values of MTU1.TCNT = FFF0h and MTU2.TCNT = 0000h are erroneously transferred.
The MTU has a function that allows simultaneous capture of MTU1.TCNT and MTU2.TCNT with a single input capture input. This function can be used to read the 32-bit counter such that MTU1.TCNT and MTU2.TCNT are captured at the same time. For details, see section 10.2.11, Timer Input Capture Control Register (TICCR).

10.6.22 Interrupt-Skipping Function 2

When interrupt-skipping function 2 is in use and the difference between the values in MTU4.TADCORA and MTU4.TADCORB is small, correct counting of the number skipped may not be possible, in which case requests for A/D conversion will not be generated with the expected timing. The conditions listed below thus apply to these settings.
For MTU6 and MTU7, the same conditions apply to the settings of MTU7.TADCORA and MTU7.TADCORB.

(1) When the number skipped is zero for skipping function 2

- The difference between the values in MTU4.TADCORA and MTU4.TADCORB must be at least four.
- The interval of comparison for MTU4.TADCORA must be at least four cycles of P1φ clock (the updated value of MTU4.TADCORA is set to the previous value plus or minus at least four).
- The interval of comparison for MTU4.TADCORB must be at least four cycles of P1φ clock (the updated value of MTU4.TADCORB is set to the previous value plus or minus at least four).

(2) When the number skipped is one or more for skipping function 2

- The difference between the values in MTU4.TADCORA and MTU4.TADCORB must be at least two.
- The interval of comparison for MTU4.TADCORB must be at least two cycles of P1φ (the updated value of MTU4.TADCORB is set to the previous value plus or minus at least two).

10.6.23 Notes When Complementary PWM Mode Output Protection Function is Not Used

The complementary PWM mode output protection function is initially enabled. For details, refer to section 11, Port Output Enable 3 (POE3).

10.6.24 Notes Regarding Timer Counter (MTU5.TCNT) and Timer General Register (MTU5.TGR)

Do not set TGRj (j = U, V, or W) in MTU5 to the value of the corresponding TCNTj in MTU5 plus one while counting by TCNTj in MTU5 is stopped. If such a setting is made, a compare match will be generated even though counting is stopped.

In this case, if the value of the compare match enable bit (TGIE5j bit of TIER in MTU5) is 1 (enabled), a compare match interrupt will also be generated. If the value of the timer compare match clear register is 1 (enabled), the TCNTj counter in MTU5 is automatically cleared to 0000h when the compare match is generated, regardless of whether compare match interrupts are enabled or disabled.

10.6.25 Notes to Prevent Malfunctions in Synchronous Clearing for Complementary PWM Mode

If control of the output waveform is enabled (the WRE bit of TWCRA or TWCRB is 1) at the time of synchronous counter clearing in complementary PWM mode, satisfaction of either condition 1 or 2 below has the following effects.

- Dead time on the PWM output pins is shortened (or disappears).
- The active level is output on the negative phase PWM output pins beyond the period for active-level output.

Condition 1: In portion (10) of the initial output inhibition period in Figure 10.149, synchronous clearing occurs within the dead-time period for PWM output.

Condition 2: In portions (10) and (11) of the initial output inhibition period in Figure 10.150, synchronous clearing occurs when any condition from among TGRB in MTU3 (MTU6) ≤ TDDR (TDDRB), TGRA in MTU4 (MTU7) ≤ TDDR (TDDRB), and TGRB in MTU4 (MTU7) ≤ TDDR (TDDRB) is satisfied.

The following method avoids the above phenomena.
Ensure that synchronous clearing proceeds with the value of each comparison register (TGRB in MTU3 (MTU6), TGRA in MTU4 (MTU7), and TGRB in MTU4 (MTU7)) set to at least double the value of the TDDRA register (TDDRB register).

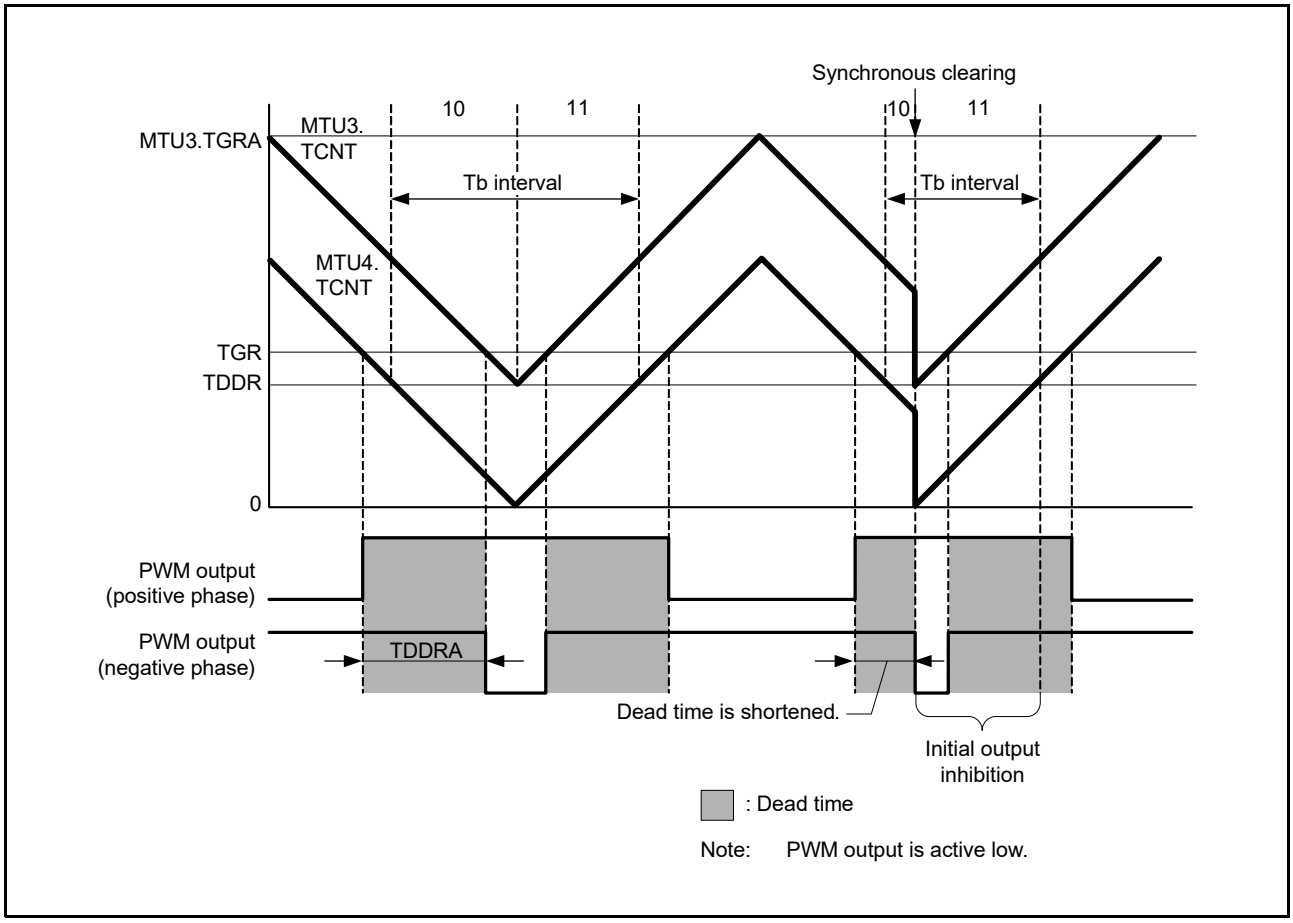


Figure 10.149 Example of Synchronous Clearing (When Condition 1 Applies)

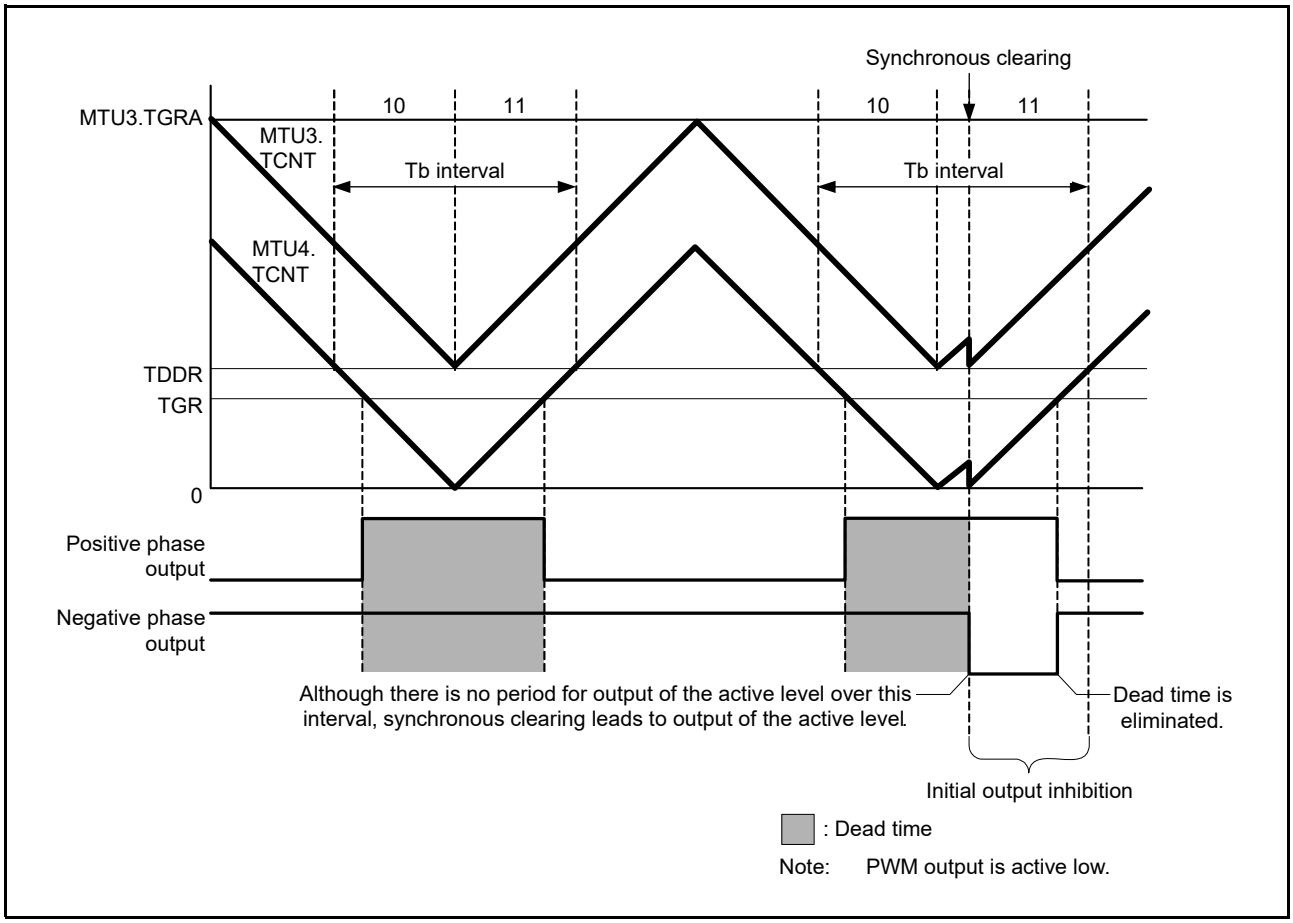


Figure 10.150 Example of Synchronous Clearing (When Condition 2 Applies)

10.6.26 Continuous Output of Interrupt Signal in Response to a Compare Match

When the TGR register is set to 0000h, the P1φ/1 clock is set as the count clock, and compare match is set as the trigger for clearing of the count clock, the value of the TCNT counter remains 0000h, and the interrupt signal will be output continuously (i.e. its level will be flat) rather than output over a single cycle. Consequently, interrupts will not be detected in response to second and subsequent compare matches.

Figure 10.151 shows the timing for continuous output of the interrupt signal in response to a compare match.

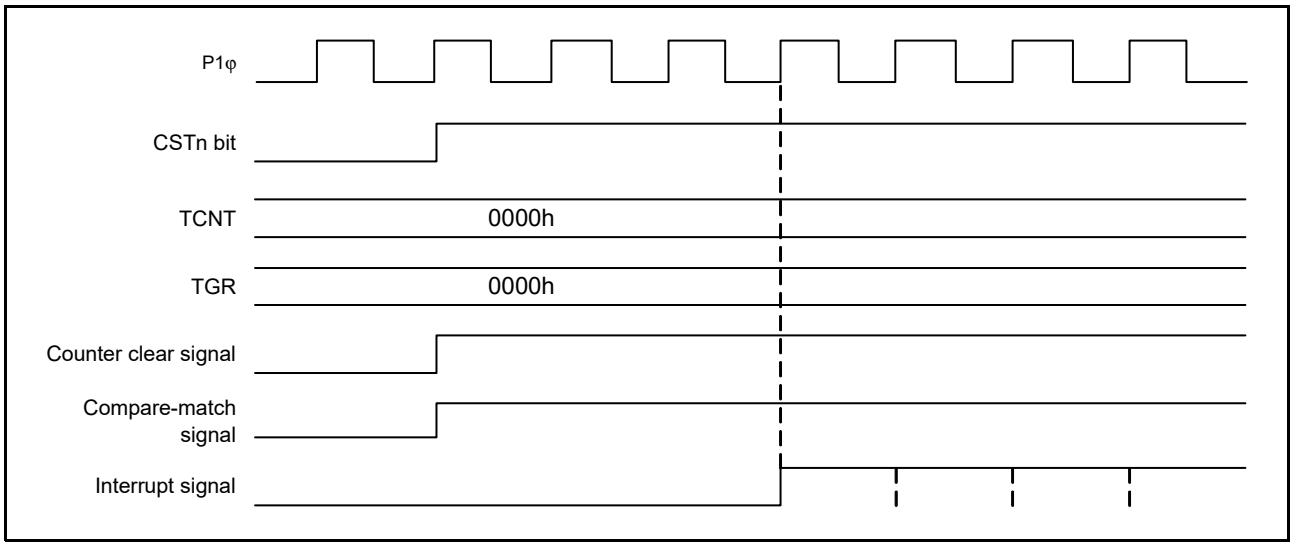


Figure 10.151 Continuous Output of Interrupt Signal in Response to a Compare Match

10.6.27 Notes on A/D Converter Start Request Delaying Function in Complementary PWM Mode

- When the TADCOBRA or TADCOBRB register in MTU4 (MTU7) is set to 0, the UT4AE or UT4BE (UT7AE or UT7BE) bit of TADCR in MTU4 (MTU7) is set to 1, and buffer transfer is performed at the trough of the value of the TCNT counter in MTU4 (MTU7), an A/D converter start request is not issued during up-counting immediately after the transfer (Figure 10.152).
- When the TADCOBRA or TADCOBRB register in MTU4 (MTU7) is set to the same value as that in the TCDR register, the DT4AE or DT4BE (DT7AE or DT7BE) bit of TADCR in MTU4 (MTU7) is set to 1, and buffer transfer is performed at the crest of the value of the TCNT counter in MTU4 (MTU7), an A/D converter start request is not issued during down-counting immediately after the transfer (Figure 10.153).
- When using A/D converter start requests in coordination with the interrupt skipping function, set up the TADCORA or TADCORB register in MTU4 (MTU7) so that the setting satisfies the condition "2 ≤ TADCORA or TADCORB in MTUn ≤ TCDR - 2" (n = 4 or 7).

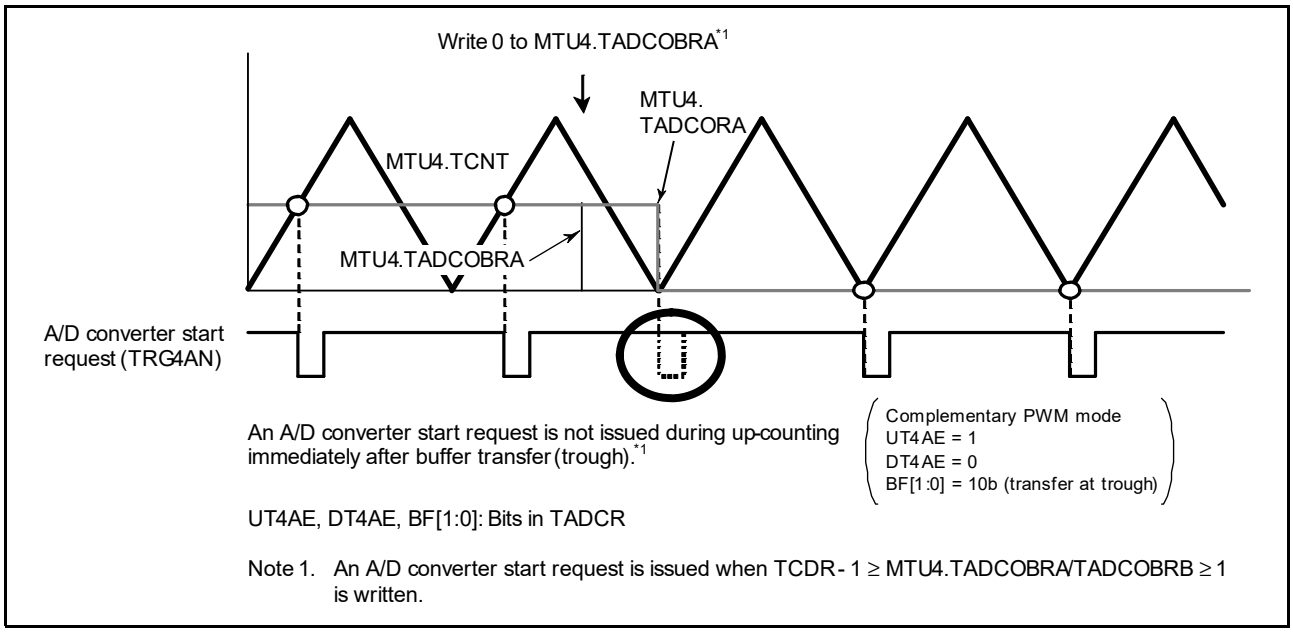


Figure 10.152 A/D Converter Start Request when TADCOBRA is Set to 0 in MTU4

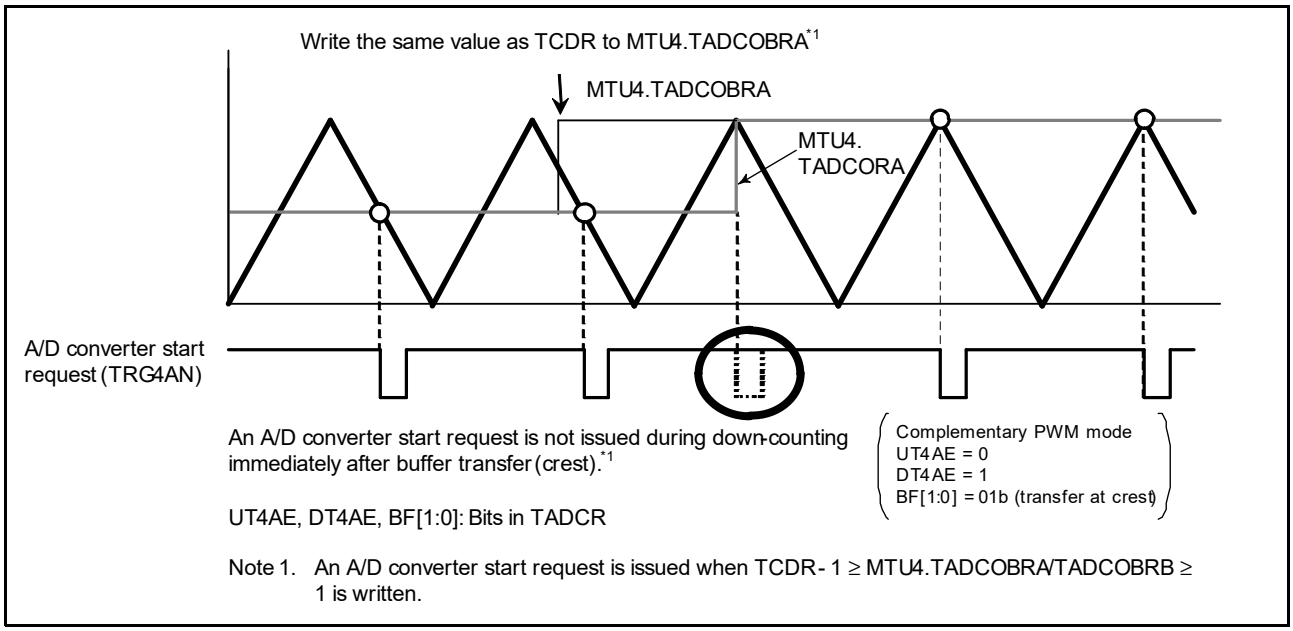


Figure 10.153 A/D Converter Start Request when TADCOBRA is Set to the Same Value as TCDR in MTU4

10.7 MTU Output Pin Initialization

10.7.1 Operating Modes

The MTU has the following six operating modes. Waveforms can be output in any of these modes.

- Normal mode (MTU0 to MTU4 and MTU6 to MTU8)
- PWM mode 1 (MTU0 to MTU4, MTU6, and MTU)
- PWM mode 2 (MTU0 to MTU2)
- Phase counting modes 1 to 5 (MTU1 and MTU2)
- Complementary PWM mode (MTU3, MTU4, MTU6, and MTU7)
- Reset-synchronized PWM mode (MTU3, MTU4, MTU6, and MTU7)

This section describes how to initialize the MTU output pins in each of these modes.

10.7.2 Operation in Case of Re-Setting Due to Error during Operation

If an error occurs during MTU operation, MTU output should be cut off by the system. The output can be cut off by allowing non-active level output from the pins by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports. MTU output can be disabled through TIOR settings. Complementary PWM output (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D, MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7B, MTIOC7C, and MTIOC7D) should be specified through TOERA and TOERB settings. For PWM output pins, output can also be cut by hardware, using port output enable 3 (POE3). The pin initialization procedures for re-setting due to an error during operation and the procedures for restarting in a different mode after re-setting are described below.

The MTU has six operating modes, as stated above. There are thus 36 mode transition combinations, but some transitions are not available with certain channel and mode combinations. Available mode transition combinations are shown in Table 10.80.

Table 10.80 Mode Transition Combinations

	Normal	PWM1	PWM2	PCM	CPWM	RPWM
Normal	(1)	(2)	(3)	(4)	(5)	(6)
PWM1	(7)	(8)	(9)	(10)	(11)	(12)
PWM2	(13)	(14)	(15)	(16)	Not available	Not available
PCM	(17)	(18)	(19)	(20)	Not available	Not available
CPWM	(21)	(22)	Not available	Not available	(23) (24)	(25)
RPWM	(26)	(27)	Not available	Not available	(28)	(29)

Normal: Normal mode
PWM1: PWM mode 1
PWM2: PWM mode 2
PCM: Phase counting modes 1 to 5
CPWM: Complementary PWM mode
RPWM: Reset-synchronized PWM mode

10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation

- When making a transition to a mode (Normal, PWM1, PWM2, or PCM) in which the pin output level is selected by the timer I/O control register (TIOR) setting, initialize the pins by means of TIOR setting.
- In PWM mode 1, waveforms are not output to the MTIOCnB and MTIOCnD pins (n = 3, 4, 6, or 7). If the MTIOCnB or MTIOCnD output is selected as the function of the pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
- In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers. If the MTIOCnm output (n = 0 to 2 and m = A to D) is selected as the function of the pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
- In normal mode or PWM mode 2, if TGRC and TGRD operate as buffer registers, waveforms are not output to the corresponding MTIOCnC and MTIOCnD pins (n = 0, 3, 4, 6, or 7). If the MTIOCnC or MTIOCnD output is selected as the function of the pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
- In PWM mode 1, if either TGRC or TGRD operates as a buffer register, waveforms are not output to the corresponding MTIOCnC and MTIOCnD pins (n = 0, 3, 4, 6, or 7). If the MTIOCnC or MTIOCnD output is selected as the function of the pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
- When making a transition to a mode (CPWM or RPWM) in which the pin output level is selected by the timer output control register (TOCR1A, TOCR2A, TOCR1B, or TOCR2B) setting, temporarily disable output in MTU3 and MTU4 (or MTU6 and MTU7) with the timer output master enable register (TOERA or TOERB). If the MTIOCnm output (n =3, 4, 6, or 7 and m = A to D) is selected as the function of pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports. Switch to normal mode, perform initialization with TIOR, restore TIOR to its initial value, then operate the MTU in accordance with the mode setting procedure (TOCR1A setting, TOCR2A setting, TMDR1 setting, and TOERA setting (TOCR1B setting, TOCR2B setting, TMDR1 setting, and TOERB setting)).

Note: Channel number is substituted for “n” indicated in this section.

Pin initialization procedures are described below for the numbered combinations in Table 10.80. The active level is assumed to be low.

(1) Operation When Error Occurs in Normal Mode and Operation is Restarted in Normal Mode

Figure 10.154 shows a case in which an error occurs in normal mode and operation is restarted in normal mode after re-setting.

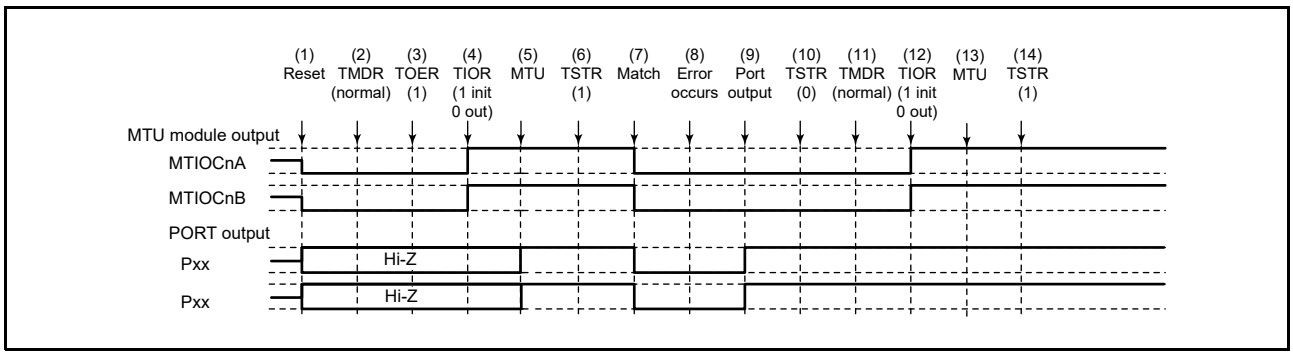


Figure 10.154 Error Occurrence in Normal Mode, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.
(2) After a reset, the TMDR1 setting is for normal mode.
(3) For MTU3 and MTU4 (MTU6 and MTU7), enable output with TOERA (TOERB) before initializing the pins with TIOR.
(4) Initialize the pins with TIOR. (In the example, the initial output is a high level, and a low level is output on compare match occurrence.)
(5) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(6) Start count operation by setting TSTRA (TSTRB).
(7) Output goes low on compare match occurrence.
(8) An error occurs.
(9) Allow non-active level output by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
(10) Stop count operation by setting TSTRA (TSTRB).
(11) This step is not necessary when restarting in normal mode.
(12) Initialize the pins with TIOR.
(13) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(14) Restart operation by setting TSTRA (TSTRB).

(2) Operation When Error Occurs in Normal Mode and Operation is Restarted in PWM Mode 1

Figure 10.155 shows a case in which an error occurs in normal mode and operation is restarted in PWM mode 1 after re-setting.

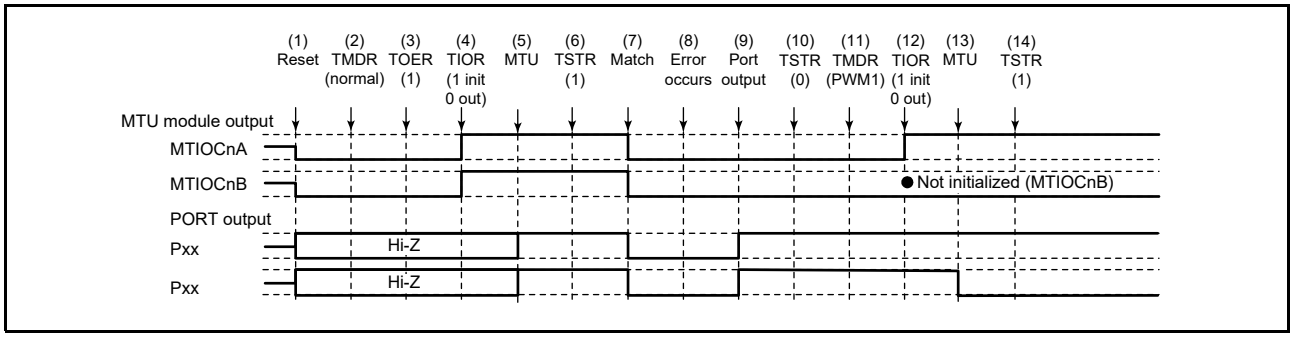


Figure 10.155 Error Occurrence in Normal Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in Figure 10.154.
(11) Set PWM mode 1.
(12) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)
(13) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(14) Restart operation by setting TSTRA (TSTRB).

(3) Operation When Error Occurs in Normal Mode and Operation is Restarted in PWM mode 2

Figure 10.156 shows a case in which an error occurs in normal mode and operation is restarted in PWM mode 2 after re-setting.

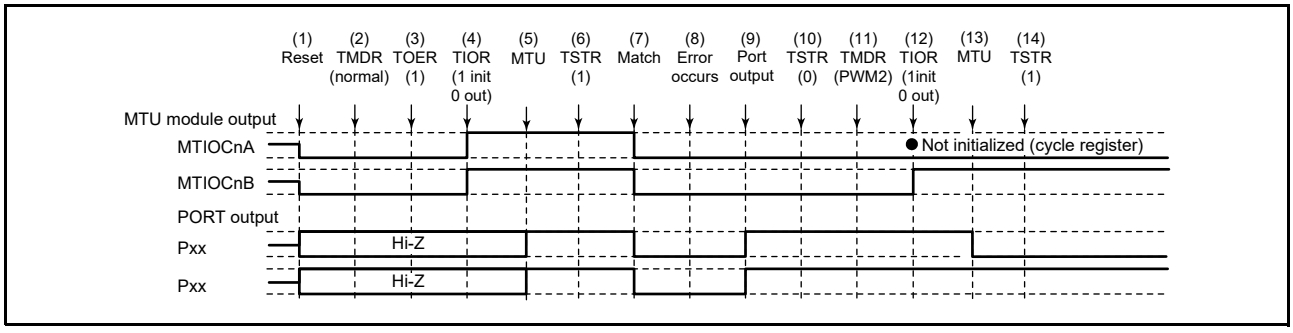


Figure 10.156 Error Occurrence in Normal Mode, Recovery in PWM Mode 2

(1) to (10) are the same as in Figure 10.154.
(11) Set PWM mode 2.
(12) Initialize the pins with TIOR. (In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers. To output a specified level, set up the general output ports with the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)
(13) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(14) Restart operation by setting TSTRA.

Note: PWM mode 2 can only be selected for MTU0 to MTU2, and therefore TOERA setting is not necessary.

(4) Operation When Error Occurs in Normal Mode and Operation is Restarted in Phase Counting Mode

Figure 10.157 shows a case in which an error occurs in normal mode and operation is restarted in phase counting mode after re-setting.

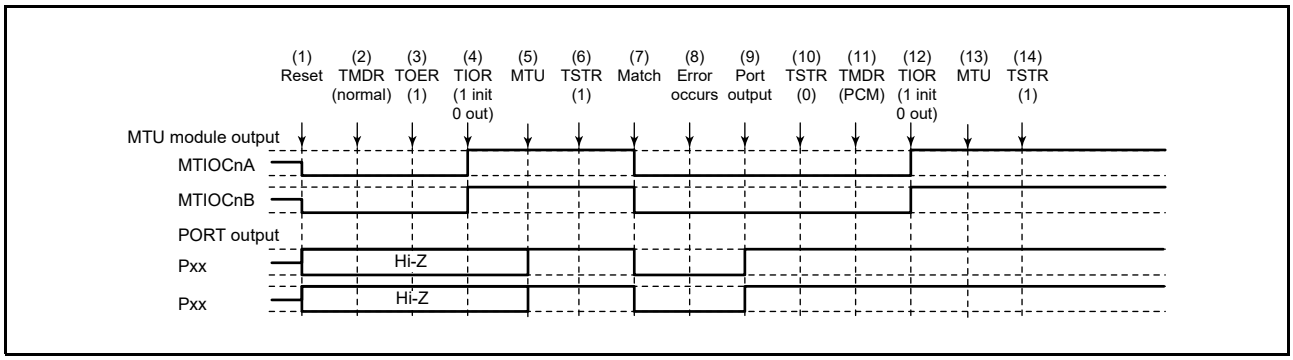


Figure 10.157 Error Occurrence in Normal Mode, Recovery in Phase Counting Mode

(1) to (10) are the same as in Figure 10.154.
(11) Set the phase counting mode.
(12) Initialize the pins with TIOR.
(13) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(14) Restart operation by setting TSTRA.

Note: The phase counting mode can only be selected for MTU1 and MTU2, and therefore TOERA setting is not necessary.

(5) Operation When Error Occurs in Normal Mode and Operation is Restarted in Complementary PWM Mode

Figure 10.158 shows a case in which an error occurs in normal mode and operation is restarted in complementary PWM mode after re-setting.

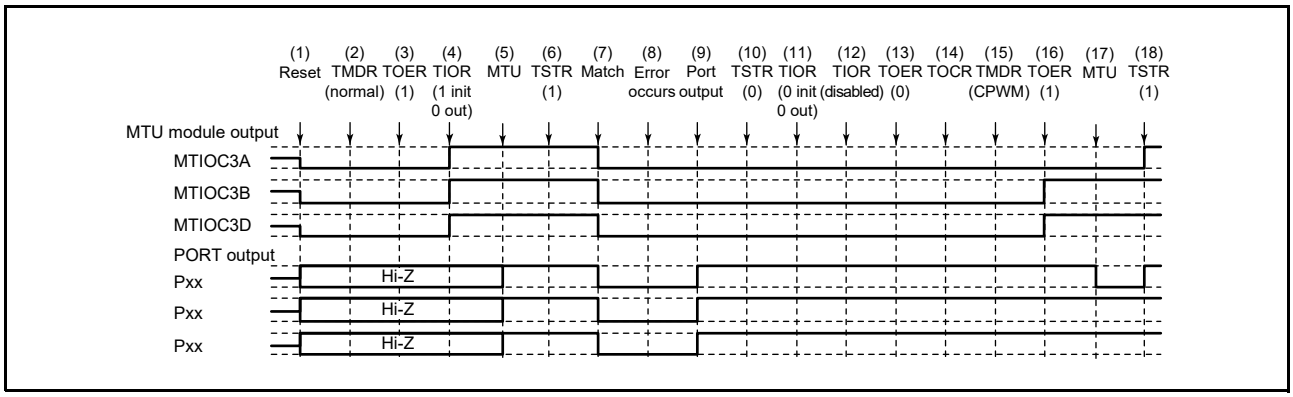


Figure 10.158 Error Occurrence in Normal Mode, Recovery in Complementary PWM Mode

(1) to (10) are the same as in Figure 10.154.
(11) Initialize the normal mode waveform generation section with TIOR.
(12) Disable operation of the normal mode waveform generation section with TIOR.
(13) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(14) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).
(15) Set complementary PWM mode.
(16) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(17) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(18) Restart operation by setting TSTRA (TSTRB).

(6) Operation When Error Occurs in Normal Mode and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 10.159 shows a case in which an error occurs in normal mode and operation is restarted in reset-synchronized PWM mode after re-setting.

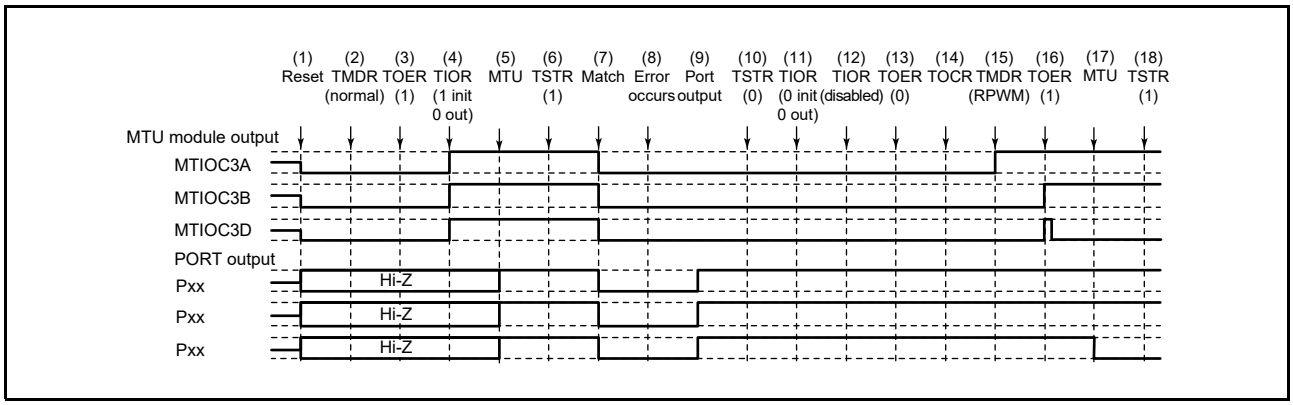


Figure 10.159 Error Occurrence in Normal Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (13) are the same as in Figure 10.156.
(14) Select the reset-synchronized PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).
(15) Set reset-synchronized PWM mode.
(16) Enable output from MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(17) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(18) Restart operation by setting TSTRA (TSTRB).

(7) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Normal Mode

Figure 10.160 shows a case in which an error occurs in PWM mode 1 and operation is restarted in normal mode after re-setting.

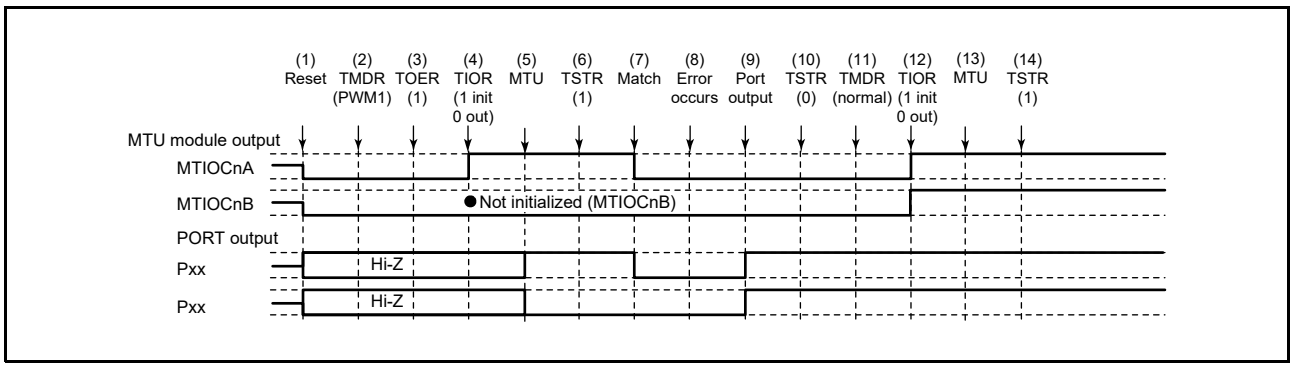


Figure 10.160 Error Occurrence in PWM Mode 1, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.
(2) Set PWM mode 1.
(3) For MTU3 and MTU4 (MTU6 and MTU7), enable output with TOERA (TOERB) before initializing the pins with TIOR.
(4) Initialize the pins with TIOR. (In the example, the initial output is a high level, and a low level is output on compare match occurrence. In PWM mode 1, the MTIOCnB side is not initialized.)
(5) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(6) Start count operation by setting TSTRA (TSTRB).
(7) Output goes low on compare match occurrence.
(8) An error occurs.
(9) Allow non-active level output by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
(10) Stop count operation by setting TSTRA (TSTRB).
(11) Set normal mode.
(12) Initialize the pins with TIOR.
(13) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(14) Restart operation by setting TSTRA (TSTRB).

(8) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in PWM mode 1

Figure 10.161 shows a case in which an error occurs in PWM mode 1 and operation is restarted in PWM mode 1 after re-setting.

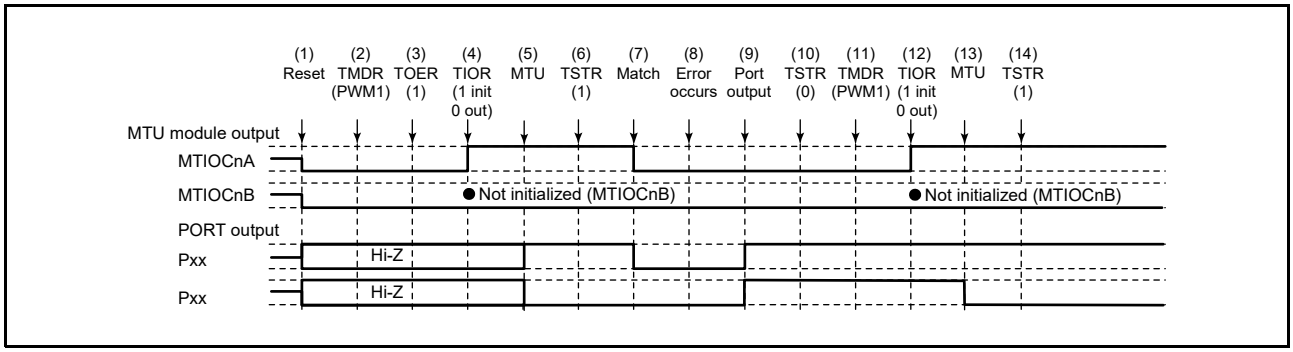


Figure 10.161 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 1

(1) to (10) are the same as in Figure 10.160.
(11) This step is not necessary when restarting in PWM mode 1.
(12) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins.
To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)
(13) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(14) Restart operation by setting TSTRA (TSTRB).

(9) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in PWM mode 2

Figure 10.162 shows a case in which an error occurs in PWM mode 1 and operation is restarted in PWM mode 2 after re-setting.

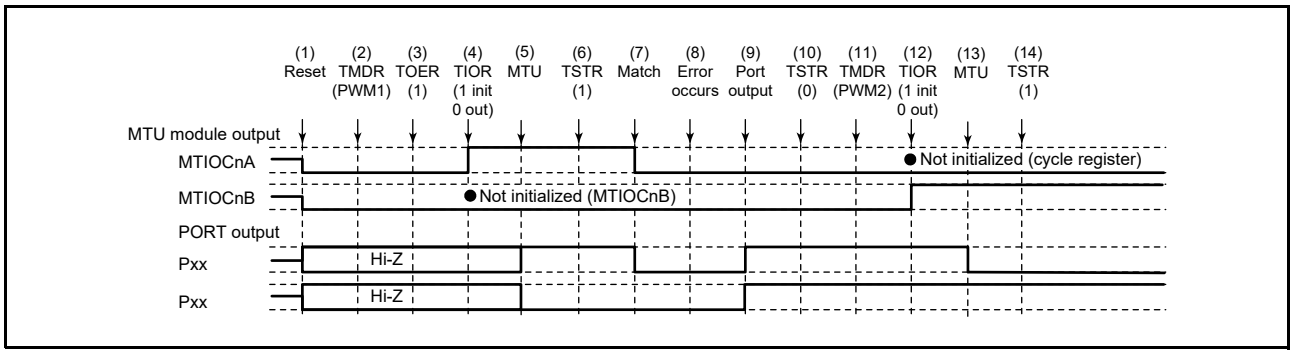


Figure 10.162 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 2

(1) to (10) are the same as in Figure 10.160.
(11) Set PWM mode 2.
(12) Initialize the pins with TIOR. (In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers.
To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)
(13) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(14) Restart operation by setting TSTRA.

Note: PWM mode 2 can only be selected for MTU0 to MTU2, and therefore TOERA setting is not necessary.

(10) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Phase Counting Mode

Figure 10.163 shows a case in which an error occurs in PWM mode 1 and operation is restarted in phase counting mode after re-setting.

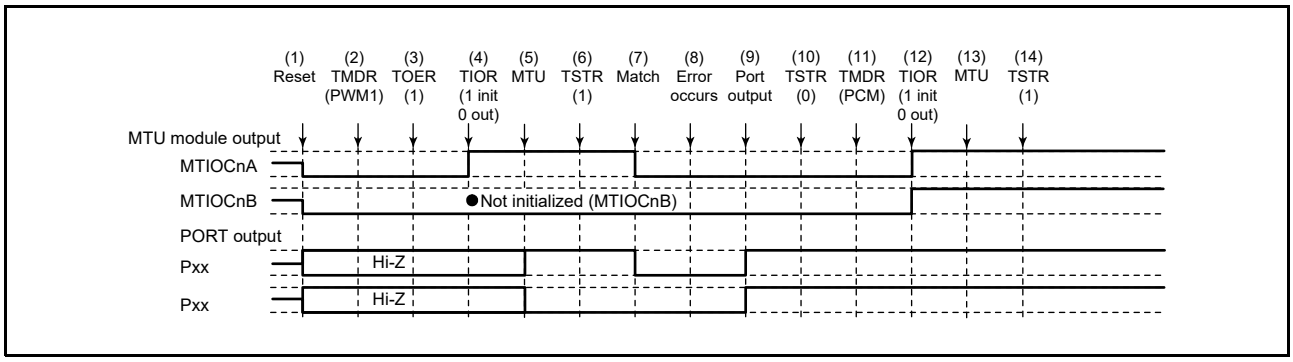


Figure 10.163 Error Occurrence in PWM Mode 1, Recovery in Phase Counting Mode

(1) to (10) are the same as in Figure 10.160.
(11) Set the phase counting mode.
(12) Initialize the pins with TIOR.
(13) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(14) Restart operation by setting TSTRA.

Note: The phase counting mode can only be selected for MTU1 and MTU2, and therefore TOERA setting is not necessary.

(11) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Complementary PWM Mode

Figure 10.164 shows a case in which an error occurs in PWM mode 1 and operation is restarted in complementary PWM mode after re-setting.

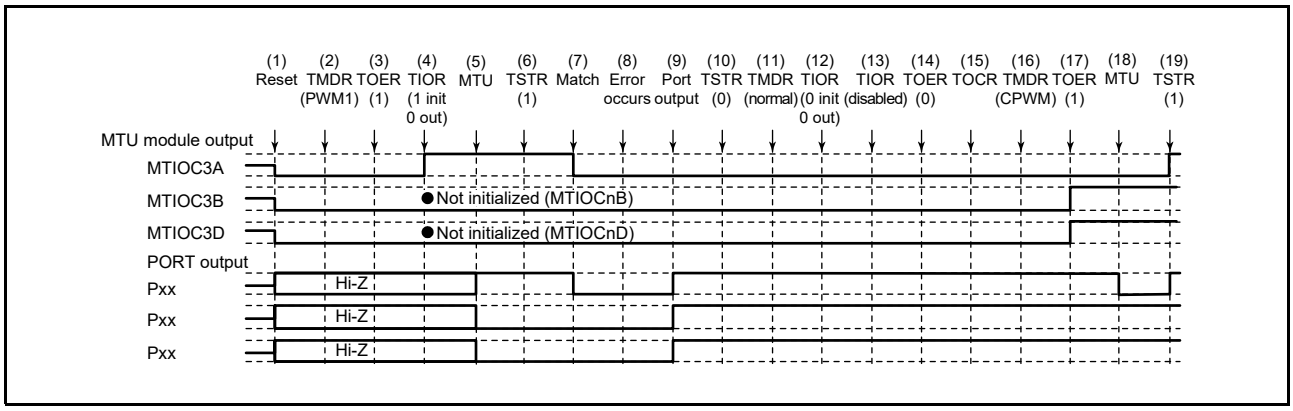


Figure 10.164 Error Occurrence in PWM Mode 1, Recovery in Complementary PWM Mode

(1) to (10) are the same as in Figure 10.160.
(11) Set normal mode to initialize the normal mode waveform generation section.
(12) Initialize the PWM mode 1 waveform generation section with TIOR.
(13) Disable operation of the PWM mode 1 waveform generation section with TIOR
(14) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(15) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).
(16) Set complementary PWM mode.
(17) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(18) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(19) Restart operation by setting TSTRA (TSTRB).

(12) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 10.165 shows a case in which an error occurs in PWM mode 1 and operation is restarted in reset-synchronized PWM mode after re-setting.

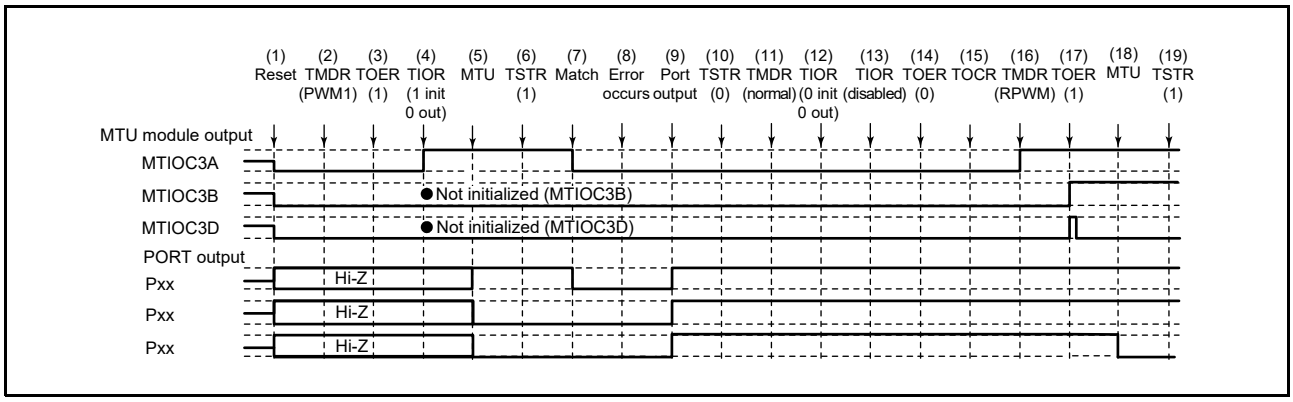


Figure 10.165 Error Occurrence in PWM Mode 1, Recovery in Reset-Synchronized PWM Mode

(1) to (14) are the same as in Figure 10.164.
(15) Select the reset-synchronized PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).
(16) Set reset-synchronized PWM mode.
(17) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(18) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(19) Restart operation by setting TSTRA (TSTRB).

(13) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in Normal Mode

Figure 10.166 shows a case in which an error occurs in PWM mode 2 and operation is restarted in normal mode after re-setting.

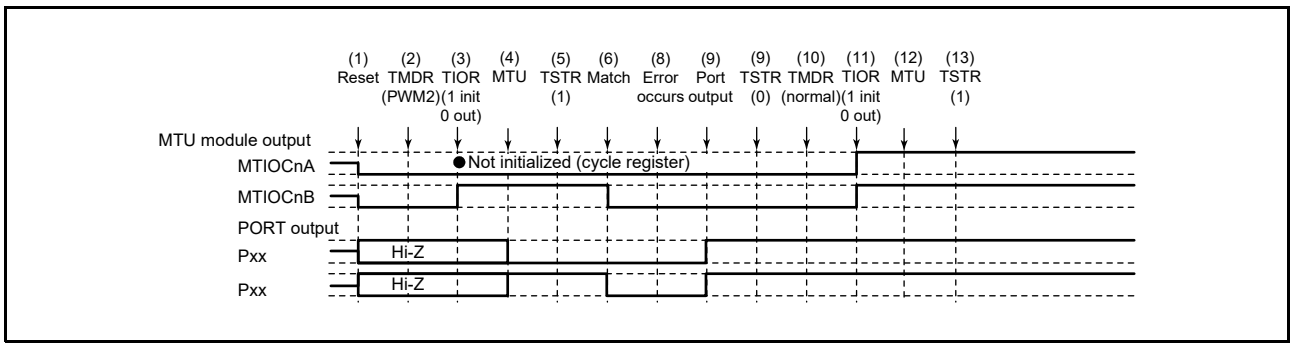


Figure 10.166 Error Occurrence in PWM Mode 2, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.
(2) Set PWM mode 2.
(3) Initialize the pins with TIOR. (In the example, the initial output is a high level, and a low level is output on compare match occurrence. In PWM mode 2, the pins corresponding to the TGR registers used as cycle registers are not initialized. In the example, TGRA in MTUn is used as a cycle register.)
(4) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(5) Start count operation by setting TSTRA.
(6) Output goes low on compare match occurrence.
(7) An error occurs.
(8) Allow non-active level output by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
(9) Stop count operation by setting TSTRA.
(10) Set normal mode.
(11) Initialize the pins with TIOR.
(12) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(13) Restart operation by setting TSTRA.

(14) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in PWM Mode 1

Figure 10.167 shows a case in which an error occurs in PWM mode 2 and operation is restarted in PWM mode 1 after re-setting.

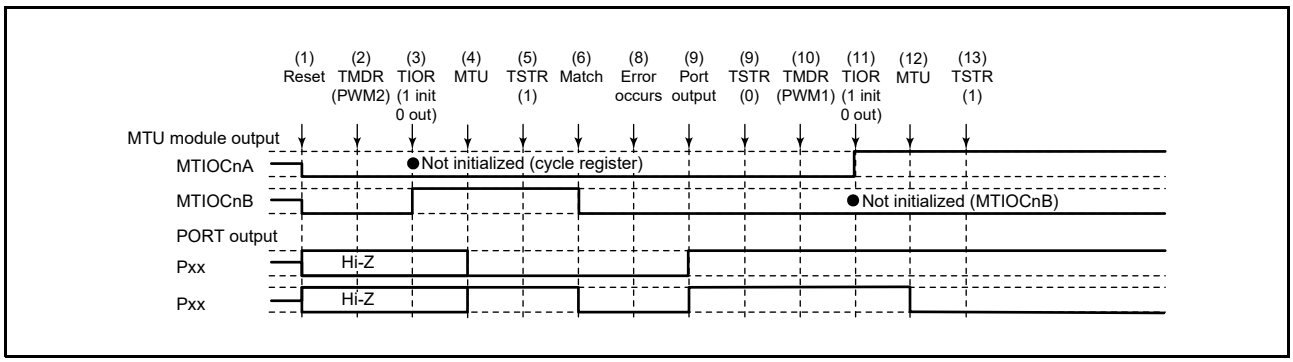


Figure 10.167 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 1

(1) to (9) are the same as in Figure 10.166.
(10) Set PWM mode 1.
(11) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)
(12) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(13) Restart operation by setting TSTRA.

(15) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in PWM Mode 2

Figure 10.168 shows a case in which an error occurs in PWM mode 2 and operation is restarted in PWM mode 2 after re-setting.

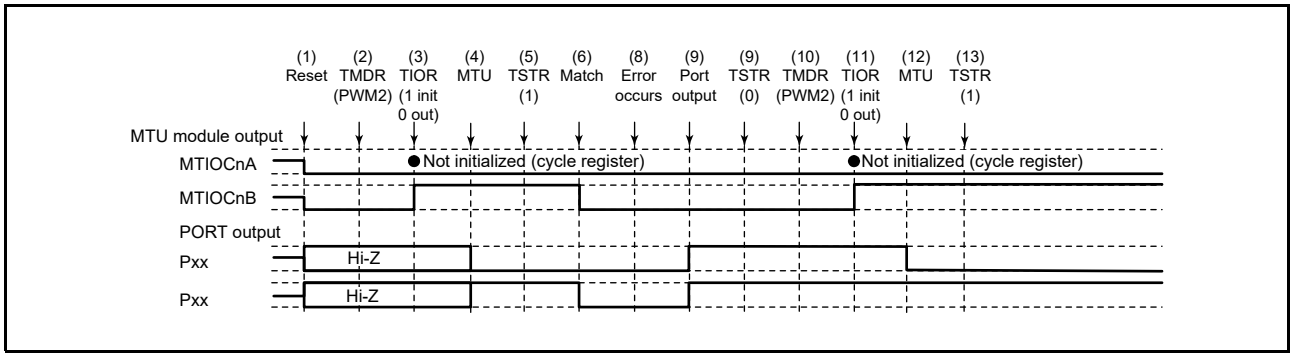


Figure 10.168 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 2

(1) to (9) are the same as in Figure 10.166.
(10) This step is not necessary when restarting in PWM mode 2.
(11) Initialize the pins with TIOR. (In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers.
To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)
(12) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(13) Restart operation by setting TSTRA.

(16) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in Phase Counting Mode

Figure 10.169 shows a case in which an error occurs in PWM mode 2 and operation is restarted in phase counting mode after re-setting.

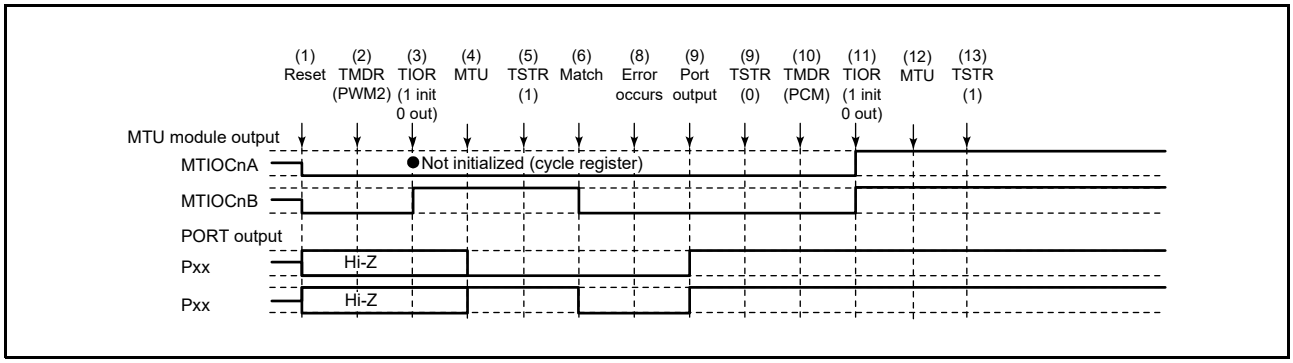


Figure 10.169 Error Occurrence in PWM Mode 2, Recovery in Phase Counting Mode

(1) to (9) are the same as in Figure 10.166.
(10) Set the phase counting mode.
(11) Initialize the pins with TIOR.
(12) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(13) Restart operation by setting TSTRA.

(17) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in Normal Mode

Figure 10.170 shows a case in which an error occurs in phase counting mode and operation is restarted in normal mode after re-setting.

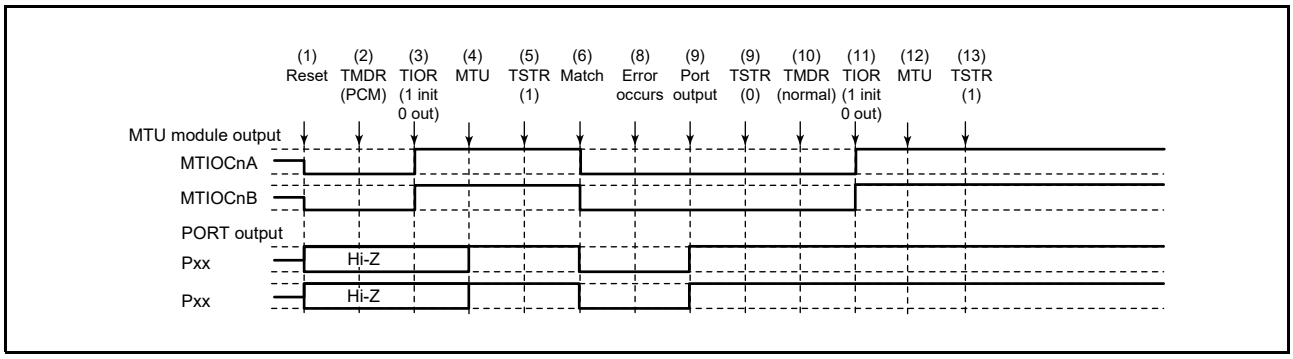


Figure 10.170 Error Occurrence in Phase Counting Mode, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.
(2) Set phase counting mode.
(3) Initialize the pins with TIOR. (In the example, the initial output is a high level, and a low level is output on compare match occurrence.)
(4) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(5) Start count operation by setting TSTRA.
(6) Output goes low on compare match occurrence.
(7) An error occurs.
(8) Allow non-active level output by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
(9) Stop count operation by setting TSTRA.
(10) Set normal mode.
(11) Initialize the pins with TIOR.
(12) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(13) Restart operation by setting TSTRA.

(18) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in PWM Mode 1

Figure 10.171 shows a case in which an error occurs in phase counting mode and operation is restarted in PWM mode 1 after re-setting.

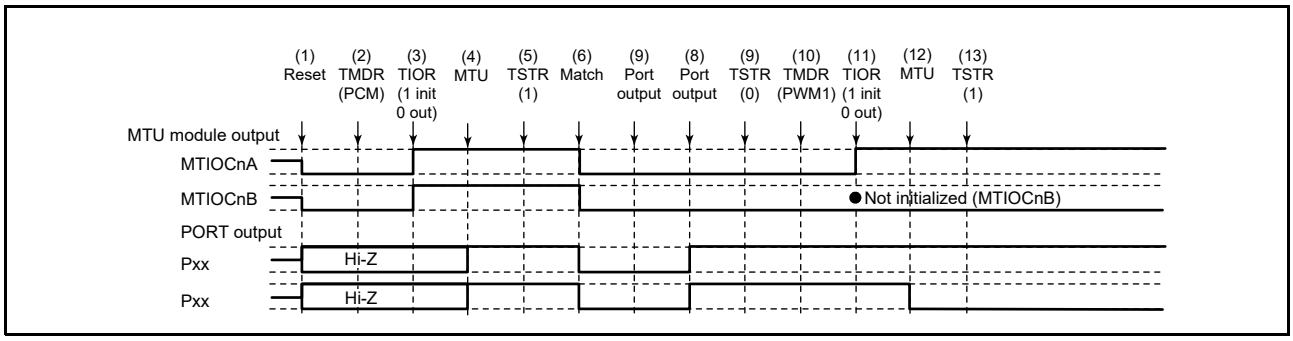


Figure 10.171 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 1

(1) to (9) are the same as in Figure 10.170.
(10) Set PWM mode 1.
(11) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)
(12) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(13) Restart operation by setting TSTRA.

(19) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in PWM Mode 2

Figure 10.172 shows a case in which an error occurs in phase counting mode and operation is restarted in PWM mode 2 after re-setting.

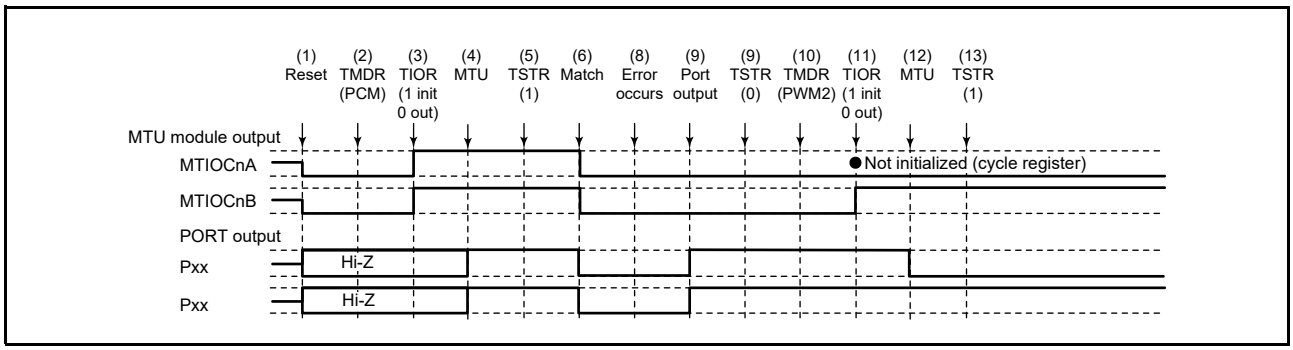


Figure 10.172 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 2

(1) to (9) are the same as in Figure 10.170.
(10) Set PWM mode 2.
(11) Initialize the pins with TIOR. (In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers.
To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)
(12) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(13) Restart operation by setting TSTRA.

(20) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in Phase Counting Mode

Figure 10.173 shows a case in which an error occurs in phase counting mode and operation is restarted in phase counting mode after re-setting.

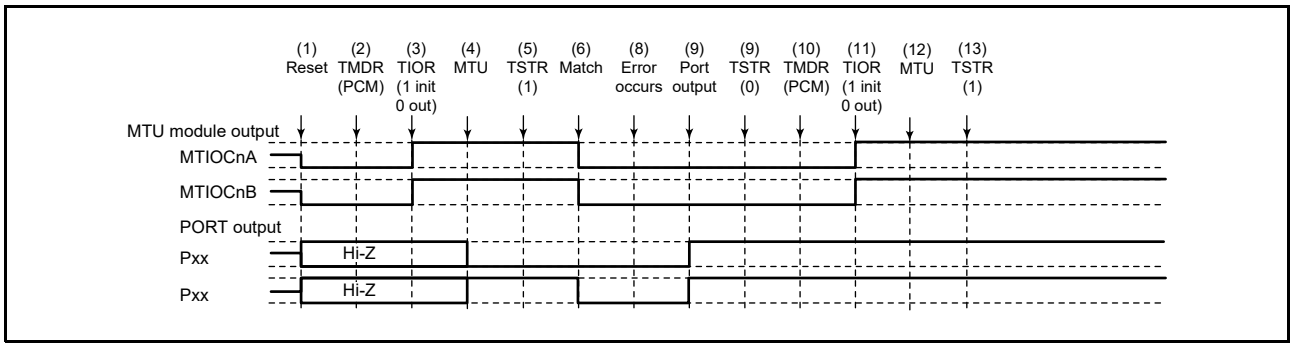


Figure 10.173 Error Occurrence in Phase Counting Mode, Recovery in Phase Counting Mode

(1) to (9) are the same as in Figure 10.170.
(10) This step is not necessary when restarting in phase counting mode.
(11) Initialize the pins with TIOR.
(12) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(13) Restart operation by setting TSTRA.

(21) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Normal Mode

Figure 10.174 shows a case in which an error occurs in complementary PWM mode and operation is restarted in normal mode after re-setting.

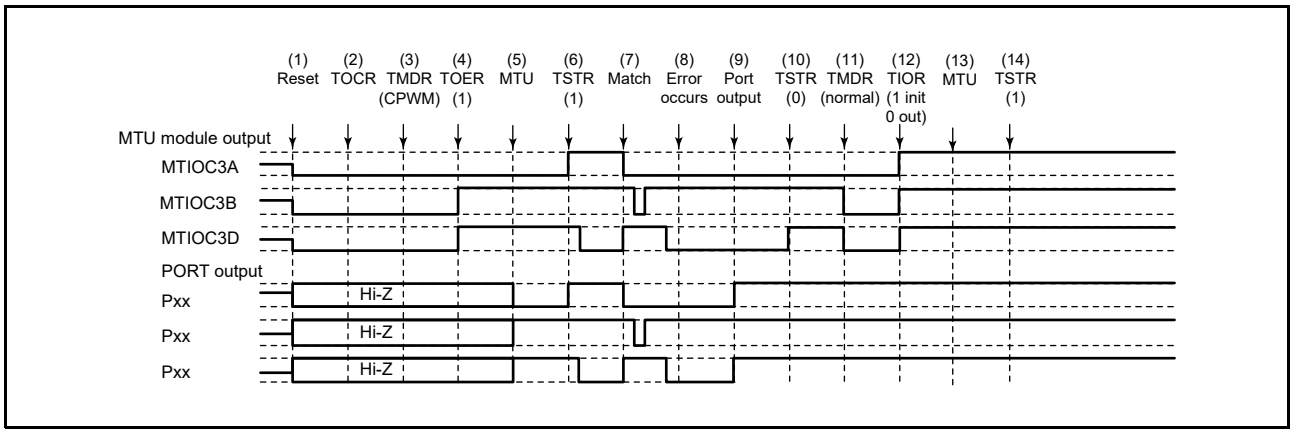


Figure 10.174 Error Occurrence in Complementary PWM Mode, Recovery in Normal Mode

(1) After a reset, the MTU3 output goes low and the ports enter high-impedance state.
(2) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).
(3) Set complementary PWM mode.
(4) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(5) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(6) Start count operation by setting TSTRA (TSTRB).
(7) The complementary PWM waveform is output on compare match occurrence.
(8) An error occurs.
(9) Allow non-active level output by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
(10) Stop count operation by setting TSTRA (TSTRB). (MTU output becomes the initial complementary PWM output value).
(11) Set normal mode (MTU output goes low).
(12) Initialize the pins with TIOR.
(13) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(14) Restart operation by setting TSTRA (TSTRB).

(22) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in PWM Mode 1

Figure 10.175 shows a case in which an error occurs in complementary PWM mode and operation is restarted in PWM mode 1 after re-setting.

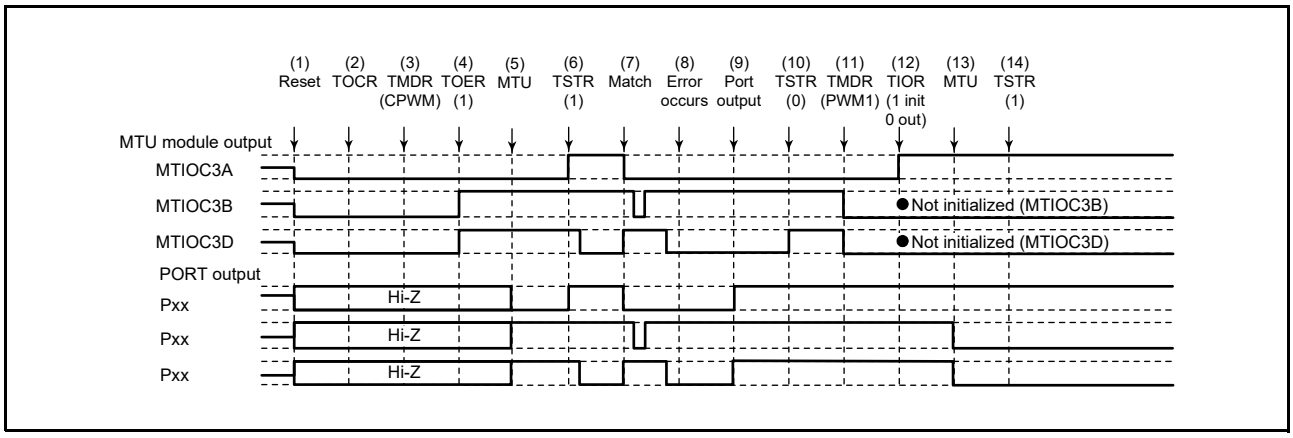


Figure 10.175 Error Occurrence in Complementary PWM Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in Figure 10.174.
(11) Set PWM mode 1 (MTU output goes low).
(12) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)
(13) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(14) Restart operation by setting TSTRA (TSTRB).

(23) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Complementary PWM Mode

Figure 10.176 shows a case in which an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (when operation is restarted using the cycle and duty settings at the time of stopping the counter).

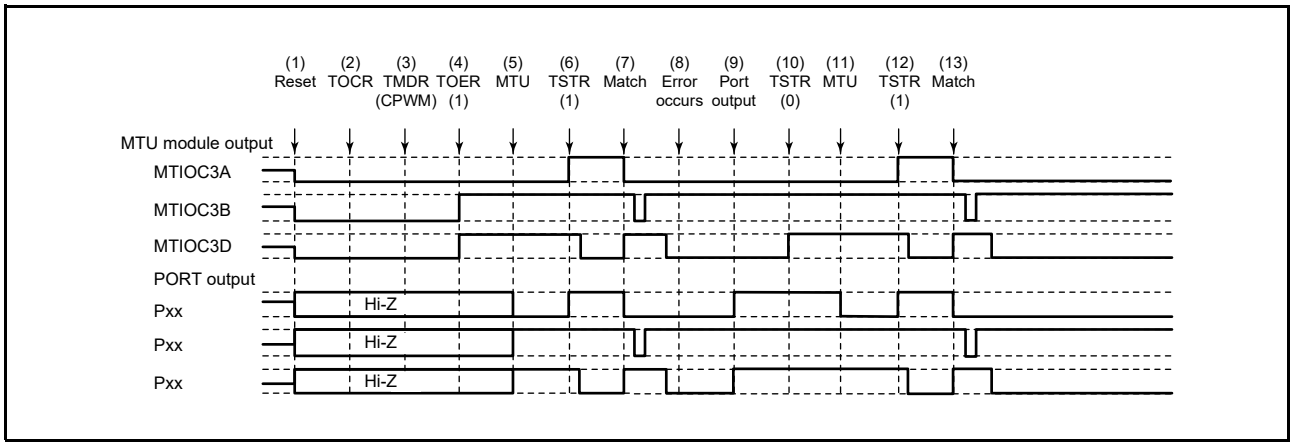


Figure 10.176 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode (1)

(1) to (10) are the same as in Figure 10.174.
(11) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(12) Restart operation by setting TSTRA (TSTRB).
(13) The complementary PWM waveform is output on compare match occurrence.

(24) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Complementary PWM Mode with New Settings

Figure 10.177 shows a case in which an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (operation is restarted using new cycle and duty ratio settings).

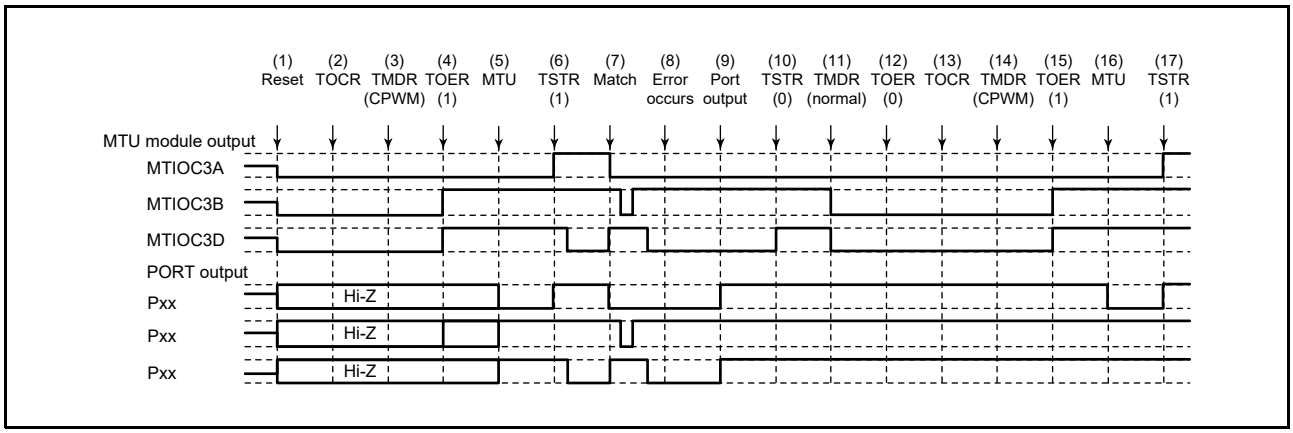


Figure 10.177 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode (2)

(1) to (10) are the same as in Figure 10.174.
(11) Set normal mode and make new settings (MTU output goes low).
(12) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(13) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).
(14) Set complementary PWM mode.
(15) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(16) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(17) Restart operation by setting TSTRA (TSTRB).

(25) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 10.178 shows a case in which an error occurs in complementary PWM mode and operation is restarted in reset-synchronized PWM mode after re-setting.

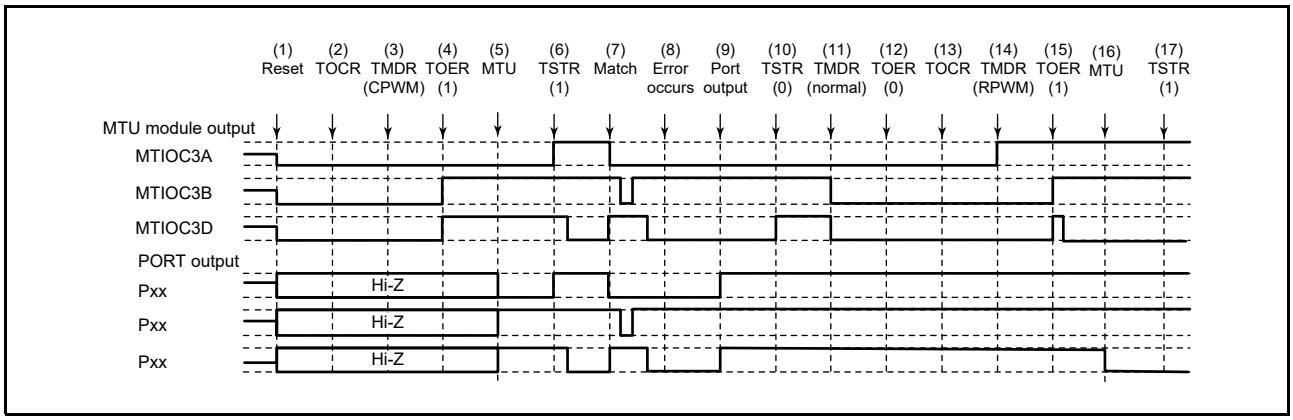


Figure 10.178 Error Occurrence in Complementary PWM Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (10) are the same as in Figure 10.174.
(11) Set normal mode (MTU output goes low).
(12) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(13) Select the reset-synchronized PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).
(14) Set reset-synchronized PWM mode.
(15) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(16) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(17) Restart operation by setting TSTRA (TSTRB).

(26) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Normal Mode

Figure 10.179 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in normal mode after re-setting.

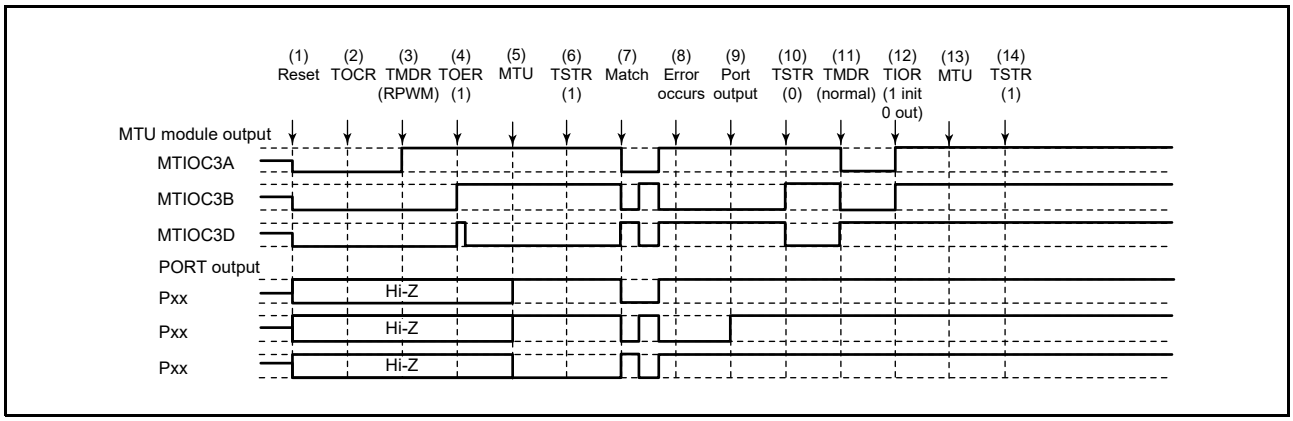


Figure 10.179 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.
(2) Select the reset-synchronized PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).
(3) Set reset-synchronized PWM mode.
(4) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(5) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(6) Start count operation by setting TSTRA (TSTRB).
(7) The reset-synchronized PWM waveform is output on compare match occurrence.
(8) An error occurs.
(9) Allow non-active level output by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
(10) Stop count operation by setting TSTRA (TSTRB). (MTU output becomes the initial reset-synchronized PWM output value.)
(11) Set normal mode (positive-phase MTU output goes low, and negative-phase output goes high).
(12) Initialize the pins with TIOR.
(13) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(14) Restart operation by setting TSTRA (TSTRB).

(27) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in PWM Mode 1

Figure 10.180 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in PWM mode 1 after re-setting.

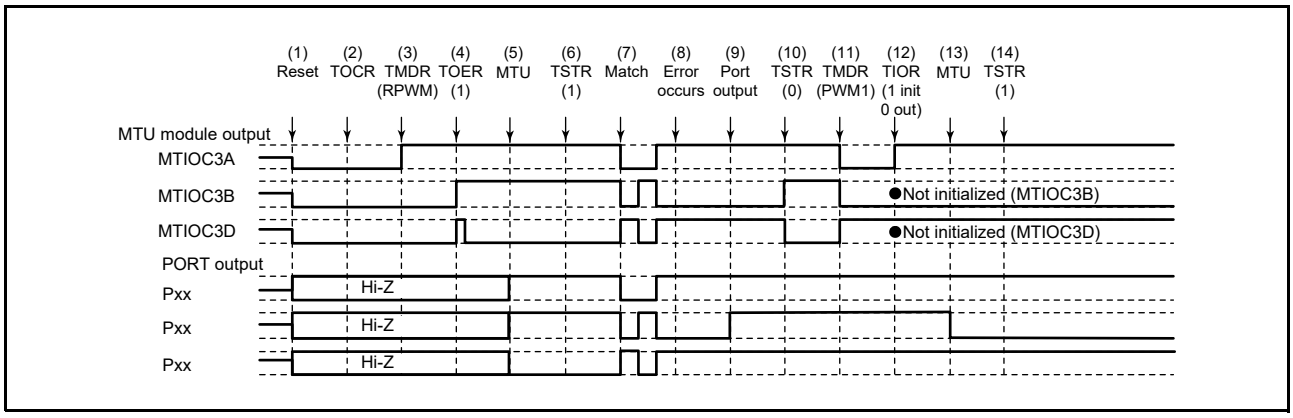


Figure 10.180 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in Figure 10.179.
(11) Set PWM mode 1 (positive-phase MTU output goes low, and negative-phase output goes high).
(12) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)
(13) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(14) Restart operation by setting TSTRA (TSTRB).

(28) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Complementary PWM Mode

Figure 10.181 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in complementary PWM mode after re-setting.

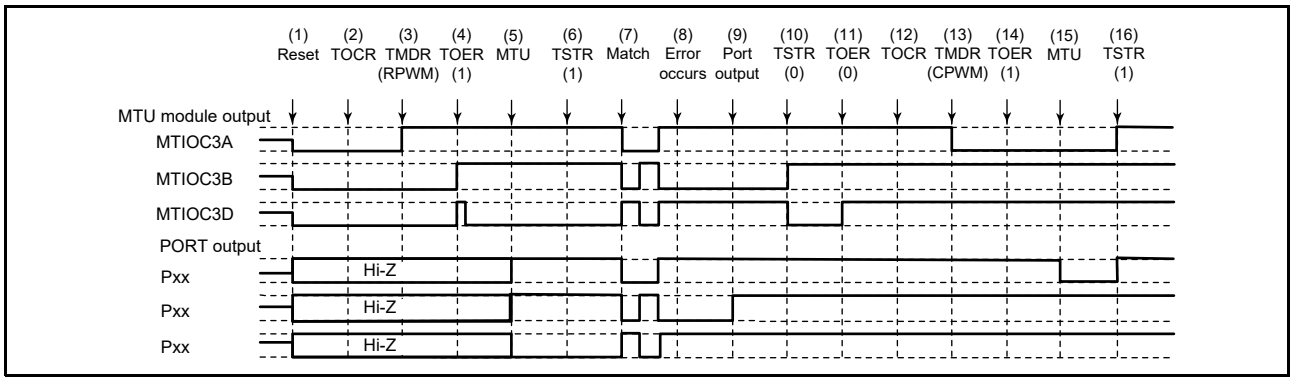


Figure 10.181 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Complementary PWM Mode

(1) to (10) are the same as in Figure 10.179.
(11) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(12) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).
(13) Set complementary PWM mode (MTU cyclic output pin goes low).
(14) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).
(15) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(16) Restart operation by setting TSTRA (TSTRB).

(29) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 10.182 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in reset-synchronized PWM mode after re-setting.

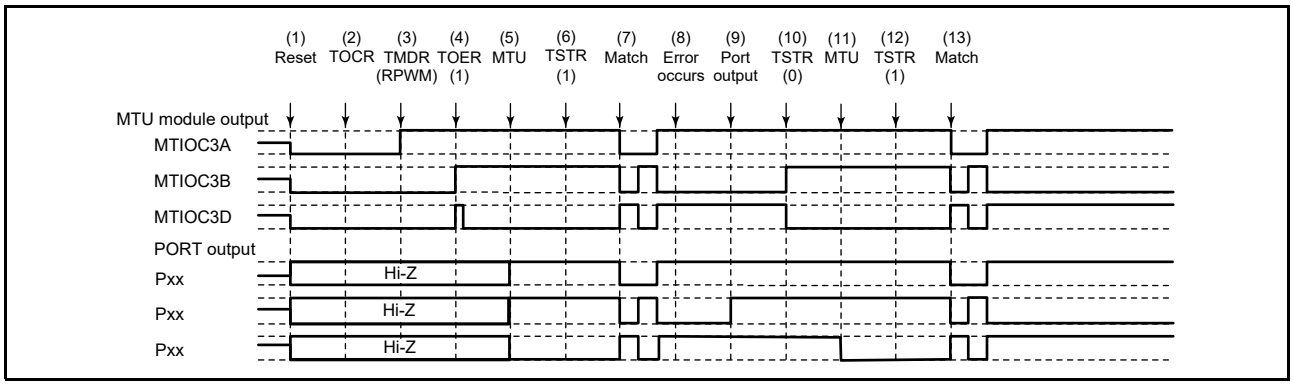


Figure 10.182 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (10) are the same as in Figure 10.179.
(11) Set MTU output using the port mode registers (PMR) and Pmn pin function control register (PmnPFS) corresponding to the GPIO ports.
(12) Restart operation by setting TSTRA (TSTRB).
(13) The reset-synchronized PWM waveform is output on compare match occurrence.

11. Port Output Enable 3 (POE3)

The port output enable 3 (POE3) can be used to place output pins for the MTU3a in the high-impedance state in response to various conditions.

11.1 Overview

Table 11.1 lists the specifications of the POE3, and Figure 11.1 shows a block diagram of the POE3.

Table 11.1 POE3 Specifications

Item	Description
Target pins to be placed in the high-impedance state	• MTU3a output pins MTU0 pin (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D) MTU3 pin (MTIOC3B, MTIOC3D) MTU4 pin (MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D) MTU6 pin (MTIOC6B, MTIOC6D) MTU7 pin (MTIOC7A, MTIOC7B, MTIOC7C, MTIOC7D)
Conditions for the high-impedance state	• Setting pins as inputs: setting the POE0#, POE4#, POE8#, or POE10# pins as inputs (falling edge or low-level sampling). • Short-circuits between output pins: A match (short circuit) between the output signal levels at the active level over one or more cycle on (the following combination of pins) MTU complementary PWM output pins 1 MTIOC3B and MTIOC3D 2 MTIOC4A and MTIOC4C 3 MTIOC4B and MTIOC4D 4 MTIOC6B and MTIOC6D 5 MTIOC7A and MTIOC7C 6 MTIOC7B and MTIOC7D • SPOER register setting being made
Function	• Each of the POE0#, POE4#, POE8#, and POE10# input pins can be set for falling edge, P1φ/4 x 16, P1φ/16 x 16, or P1φ/128 x 16 low-level sampling. • Pins for the MTU complementary PWM output, and MTU0 pin can be placed in high-impedance state by POE0#, POE4#, POE8#, and POE10# pin falling-edge or low-level sampling. • Pins for the MTU complementary PWM output can be placed in high-impedance state when output levels of the MTU complementary PWM output pins are compared and simultaneous active-level output continues for one cycle or more. • Pins for the MTU complementary PWM output, and MTU0 can be placed in the high-impedance state by modifying the settings of the SPOER register of POE3. • Interrupts can be generated by input-level sampling or output level comparison results.

The POE3 has input level detection circuits, pin selection circuits, output level comparison circuits, and a high-impedance request/interrupt request generating circuit as shown in Figure 11.1.

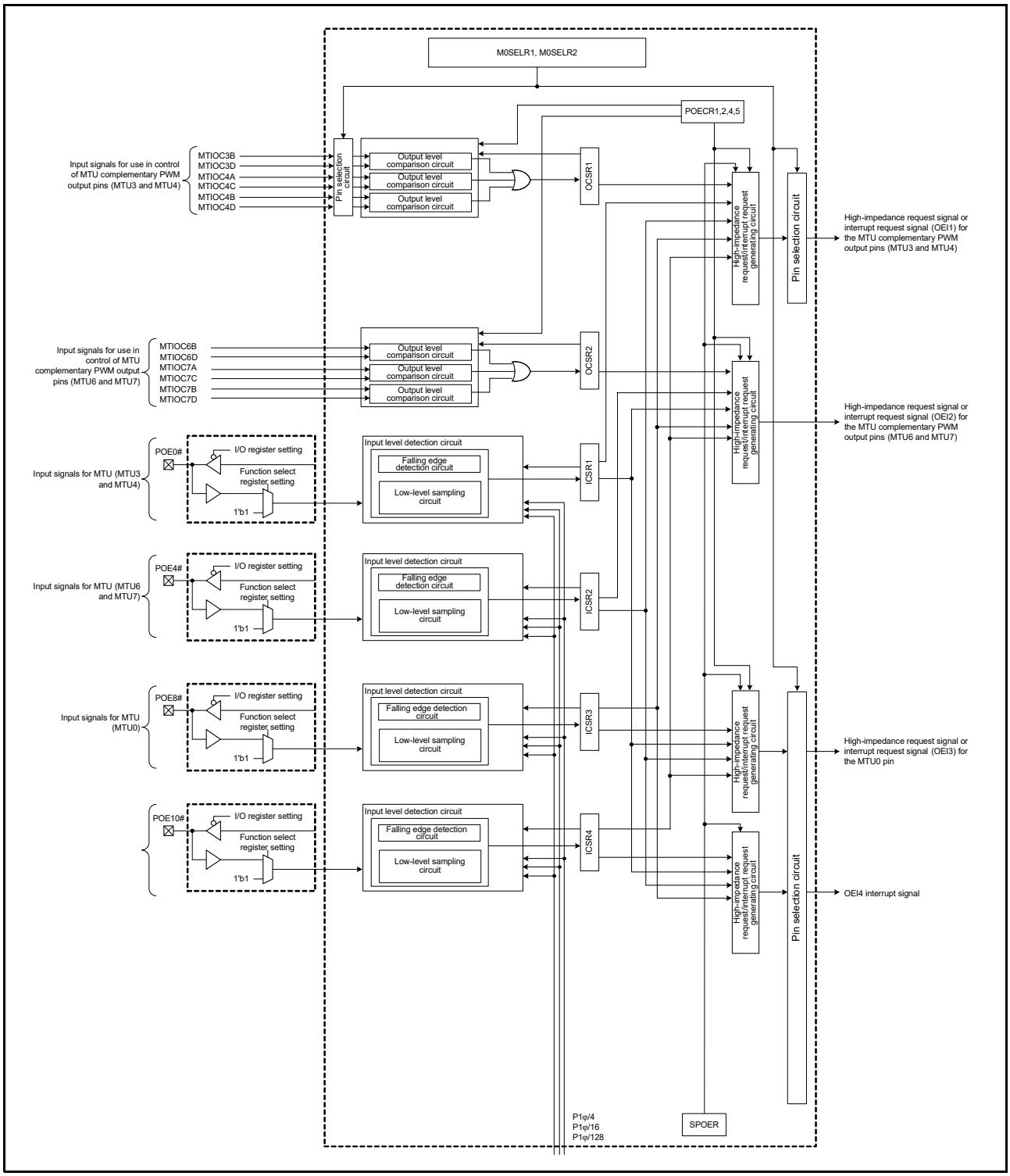


Figure 11.1 POE3 Block Diagram

Table 11.2 shows input/output pins to be used by the POE3.

Table 11.2 POE3 Input/Output Pins

Pin Name	I/O	Description
POE0#	Input	Input pin for the request signal to place the MTU3 and MTU4 pins for MTU complementary PWM output in high-impedance state. In accord with register settings, this pin is also capable of placing the MTU0, MTU6, and MTU7 pins in the high-impedance state.
POE4#	Input	Input pin for the request signal to place the MTU6 and MTU7 pins for MTU complementary PWM output in high-impedance state. In accord with register settings, this pin is also capable of placing the MTU0, MTU3, and MTU4 pins in the high-impedance state.
POE8#	Input	Input pin for the request signal to place the pins for MTU0 in high-impedance state. In accord with register settings, this pin is also capable of placing the MTU3 and MTU4 pins or MTU6 and MTU7 pins for MTU complementary PWM output in high-impedance state.
POE10#	Input	In accord with register settings, this pin is also capable of placing the MTU3 and MTU4 pins or MTU6 and MTU7 pins for MTU complementary PWM output, and MTU0 pins in high-impedance state.

Table 11.3 shows output level comparisons with pin combinations.

Table 11.3 Pin Combinations

Pin Combination	I/O	Description
MTIOC3B and MTIOC3D	Output	The MTU3 and MTU4 pins for MTU complementary PWM output set in the M3SELR, M4SELR1, and M4SELR2 registers are placed in high-impedance state when both pins of a pair simultaneously output the active level*1 for one or more cycles.of the peripheral clock 1C (P1φ). Pin combinations for output comparison and high-impedance control can be selected by registers of POE3. Note 1. The low level is output when the OLSP bit in TOCR1A of MTUn is 0 with the TOCS bit in TOCR1A of MTUn cleared to 0, or the high level is output when the OLSP bit is 1. Otherwise, the low level is output when the OLS3N, OLS3P, OLS2N, OLS2P, OLS1N, and OLS1P bits in TOCR2A of MTUn are 0 with the TOCS bit in TOCR1A of MTUn set to 1, or the high level is output when these bits are 1.
MTIOC4A and MTIOC4C	Output	
MTIOC4B and MTIOC4D	Output	
MTIOC6B and MTIOC6D	Output	The MTU6 and MTU7 pins for MTU complementary PWM output are placed in high-impedance state when both pins of a pair simultaneously output the active level*1 for one or more cycles.of the peripheral clock 1C (P1φ). Pin combinations for output comparison and high-impedance control can be selected by registers of POE3. Note 1. The low level is output when the OLSP bit in TOCR1B of MTUn is 0 with the TOCS bit in TOCR1B of MTUn cleared to 0, or the high level is output when the OLSP bit is 1. Otherwise, the low level is output when the OLS3N, OLS3P, OLS2N, OLS2P, OLS1N, and OLS1P bits in TOCR2B of MTUn are 0 with the TOCS bit in TOCR1B of MTUn set to 1, or the high level is output when these bits are 1.
MTIOC7A and MTIOC7C	Output	
MTIOC7B and MTIOC7D	Output	

11.2 Register Descriptions

Table 11.4 shows the register configuration.
The POE3 registers are initialized by a reset.

Table 11.4 Register configuration

Register Name	Abbreviation	Address	Access size
Input level control/status register 1	ICSR1	H'E804_2000	16
Output level control/status register 1	OCSR1	H'E804_2002	16
Input level control/status register 2	ICSR2	H'E804_2004	16
Output level control/status register 2	OCSR2	H'E804_2006	16
Input level control/status register 3	ICSR3	H'E804_2008	16
Software port output enable register	SPOER	H'E804_200A	8
Port output enable control register 1	POECR1	H'E804_200B	8
Port output enable control register 2	POECR2	H'E804_200C	16
Port output enable control register 4	POECR4	H'E804_2010	16
Port output enable control register 5	POECR5	H'E804_2012	16
Input level control/status register 4	ICSR4	H'E804_2016	16
MTU0 pin select register 1	M0SELR1	H'E804_2024	8
MTU0 pin select register 2	M0SELR2	H'E804_2025	8

11.2.1 Input Level Control/Status Register 1 (ICSR1)

ICSR1 selects the input modes for the POE0# pins, controls the enable/disable of interrupts, and indicates status.

Address(es): H'E804_2000

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	POE0F	—	—	—	PIE1	—	—	—	—	—	—	POE0M[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	POE0M[1:0]	POE0 Mode Select	b1 b0 0 0: Accepts a request on the falling edge of POE0# input. 0 1: Accepts a request when POE0# input has been sampled 16 times at P1φ/4 clock pulses and all are low level. 1 0: Accepts a request when POE0# input has been sampled 16 times at P1φ/16 clock pulses and all are low level. 1 1: Accepts a request when POE0# input has been sampled 16 times at P1φ/128 clock pulses and all are low level.	R/W*1
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	PIE1	Port Interrupt Enable 1	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b11 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b12	POE0F	POE0 Flag	0: Indicates that a high-impedance request has not been input to the POE0# pin. 1: Indicates that a high-impedance request has been input to the POE0# pin.	R/(W)*2
b15 to b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset. After the bits are modified once, writing these bits does not modify their values until the register is reset.
Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

POE0M[1:0] Bits (POE0 Mode Select)

These bits select the input mode of the POE0# pin.

PIE1 Bit (Port Interrupt Enable 1)

This bit enables or disables interrupt requests when any one of the POE0F bit of the ICSR1 is set to 1.

POE0F Flag (POE0 Flag)

This flag indicates that a high-impedance request has been input to the POE0# pin.
[Setting condition]

- When the input set by POE0M[1:0] occurs at the POE0# pin

[Clearing condition]

- By writing 0 to POE0F after reading POE0F = 1

11.2.2 Input Level Control/Status Register 2 (ICSR2)

ICSR2 selects the input mode for the POE4# pin, controls the enable/disable of interrupts, and indicates status.

Address(es): H'E804_2004

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	POE4F	—	—	—	PIE2	—	—	—	—	—	—	POE4M[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	POE4M[1:0]	POE4 Mode Select	b1 b0 0 0: Accepts a request on the falling edge of POE4# input 0 1: Accepts a request when POE4# input has been sampled 16 times at P1φ/4 clock pulses and all are low level. 1 0: Accepts a request when POE4# input has been sampled 16 times at P1φ/16 clock pulses and all are low level. 1 1: Accepts a request when POE4# input has been sampled 16 times at P1φ/128 clock pulses and all are low level.	R/W*1
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	PIE2	Port Interrupt Enable 2	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b11 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b12	POE4F	POE4 Flag	0: Indicates that a high-impedance request has not been input to the POE4# pin. 1: Indicates that a high-impedance request has been input to the POE4# pin.	R/(W)*2
b15 to b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset. After the bits are modified once, writing these bits does not modify their values until the register is reset.
Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

POE4M[1:0] Bits (POE4 Mode Select)

These bits select the input mode of the POE4# pin.

PIE2 Bit (Port Interrupt Enable 2)

This bit enables or disables interrupt requests when the POE4F flag in ICSR2 is set to 1.

POE4F Flag (POE4 Flag)

This flag indicates that a high-impedance request has been input to the POE4# pin.
[Setting condition]

- When the input set by POE4M[1:0] occurs at the POE4# pin

[Clearing condition]

- By writing 0 to POE4F after reading POE4F = 1

11.2.3 Input Level Control/Status Register 3 (ICSR3)

ICSR3 selects the input mode for the POE8# pin, controls the enable/disable of interrupts, and indicates status.

Address(es): H'E804_2008

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	POE8F	—	—	POE8E	PIE3	—	—	—	—	—	—	POE8M[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	POE8M[1:0]	POE8 Mode Select	b1 b0 0 0: Accepts a request on the falling edge of POE8# input 0 1: Accepts a request when POE8# input has been sampled 16 times at P1φ/4 clock pulses and all are low level. 1 0: Accepts a request when POE8# input has been sampled 16 times at P1φ/16 clock pulses and all are low level. 1 1: Accepts a request when POE8# input has been sampled 16 times at P1φ/128 clock pulses and all are low level.	R/W*1
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	PIE3	Port Interrupt Enable 3	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b9	POE8E	POE8 High-Impedance Enable	0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.	R/W*1
b11, b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b12	POE8F	POE8 Flag	0: Indicates that a high-impedance request has not been input to the POE8# pin. 1: Indicates that a high-impedance request has been input to the POE8# pin.	R/(W)*2
b15 to b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset. After the bits are modified once, writing these bits does not modify their values until the register is reset.
Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

POE8M[1:0] Bits (POE8 Mode Select)

These bits select the input mode of the POE8# pin.

PIE3 Bit (Port Interrupt Enable 3)

This bit enables or disables interrupt requests when the POE8F bit in ICSR3 is set to 1.

POE8E Bit (POE8 High-Impedance Enable)

This bit specifies whether to place the corresponding pin in high-impedance state when the POE8F bit is set to 1.

POE8F Flag (POE8 Flag)

This flag indicates that a high-impedance request has been input to the POE8# pin.

[Setting condition]

- When the input set by POE8M[1:0] occurs at the POE8# pin

[Clearing condition]

- By writing 0 to POE8F after reading POE8F = 1

11.2.4 Input Level Control/Status Register 4 (ICSR4)

ICSR4 selects the POE10# pin input mode, controls the enable/disable of interrupts, and indicates status.

Address(es): H'E804_2016

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	POE10F	—	—	POE10E	PIE4	—	—	—	—	—	—	POE10M[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	POE10M[1:0]	POE10 Mode Select	b1 b0 0 0: Accepts a request on the falling edge of POE10# input 0 1: Accepts a request when POE10# input has been sampled 16 times at P1φ/4 clock pulses and all are low level. 1 0: Accepts a request when POE10# input has been sampled 16 times at P1φ/16 clock pulses and all are low level. 1 1: Accepts a request when POE10# input has been sampled 16 times at P1φ/128 clock pulses and all are low level.	R/W*1
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	PIE4	Port Interrupt Enable 4	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b9	POE10E	POE10 High-Impedance Enable	0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.	R/W*1
b11, b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b12	POE10F	POE10 Flag	0: Indicates that a high-impedance request has not been input to the POE10# pin. 1: Indicates that a high-impedance request has been input to the POE10# pin.	R/(W)*2
b15 to b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset. After the bits are modified once, writing these bits does not modify their values until the register is reset.
Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

POE10M[1:0] Bits (POE10 Mode Select)

These bits select the input mode of the POE10# pin.

PIE4 Bit (Port Interrupt Enable 4)

This bit enables or disables interrupt requests when the POE10F bit is set to 1.

POE10E Bit (POE10 High-Impedance Enable)

This bit specifies whether to place the corresponding pin in high-impedance state when the POE10F bit is set to 1.

POE10F Bit (POE10 Flag)

This flag indicates that a request for the high-impedance state has been input to the POE10# pin.
[Setting condition]

- When the input set by POE10M[1:0] occurs at the POE10# pin

[Clearing condition]

- By writing 0 to POE10F after reading POE10F = 1

11.2.5 Output Level Control/Status Register 1 (OCSR1)

OCSR1 controls the enable/disable of output level comparison and interrupts, and indicates status.

Address(es): H'E804_2002

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OSF1	—	—	—	—	—	OCE1	OIE1	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	OIE1	Output Short Interrupt Enable 1	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b9	OCE1	Output Short High-Impedance Enable 1	0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.	R/W*1
b14 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	OSF1	Output Short Flag 1	0: Indicates that outputs have not simultaneously become an active level. 1: Indicates that outputs have simultaneously become an active level.	R/(W)*2

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.
Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

OIE1 Bit (Output Short Interrupt Enable 1)

This bit enables or disables interrupt requests when the OSF1 bit in OCSR1 is set to 1.

OCE1 Bit (Output Short High-Impedance Enable 1)

This bit specifies whether to place the pins in high-impedance state when the OSF1 bit in OCSR1 is set to 1.

OSF1 Flag (Output Short Flag 1)

This flag indicates that any one of the three pairs of two-phase MTU3 and MTU4 pins for MTU complementary PWM output to be compared has simultaneously become an active level.

[Setting condition]

- When any one of the three pairs of two-phase outputs has simultaneously become an active level

[Clearing condition]

- By writing 0 to OSF1 after reading OSF1 = 1

11.2.6 Output Level Control/Status Register 2 (OCSR2)

OCSR2 controls the enable/disable of output level comparison and interrupts, and indicates status.

Address(es): H'E804_2006

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OSF2	—	—	—	—	—	OCE2	OIE2	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	OIE2	Output Short Interrupt Enable 2	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b9	OCE2	Output Short High-Impedance Enable 2	0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.	R/W*1
b14 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	OSF2	Output Short Flag 2	0: Indicates that outputs have not simultaneously become an active level. 1: Indicates that outputs have simultaneously become an active level.	R/(W)*2

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.
Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

OIE2 Bit (Output Short Interrupt Enable 2)

This bit enables or disables interrupt requests when the OSF2 bit in OCSR2 is set to 1.

OCE2 Bit (Output Short High-Impedance Enable 2)

This bit specifies whether to place the pins in high-impedance state when the OSF2 bit in OCSR2 is set to 1.

OSF2 Flag (Output Short Flag 2)

This flag indicates that any one of the three pairs of two-phase MTU6 and MTU7 pins for MTU complementary PWM output to be compared has simultaneously become an active level.

[Setting condition]

- When any one of the three pairs of two-phase outputs has simultaneously become an active level

[Clearing condition]

- By writing 0 to OSF2 after reading OSF2 = 1

11.2.7 Software Port Output Enable Register (SPOER)

SPOER controls high-impedance state of the pins.

Address(es): H'E804_200A

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	MTUCH0HIZ	MTUCH67HIZ	MTUCH34HIZ
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MTUCH34HIZ	MTU3 or MTU4 Output High-Impedance Enable	0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.	R/W
b1	MTUCH67HIZ	MTU6 and MTU7 Output High-Impedance Enable	0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.	R/W
b2	MTUCH0HIZ	MTU0 Output High-Impedance Enable	0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

MTUCH34HIZ Bit (MTU3 or MTU4 Output High-Impedance Enable)

This bit specifies whether to place the MTU complementary PWM output pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D) in high-impedance state.

[Setting condition]

- By writing 1 to MTUCH34HIZ

[Clearing conditions]

- Reset
- By writing 0 to MTUCH34HIZ after reading MTUCH34HIZ = 1*1

Note 1. To write 0 to this bit, be sure to read 1 and then write 0.

MTUCH67HIZ Bit (MTU6 and MTU7 Output High-Impedance Enable)

This bit specifies whether to place the MTU complementary PWM output pins (MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7B, MTIOC7C, and MTIOC7D) in high-impedance state.

[Setting condition]

- By writing 1 to MTUCH67HIZ

[Clearing conditions]

- Reset
- By writing 0 to MTUCH67HIZ after reading MTUCH67HIZ = 1*1

Note 1. To write 0 to this bit, be sure to read 1 and then write 0.

MTUCH0HIZ Bit (MTU0 Output High-Impedance Enable)

This bit specifies whether to place the MTU0 pins in high-impedance state.

[Setting condition]

- By writing 1 to MTUCH0HIZ

[Clearing conditions]

- Reset
- By writing 0 to MTUCH0HIZ after reading MTUCH0HIZ = 1*1

Note 1. To write 0 to this bit, be sure to read 1 and then write 0.

11.2.8 Port Output Enable Control Register 1 (POECR1)

POECR1 controls high-impedance state of the MTU0 pins.

Address(es): H'E804_200B

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	MTU0DZE	MTU0CZE	MTU0BZE	MTU0AZE
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MTU0AZE	MTIOC0A High-Impedance Enable	0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.	R/W*1
b1	MTU0BZE	MTIOC0B High-Impedance Enable	0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.	R/W*1
b2	MTU0CZE	MTIOC0C High-Impedance Enable	0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.	R/W*1
b3	MTU0DZE	MTIOC0D High-Impedance Enable	0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.	R/W*1
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

MTU0AZE Bit (MTIOC0A High-Impedance Enable)

This bit specifies whether to place the MTIOC0A output for the MTU0 pin in high-impedance state when any of the POE8F flag in ICSR3, MTUCH0HIZ bit in SPOER or, as additionally specified in POECR5, the POEmF (n = 1, 2, 4; m = 0, 4, 10) flag in ICSRn, is set to 1.

MTU0BZE Bit (MTIOC0B High-Impedance Enable)

This bit specifies whether to place the MTIOC0B output for the MTU0 pin in high-impedance state when any of the POE8F flag in ICSR3, MTUCH0HIZ bit in SPOER or, as additionally specified in POECR5, the POEmF (n = 1, 2, 4; m = 0, 4, 10) flag in ICSRn, is set to 1.

MTU0CZE Bit (MTIOC0C High-Impedance Enable)

This bit specifies whether to place the MTIOC0C output for the MTU0 pin in high-impedance state when any of the POE8F flag in ICSR3, MTUCH0HIZ bit in SPOER or, as additionally specified in POECR5, the POEmF (n = 1, 2, 4; m = 0, 4, 10) flag in ICSRn, is set to 1.

MTU0DZE Bit (MTIOC0D High-Impedance Enable)

This bit specifies whether to place the MTIOC0D output for the MTU0 pin in high-impedance state when any of the POE8F flag in ICSR3, MTUCH0HIZ bit in SPOER or, as additionally specified in POECR5, the POEmF (n = 1, 2, 4; m = 0, 4, 10) flag in ICSRn, is set to 1.

11.2.9 Port Output Enable Control Register 2 (POECR2)

POECR2 controls high-impedance state of the MTU complementary PWM output pins (MTU3, MTU4, MTU6, and MTU7 pins).

Address(es): H'E804_200C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	MTU3BDZE	MTU4ACZE	MTU4BDZE	—	—	—	—	—	MTU6BDZE	MTU7ACZE	MTU7BDZE
Value after reset:	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b0	MTU7BDZE	MTIOC7B/7D High-Impedance Enable*2	0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.	R/W*1
b1	MTU7ACZE	MTIOC7A/7C High-Impedance Enable*2	0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.	R/W*1
b2	MTU6BDZE	MTIOC6B/6D High-Impedance Enable*2	0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.	R/W*1
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	MTU4BDZE	MTIOC4B/4D High-Impedance Enable	0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.	R/W*1
b9	MTU4ACZE	MTIOC4A/4C High-Impedance Enable	0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.	R/W*1
b10	MTU3BDZE	MTIOC3B/3D High-Impedance Enable	0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.	R/W*1
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.
Note 2. Set this bit to 0 when MTU6 or MTU7 is not used.

MTU7BDZE Bit (MTIOC7B/7D High-Impedance Enable)

This bit specifies whether to place the MTIOC7B output and MTIOC7D output for the MTU3 pin in high-impedance state when any one of the OSF2 flag in OCSR2, POE4F flag in ICSR2, MTUCH67HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 1, 3, 4; m = 0, 8, 10) flag in ICSRn, is set to 1.

MTU7ACZE Bit (MTIOC7A/7C High-Impedance Enable)

This bit specifies whether to place the MTIOC7A output and MTIOC7C output for the MTU3 pin in high-impedance state when any one of the OSF2 flag in OCSR2, POE4F flag in ICSR2, MTUCH67HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 1, 3, 4; m = 0, 8, 10) flag in ICSRn, is set to 1.

MTU6BDZE Bit (MTIOC6B/6D High-Impedance Enable)

This bit specifies whether to place the MTIOC6B output and MTIOC6D output for the MTU3 pin in high-impedance state when any one of the OSF2 flag in OCSR2, POE4F flag in ICSR2, MTUCH67HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 1, 3, 4; m = 0, 8, 10) flag in ICSRn, is set to 1.

MTU4BDZE Bit (MTIOC4B/4D High-Impedance Enable)

This bit specifies whether to place the MTIOC4B output and MTIOC4D output for the MTU4 pin in high-impedance state when any one of the OSF1 flag in OCSR1, POE0F flag in ICSR1, MTUCH34HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 2 to 4; m = 4, 8, 10) flag in ICSRn, is set to 1.

MTU4ACZE Bit (MTIOC4A/4C High-Impedance Enable)

This bit specifies whether to place the MTIOC4A output and MTIOC4C output for the MTU4 pin in high-impedance state when any one of the OSF1 flag in OCSR1, POE0F flag in ICSR1, MTUCH34HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 2 to 4; m = 4, 8, 10) flag in ICSRn, is set to 1.

MTU3BDZE Bit (MTIOC3B/3D High-Impedance Enable)

This bit specifies whether to place the MTIOC3B output and MTIOC3D output for the MTU3 pin in high-impedance state when any one of the OSF1 flag in OCSR1, POE0F flag in ICSR1, MTUCH34HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 2 to 4; m = 4, 8, 10) flag in ICSRn, is set to 1.

11.2.10 Port Output Enable Control Register 4 (POECR4)

The POECR4 is used to extend the control conditions of the high-impedance state for the MTU3, MTU4, MTU6, and MTU7 pins for the MTU complementary PWM output.
For details about the targets and conditions of high-impedance control, see Figure 11.2.

Address(es): H'E804_2010

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	IC4AD DMT67	IC3AD DMT67	—	IC1AD DMT67	—	—	—	—	IC4AD DMT34	IC3AD DMT34	IC2AD DMT34	—	—
Value after reset:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b1	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b2	IC2ADDMT34ZE	MTU3 and MTU4 High-Impedance POE4F Add	0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.	R/W*1
b3	IC3ADDMT34ZE	MTU3 and MTU4 High-Impedance POE8F Add	0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.	R/W*1
b4	IC4ADDMT34ZE	MTU3 and MTU4 High-Impedance POE10F Add	0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.	R/W*1
b8 to b5	—	Reserved	These bits are read as 0. The write value should be 0	R/W
b9	IC1ADDMT67ZE	MTU6 and MTU7 High-Impedance POE0F Add	0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.	R/W*1
b10	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b11	IC3ADDMT67ZE	MTU6 and MTU7 High-Impedance POE8F Add	0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.	R/W*1
b12	IC4ADDMT67ZE	MTU6 and MTU7 High-Impedance POE10F Add	0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.	R/W*1
b15 to b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

IC2ADDMT34ZE Bit (MTU3 and MTU4 High-Impedance POE4F Add)

Adds the POE4F flag in ICSR2 (POE4#) to the high-impedance control conditions for the MTU3 and MTU4 pins (MTIOC3B/MTIOC3D/MTIOC4A/MTIOC4C/MTIOC4B/MTIOC4D).

IC3ADDMT34ZE Bit (MTU3 and MTU4 High-Impedance POE8F Add)

Adds the POE8F flag in ICSR3 (POE8#) to the high-impedance control conditions for the MTU3 and MTU4 pins (MTIOC3B/MTIOC3D/MTIOC4A/MTIOC4C/MTIOC4B/MTIOC4D).

IC4ADDMT34ZE Bit (MTU3 and MTU4 High-Impedance POE10F Add)

Adds the POE10F flag in ICSR4 (POE10#) to the high-impedance control conditions for the MTU3 and MTU4 pins (MTIOC3B/MTIOC3D/MTIOC4A/MTIOC4C/MTIOC4B/MTIOC4D).

IC1ADDMT67ZE Bit (MTU6 and MTU7 High-Impedance POE0F Add)

Adds the POE0F flag in ICSR1 (POE0#) to the high-impedance control conditions for the MTU6, and MTU7 pins (MTIOC6B/MTIOC6D/MTIOC7A/MTIOC7C/MTIOC7B/MTIOC7D).

IC3ADDMT67ZE Bit (MTU6 and MTU7 High-Impedance POE8F Add)

Adds the POE8F flag in ICSR3 (POE8#) to the high-impedance control conditions for the MTU6, and MTU7 pins (MTIOC6B/MTIOC6D/MTIOC7A/MTIOC7C/MTIOC7B/MTIOC7D).

IC4ADDMT67ZE Bit (MTU6 and MTU7 High-Impedance POE10F Add)

Adds the POE10F flag in ICSR4 (POE10#) to the high-impedance control conditions for the MTU6, and MTU7 pins (MTIOC6B/MTIOC6D/MTIOC7A/MTIOC7C/MTIOC7B/MTIOC7D).

11.2.11 Port Output Enable Control Register 5 (POECR5)

The POECR5 is used to extend the control conditions of the high-impedance for the MTU0 pin.
For details about the targets and conditions of high-impedance control, see Figure 11.2.

Address(es): H'E804_2012

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	IC4ADDMT0ZE	—	IC2ADDMT0ZE	IC1ADDMT0ZE	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b1	IC1ADDMT0ZE	MTU0 High-Impedance POE0F Add	0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.	R/W*1
b2	IC2ADDMT0ZE	MTU0 High-Impedance POE4F Add	0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.	R/W*1
b3	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b4	IC4ADDMT0ZE	MTU0 High-Impedance POE10F Add	0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.	R/W*1
b15 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

IC1ADDMT0ZE Bit (MTU0 High-Impedance POE0F Add)

Adds the POE0F flag in ICSR1 (POE0#) to the high-impedance control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

IC2ADDMT0ZE Bit (MTU0 High-Impedance POE4F Add)

Adds the POE4F flag in ICSR2 (POE4#) to the high-impedance control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

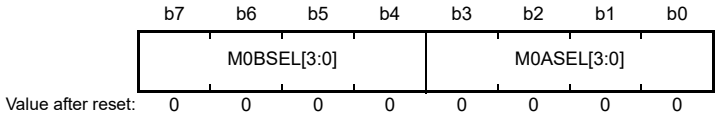
IC4ADDMT0ZE Bit (MTU0 High-Impedance POE10F Add)

Adds the POE10F flag in ICSR4 (POE10#) to the high-impedance control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

11.2.12 MTU0 Pin Select Register 1 (M0SELR1)

M0SELR1 is an 8-bit readable/writable register that selects the A and B pins on MTU0 as targets for high-impedance control.

Address(es): H'E804_2024



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	M0ASEL[3:0]	MTU0-A (MTIOC0A) Pin Select	b3 b0 0000: Controls the high-impedance state of PE3 when it is in use as the MTIOC0A pin. 0010: Controls the high-impedance state of PA6 when it is in use as the MTIOC0A pin. Settings other than above are prohibited.	R/W*1
b7 to b4	M0BSEL[3:0]	MTU0-B (MTIOC0B) Pin Select	b7 b4 0000: Controls the high-impedance state of PE4 when it is in use as the MTIOC0B pin. 0001: Controls the high-impedance state of PA5 when it is in use as the MTIOC0B pin. Settings other than above are prohibited.	R/W*1

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

M0ASEL[3:0] Bits (MTU0-A (MTIOC0A) Pin Select)

These bits select the target MTIOC0A pin for high-impedance control.

M0BSEL[3:0] Bits (MTU0-B (MTIOC0B) Pin Select)

These bits select the target MTIOC0B pin for high-impedance control.

11.2.13 MTU0 Pin Select Register 2 (M0SELR2)

M0SELR2 is an 8-bit readable/writable register that selects the C and D pins on MTU0 as targets for high-impedance control.

Address(es): H'E804_2025

	b7	b6	b5	b4	b3	b2	b1	b0
	M0DSEL[3:0]				M0CSEL[3:0]			
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	M0CSEL[3:0]	MTU0-C (MTIOC0C) Pin Select	b3 b0 0000: Controls the high-impedance state of PE5 when it is in use as the MTIOC0C pin. 0010: Controls the high-impedance state of PA4 when it is in use as the MTIOC0C pin. Settings other than above are prohibited.	R/W*1
b7 to b4	M0DSEL[3:0]	MTU0-D (MTIOC0D) Pin Select	b7 b4 0000: Controls the high-impedance state of PE6 when it is in use as the MTIOC0D pin. 0010: Controls the high-impedance state of PA3 when it is in use as the MTIOC0D pin. Settings other than above are prohibited.	R/W*1

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

M0CSEL[3:0] Bits (MTU0-C (MTIOC0C) Pin Select)

These bits select the target MTIOC0C pin for high-impedance control.

M0DSEL[3:0] Bits (MTU0-D (MTIOC0D) Pin Select)

These bits select the target MTIOC0D pin for high-impedance control.

11.3 Operation

Table 11.5 shows the target pins for high-impedance control and conditions to place the pins in high-impedance state.

Table 11.5 Target Pins and Conditions for High-Impedance Control (1/2)

Pins	Conditions	Detailed Conditions
MTU3 pins (MTIOC3B and MTIOC3D)	• Operation for detection of the POE0# input level • Operation for comparison of the output levels on the MTIOC3B and MTIOC3D pins • SPOER setting • Additional conditions of the POECR4	MTU3BDZE• ((POE0F) +(OSF1•OCE1) +(MTUCH34HIZ) +(IC2ADDMT34ZE•POE4F) +(IC3ADDMT34ZE•POE8E•POE8F) +(IC4ADDMT34ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU4 pins (MTIOC4A and MTIOC4C)	• Operation for detection of the POE0# input level • Operation for comparison of the output levels on the MTIOC4A and MTIOC4C pins • SPOER setting • Additional conditions of the POECR4	MTU4ACZE• ((POE0F) +(OSF1•OCE1) + (MTUCH34HIZ) +(IC2ADDMT34ZE•POE4F) +(IC3ADDMT34ZE•POE8E•POE8F) +(IC4ADDMT34ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU4 pins (MTIOC4B and MTIOC4D)	• Operation for detection of the POE0# input level • Operation for comparison of the output levels on the MTIOC4B and MTIOC4D pins • SPOER setting • Additional conditions of the POECR4	MTU4BDZE• ((POE0F) +(OSF1•OCE1) +(MTUCH34HIZ) +(IC2ADDMT34ZE•POE4F) +(IC3ADDMT34ZE•POE8E•POE8F) +(IC4ADDMT34ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU6 pins (MTIOC6B and MTIOC6D)	• Operation for detection of the POE4# input level • Operation for comparison of the output levels on the MTIOC6B and MTIOC6D pins • SPOER setting • Additional conditions of the POECR4	MTU6BDZE• ((POE4F) +(OSF2•OCE2) +(MTUCH67HIZ) +(IC1ADDMT67ZE•POE0F) +(IC3ADDMT67ZE•POE8E•POE8F) +(IC4ADDMT67ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU7 pins (MTIOC7A and MTIOC7C)	• Operation for detection of the POE4# input level • Operation for comparison of the output levels on the MTIOC7A and MTIOC7C pins • SPOER setting • Additional conditions of the POECR4	MTU7ACZE• ((POE4F) +(OSF2•OCE2) +(MTUCH67HIZ) +(IC1ADDMT67ZE•POE0F) +(IC3ADDMT67ZE•POE8E•POE8F) +(IC4ADDMT67ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU7 pins (MTIOC7B and MTIOC7D)	• Operation for detection of the POE4# input level • Operation for comparison of the output levels on the MTIOC7B and MTIOC7D pins • SPOER setting • Additional conditions of the POECR4	MTU7BDZE• ((POE4F) +(OSF2•OCE2) +(MTUCH67HIZ) +(IC1ADDMT67ZE•POE0F) +(IC3ADDMT67ZE•POE8E•POE8F) +(IC4ADDMT67ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU0 pin (MTIOC0A)	• Operation for detection of the POE8# input level • SPOER setting • Additional conditions of the POECR5	MTU0AZE• ((POE8F•POE8E) +(MTUCH0HIZ) +(IC1ADDMT0ZE•POE0F) +(IC2ADDMT0ZE•POE4F) +(IC4ADDMT0ZE•ICSR4.POE10E•ICSR4.POE10F)

Table 11.5 Target Pins and Conditions for High-Impedance Control (2/2)

Pins	Conditions	Detailed Conditions
MTU0 pin (MTIOC0B)	• Operation for detection of the POE8# input level • SPOER setting • Additional conditions of the POECR5	MTU0BZE• ((POE8F•POE8E) +(MTUCH0HIZ) +(IC1ADDMT0ZE•POE0F) +(IC2ADDMT0ZE•POE4F) +(IC4ADDMT0ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU0 pin (MTIOC0C)	• Operation for detection of the POE8# input level • SPOER setting • Additional conditions of the POECR5	MTU0CZE• ((POE8F•POE8E) +(MTUCH0HIZ) +(IC1ADDMT0ZE•POE0F) +(IC2ADDMT0ZE•POE4F) +(IC4ADDMT0ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU0 pin (MTIOC0D)	• Operation for detection of the POE8# input level • SPOER setting • Additional conditions of the POECR5	MTU0DZE• ((POE8F•POE8E) +(MTUCH0HIZ) +(IC1ADDMT0ZE•POE0F) +(IC2ADDMT0ZE•POE4F) +(IC4ADDMT0ZE•ICSR4.POE10E•ICSR4.POE10F)

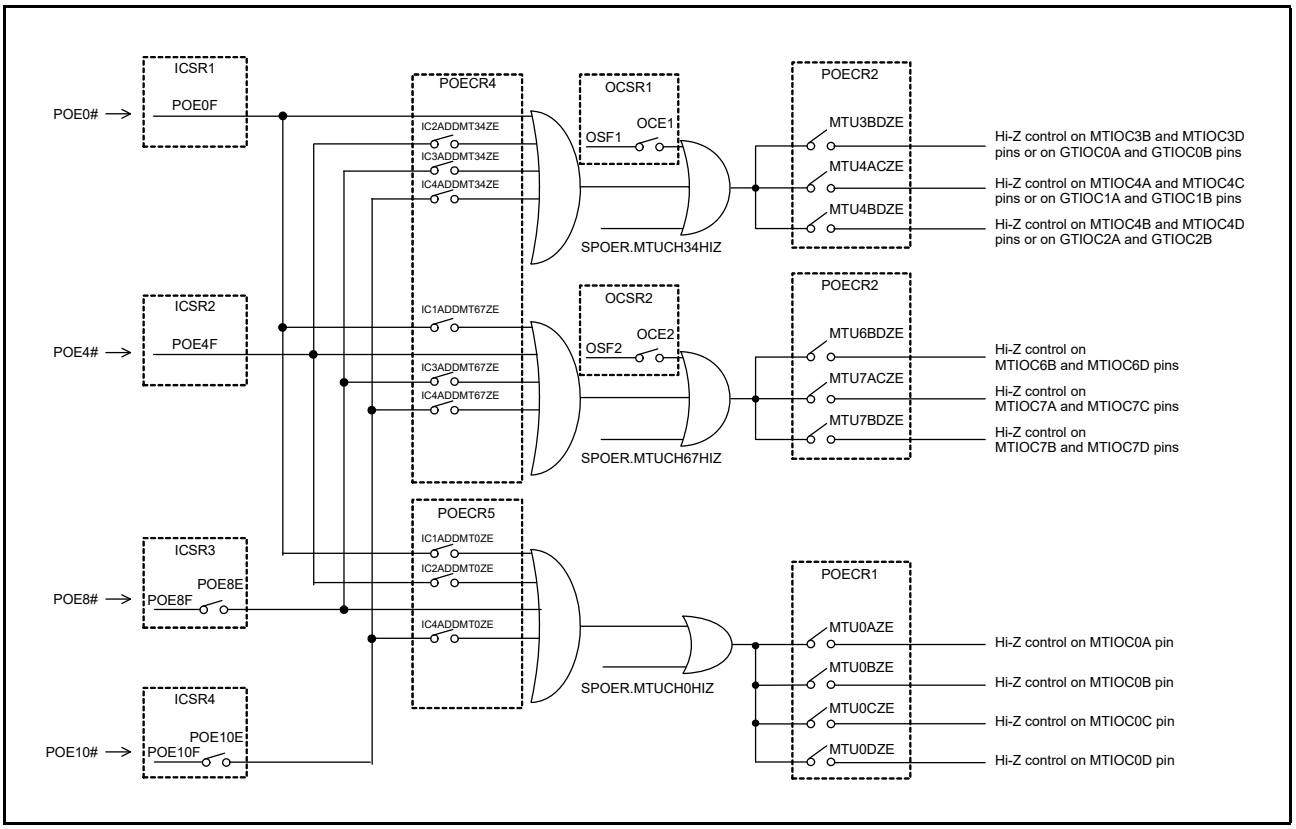


Figure 11.2 Target Pins and Conditions for High-Impedance Control

High-impedance requests to individual pins can be controlled by the settings in the POECR1 to POECR2 registers. The following input pins can be added to high-impedance control conditions by the settings in the POECR4 to POECR5 registers: input pins other than the POE0# pin for the MTU3 and MTU4 pins; input pins other than the POE4# pin for the MTU6 and MTU7 pins; input pins other than the POE8# pin for the MTU0 pins.

For example, setting the IC2ADDMT34ZE bit in POECR4 to 1 outputs high-impedance requests to the MTU3 and

MTU4 pins even when the POE4# input is detected.
High-impedance requests due to the detection of POE8# input or POE10# input can be controlled by the settings in the ICSR3 to ICSR4 registers. (The ICSR1 and ICSR2 registers do not control enabling or disabling of output of high-impedance requests.)
High-impedance requests to the MTU3, MTU4, MTU6, and MTU7 pins as the results of output level comparison can be controlled by the settings in the OCSR1 and OCSR2 registers.

11.3.1 MTU Pin Selection

In this LSI, the pin functions for MTU are respectively assigned to multiple sets of port pins. The target pins for high-impedance control can be selected by the pin selection register in POE3 (M0SELR1 or M0SELR2 register). Table 11.6 shows the correspondence between MTU pins and selection registers.
Note that settings for pins to be used as MTU must be separately made in the registers of the general purpose input/output port (GPIO). Take care so that there are no differences between the pins selected in the POE3 registers and the pins selected in the GPIO registers.

Table 11.6 Correspondence between MTU Pins

MTU Pin Functions	Corresponding Ports	Selection Registers
MTIOC0A	PA6*1	M0SELR1
	PE3	
MTIOC0B	PA5*1	
	PE4	
MTIOC0C	PA4*1	M0SELR2
	PE5	
MTIOC0D	PA3*1	
	PE6	
MTIOC3B	PG2	—
MTIOC3D	PG3	
MTIOC4A	PG4	—
MTIOC4C	PG6	
MTIOC4B	PG5	—
MTIOC4D	PG7	
MTIOC6B	PF5*1	—
	P00*2	
MTIOC6D	PH2*1	
	P02*2	
MTIOC7A	PF0*1	
	P03*2	
MTIOC7C	PF2*1	
	P05*2	
MTIOC7B	PF1*1	
	P04*2	
MTIOC7D	PF3*1	
	P06*2	

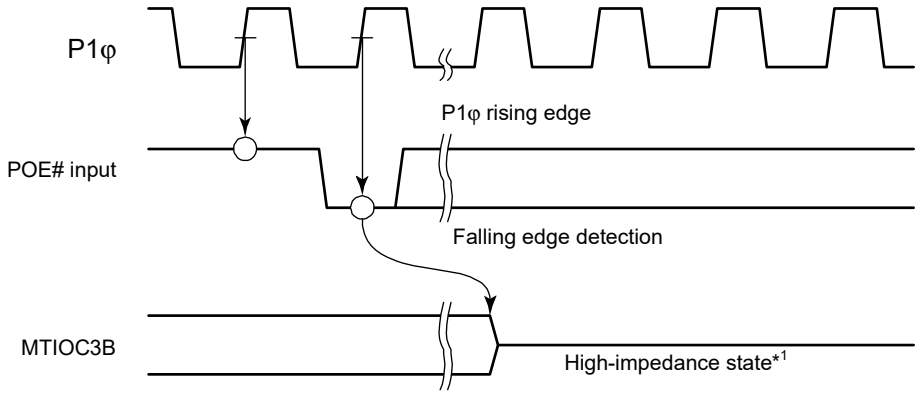
Note 1. This selection is not available in 176-pin products.
Note 2. These are not target pins for high-impedance control by the POE3.

11.3.2 Input Level Detection Operation

If the input conditions set by ICSR1 to ICSR4 occur on the POE0#, POE4#, POE8#, and POE10# pins, the MTU3 and MTU4 or MTU6 and MTU7 pins for the MTU complementary PWM output, and MTU0 pin are placed in high-impedance state. Note however, that these pins are still placed in the high-impedance state even when the MTU functions are not selected for the pins.

(1) Falling Edge Detection

When a change from a high to low level is input to the POE0#, POE4#, POE8#, and POE10# pins, the pins for the MTU complementary PWM output, and pin functions multiplexed with the MTU0 pins are placed in highimpedance state. Figure 11.3 shows a sample timing after the level changes in input to the POE0#, POE4#, POE8#, and POE10# pins until the respective pins enter high-impedance state.

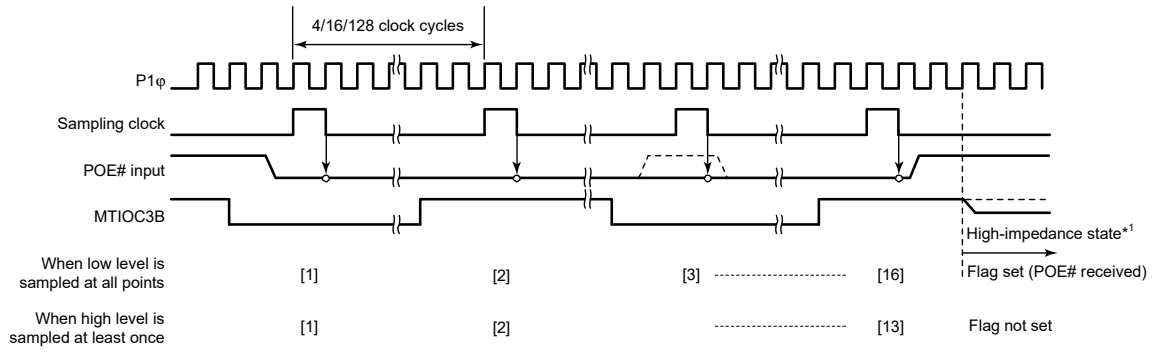


Note 1. Other pins also enter the high-impedance state at the same timing.

Figure 11.3 Falling Edge Detection

(2) Low-Level Detection

Figure 11.4 shows the low-level detection operation. Sixteen continuous low levels are sampled with the sampling clock selected by ICSR1 to ICSR4. If even one high level is detected during this interval, the low level is not accepted. The timing when pins for the MTU complementary PWM output, and MTU0 pins enter the highimpedance state after the sampling clock is input is the same in both falling-edge detection and in low-level detection.



Note 1. Other pins also enter the high-impedance state at the same timing.

Figure 11.4 Low-Level Detection Operation

11.3.3 Output Level Compare Operation

Figure 11.5 shows an example of the output level compare operation for the combination of MTIOC3B and MTIOC3D. The operation is the same for the other pin combinations.

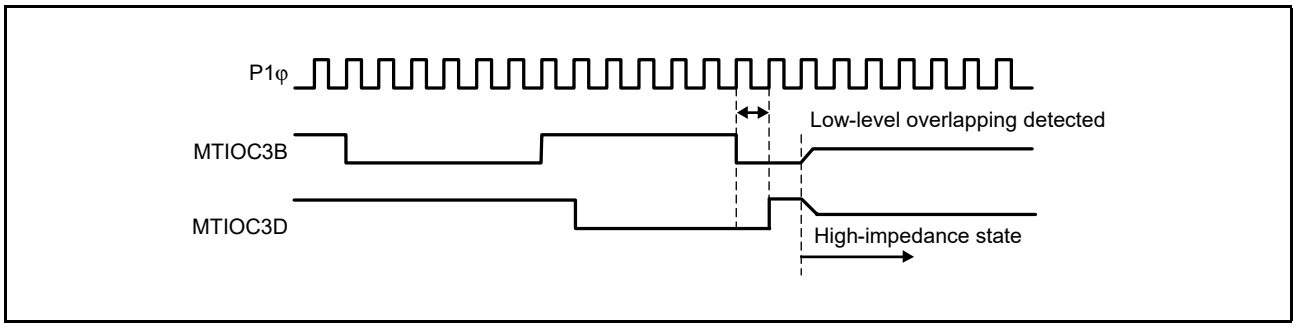


Figure 11.5 Output Level Compare Operation

11.3.4 High-Impedance Control Using Registers

The high-impedance state of the MTU pins (MTU0, MTU3, MTU4, MTU6, and MTU7) can be directly controlled by using the software port output enable register (SPOER).
For instance, setting the MTUCH34HIZ bit in SPOER to 1 places the MTU3 and MTU4 pins specified by the port output enable control register 2 (POECR2) in the high-impedance state.
The high-impedance state of other pins can also be controlled by setting the appropriate bits in SPOER.

11.3.5 Additional Functions for Controlling High-Impedance States

Settings in port enable registers 4 to 5 (POECR4 and POECR5) can add further high-impedance control conditions for the MTU complementary PWM output and MTU0 pins.
For instance, the settings listed below can be added as high-impedance control conditions for the MTU3 and MTU4 pins.

- Setting the IC2ADDMT34ZE bit in POECR4 to 1 adds the input level detection by the POE4#
- Setting the IC3ADDMT34ZE bit in POECR4 to 1 and adds the input level detection by the POE8#
- Setting the IC4ADDMT34ZE bit in POECR4 to 1 and adds the input level detection by the POE10# (ICSR4.POE10F)

The high-impedance state of other pins can also be controlled by setting the appropriate bits in the POECR4 and POECR5.

11.3.6 Release from High-Impedance State

MTU pins which have entered high-impedance state due to input-level detection can be released from the state either by returning them to their initial state with a reset, or by clearing all of the ICSR1.POE0F, ICSR2.POE4F, ICSR3.POE8F, and ICSR4.POE10F flags. However, note that when low-level sampling is selected with the ICSR1.POE0M[1:0], ICSR2.POE4M[1:0], ICSR3.POE8M[1:0], ICSR4.POE10M[1:0], and ICSR5.POE10M[1:0] bits, just writing 0 to a flag is ignored (the flag is not cleared); flags can be cleared by writing 0 to it only after a high level is input to the POE0#, POE4#, POE8#, and POE10# pins and is sampled.

MTU pins which have entered high-impedance state due to output level detection can be released from the state either by returning them to their initial state with a reset, or by clearing the OCSR1.OSF1 flag or the OCSR2.OSF2 flag.

11.4 POE3 Setting Procedure

Figure 11.6 shows the procedure for setting POE3. It illustrates an example of high-impedance control in response to comparison of the output levels on the MTU0 pins (MTIOC3B/MTIOC3D). In the figure, PG2 is selected as the MTIOC3B pin and PG3 is selected as the MTIOC3D pin.

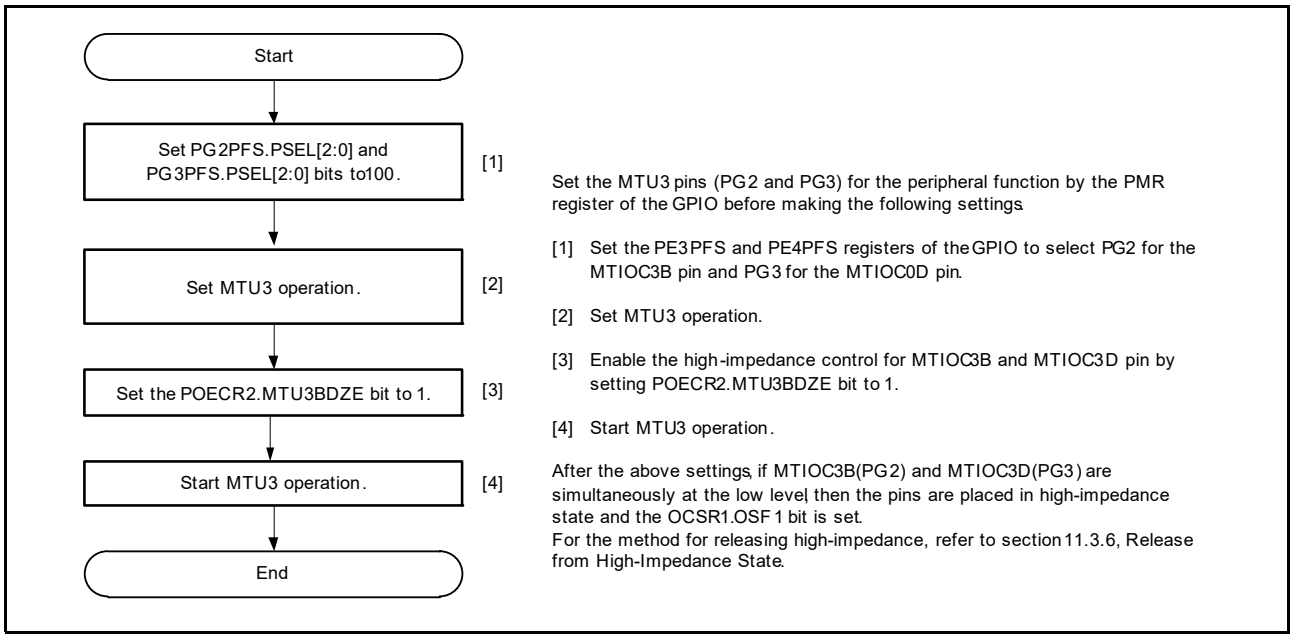


Figure 11.6 Procedure for Setting POE3

11.5 Interrupts

The POE3 issues a request to generate an interrupt when the specified condition is satisfied during input level detection or output level comparison. Table 11.7 shows the interrupt sources and their conditions.

Table 11.7 Interrupt Sources and Conditions

Name	Interrupt Source	Interrupt Flag	Condition
OEI1	Output enable interrupt 1	POE0F and OSF1	PIE1•POE0F+OIE1•OSF1
OEI2	Output enable interrupt 2	POE4F and OSF2	PIE2•POE4F+OIE2•OSF2
OEI3	Output enable interrupt 3	POE8F	PIE3•POE8F
OEI4	Output enable interrupt 4	ICSR4.POE10F	PIE4•ICSR4.POE10F

11.6 Usage Notes

11.6.1 Transition to Low Power Consumption Mode

When the POE3 is used, do not make a transition to software standby mode or deep standby mode. In these modes, the POE3 stops and thus the high-impedance state of pins cannot be controlled.

11.6.2 High-Impedance Control when the MTU Pins are not Selected

If high-impedance control for a pin having a multiplexed MTU pin function is enabled by the POECR1 or POECR2 register and the high-impedance condition is satisfied, the pin is placed in the high-impedance state even if the MTU function is not selected for the given pin at the time.
To avoid unintended high-impedance states, ensure that there are no differences between the settings for MTU pin selection in the PmnPFS registers of the GPIO and for MTU pin selection in the pin selection register of the POE3.

11.6.3 High-Impedance Control when MTU6 and MTU7 are not Used

When MTU6 and MTU7 are not to be used, set the POECR2.MTU6BDZE, MTU7ACZE, and MTU7BDZE bits to 0 to disable high-impedance control.

12. General PWM Timer (GPT)

12.1 Overview

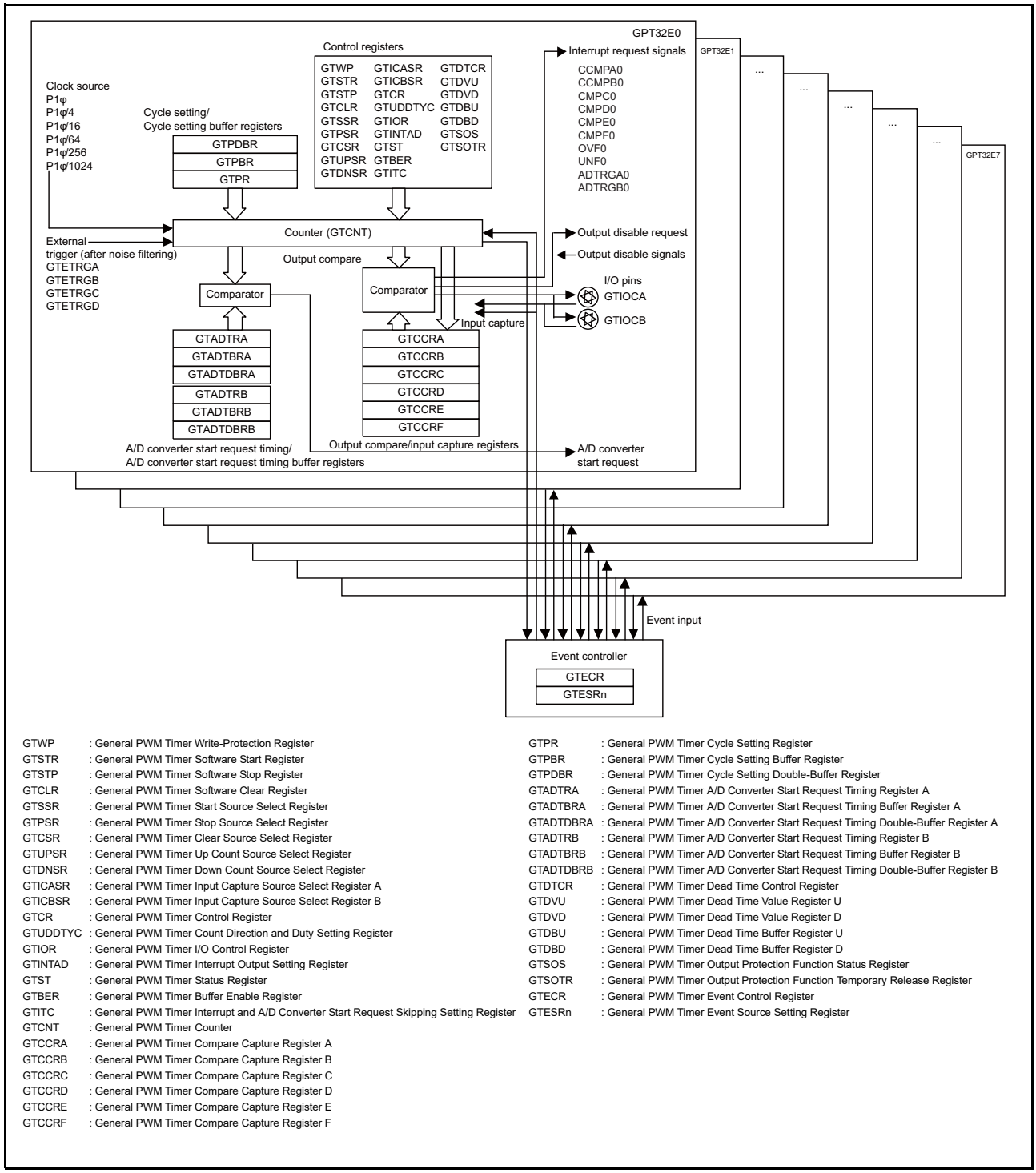
This LSI has a general purpose PWM timer (GPT) composed of 8 channels of 32-bit timer (GPT32E). Table 12.1 lists the GPT specifications, Table 12.2 shows the GPT functions, Figure 12.1 shows the block diagram, and Table 12.3 lists the I/O pins.

Table 12.1 GPT specifications

Parameter	Specifications
Functions	• 32 bits × 8 channels • Up-counting or down-counting (saw waves) or up/down-counting (triangle waves) for each counter. • Clock sources independently selectable for each channel • Two I/O pins per channel • Two output compare/input capture registers per channel • For the two output compare/input capture registers of each channel, four registers are provided as buffer registers and are capable of operating as comparison registers when buffering is not in use. • In output compare operation, buffer switching can be at crests or troughs, enabling the generation of laterally asymmetric PWM waveforms. • Registers for setting up frame cycles in each channel (with capability for generating interrupts at overflow or underflow) • Generation of dead times in PWM operation • Synchronous starting, stopping and clearing counters for arbitrary channels • Starting, stopping, clearing and up/down counters in response to a maximum of eight events • Starting, stopping, clearing and up/down counters in response to input level comparison • Starting, clearing, stopping and up/down counters in response to a maximum of four external triggers • Output pin disable function by dead time error and detected short-circuits between output pins • A/D converter start triggers can be generated (GPT32E0 to GPT32E3) • Compare match A to F event and overflow/underflow event can be output • Enables the noise filter for input capture and external trigger operation

Table 12.2 GPT functions

Parameter		GPT32E
Count clock		P1φ P1φ/4 P1φ/16 P1φ/64 P1φ/256 P1φ/1024
Output compare/input capture registers (GTCCR)		GTCCRA GTCCRB
Compare/buffer registers		GTCCRC GTCCRD GTCCRE GTCCRF
Cycle setting register		GTPR
Cycle setting buffer registers		GTPBR GTPDBR
I/O pins		GTIOCA GTIOCB
External trigger input pin		GTETRGA GTETRGB GTETRGC GTETRGD
Counter clear sources		GTPR register compare match, input capture, input pin status, event input, or input on the GTETRGA, GTETRGB, GTETRGC, or GTETRGD pins
Compare match output	Low output	Available
	High output	Available
	Toggle output	Available
Input capture function		Available
Automatic addition of dead time		Available
PWM mode		Available
Phase count function		Available
Buffer operation		Double buffer
One-shot operation		Available
DMA activation		All the interrupt sources
A/D converter start trigger		Compare match of GTADTRA or GTADTRB (Channel 0 to 3)
Interrupt sources		10 sources • GTCCRA compare match/input capture (CCMPAn) • GTCCRB compare match/input capture (CCMPBn) • GTCCRC compare match (CMPCn) • GTCCRD compare match (CMPDn) • GTCCRE compare match (CMPEn) • GTCCRF compare match (CMPFn) • GTADTRA compare match (ADTRGAn) • GTADTRB compare match (ADTRGBn) • GTCNT overflow (GTPR compare match) (OVFn) • GTCNT underflow (UNFn)
Interrupt skipping function		Skips GTCNT overflows (GTPR compare match) (OVFn)/ GTCNT underflow (UNFn) interrupts (with interlocking function for other interrupts or A/D conversion requests).
Event cooperation function		Available
Noise filtering function		Available



GTWP : General PWM Timer Write-Protection Register
GTSTR : General PWM Timer Software Start Register
GTSTP : General PWM Timer Software Stop Register
GTCLR : General PWM Timer Software Clear Register
GTSSR : General PWM Timer Start Source Select Register
GTPSR : General PWM Timer Stop Source Select Register
GTCSR : General PWM Timer Clear Source Select Register
GTUPSR : General PWM Timer Up Count Source Select Register
GTDNSR : General PWM Timer Down Count Source Select Register
GTICASR : General PWM Timer Input Capture Source Select Register A
GTICBSR : General PWM Timer Input Capture Source Select Register B
GTCR : General PWM Timer Control Register
GTUDDTYC : General PWM Timer Count Direction and Duty Setting Register
GTIOR : General PWM Timer I/O Control Register
GTINTAD : General PWM Timer Interrupt Output Setting Register
GTST : General PWM Timer Status Register
GTBER : General PWM Timer Buffer Enable Register
GTITC : General PWM Timer Interrupt and A/D Converter Start Request Skipping Setting Register
GTCNT : General PWM Timer Counter
GTCCRA : General PWM Timer Compare Capture Register A
GTCCRB : General PWM Timer Compare Capture Register B
GTCCRC : General PWM Timer Compare Capture Register C
GTCCRD : General PWM Timer Compare Capture Register D
GTCCRE : General PWM Timer Compare Capture Register E
GTCCRF : General PWM Timer Compare Capture Register F
GTPR : General PWM Timer Cycle Setting Register
GTPBR : General PWM Timer Cycle Setting Buffer Register
GTPDBR : General PWM Timer Cycle Setting Double-Buffer Register
GTADTRA : General PWM Timer A/D Converter Start Request Timing Register A
GTADTBRA : General PWM Timer A/D Converter Start Request Timing Buffer Register A
GTADTDBRA : General PWM Timer A/D Converter Start Request Timing Double-Buffer Register A
GTADTRB : General PWM Timer A/D Converter Start Request Timing Register B
GTADTBRB : General PWM Timer A/D Converter Start Request Timing Buffer Register B
GTADTDBRB : General PWM Timer A/D Converter Start Request Timing Double-Buffer Register B
GTDTCR : General PWM Timer Dead Time Control Register
GTDVU : General PWM Timer Dead Time Value Register U
GTDVD : General PWM Timer Dead Time Value Register D
GTDBU : General PWM Timer Dead Time Buffer Register U
GTDBD : General PWM Timer Dead Time Buffer Register D
GTSOS : General PWM Timer Output Protection Function Status Register
GTSOTR : General PWM Timer Output Protection Function Temporary Release Register
GTECR : General PWM Timer Event Control Register
GTESRn : General PWM Timer Event Source Setting Register

Figure 12.1 GPT block diagram

Table 12.3 GPT I/O pins

Channel	Pin name	I/O	Function
Shared	GTETRGA	Input	External trigger input pin A (after noise filtering)
	GTETRGB	Input	External trigger input pin B (after noise filtering)
	GTETRGC	Input	External trigger input pin C (after noise filtering)
	GTETRGD	Input	External trigger input pin D (after noise filtering)
GPT32E0	GTIOC0A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC0B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E1	GTIOC1A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC1B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E2	GTIOC2A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC2B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E3	GTIOC3A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC3B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E4	GTIOC4A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC4B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E5	GTIOC5A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC5B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E6	GTIOC6A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC6B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E7	GTIOC7A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC7B	I/O	GTCCRB register input capture input/output compare output/PWM output pin

12.2 Register Descriptions

Table 12.4 lists the registers in the GPT.

Table 12.4 GPT registers

Module symbol	Register name	Register symbol	Reset value	Address (m = 0 to 7)	Access size
GPT32Em (m=0 to 7)	General PWM Timer Write-Protection Register	GTWP	H'00000000	H'E804_3000+H'0100×m	32
	General PWM Timer Software Start Register	GTSTR	H'00000000	H'E804_3004+H'0100×m	32
	General PWM Timer Software Stop Register	GTSTP	H'FFFFFFFF	H'E804_3008+H'0100×m	32
	General PWM Timer Software Clear Register	GTCLR	H'00000000	H'E804_300C+H'0100×m	32
	General PWM Timer Start Source Select Register	GTSSR	H'00000000	H'E804_3010+H'0100×m	32
	General PWM Timer Stop Source Select Register	GTPSR	H'00000000	H'E804_3014+H'0100×m	32
	General PWM Timer Clear Source Select Register	GTCSR	H'00000000	H'E804_3018+H'0100×m	32
	General PWM Timer Up Count Source Select Register	GTUPSR	H'00000000	H'E804_301C+H'0100×m	32
	General PWM Timer Down Count Source Select Register	GTDNSR	H'00000000	H'E804_3020+H'0100×m	32
	General PWM Timer Input Capture Source Select Register A	GTICASR	H'00000000	H'E804_3024+H'0100×m	32
	General PWM Timer Input Capture Source Select Register B	GTICBSR	H'00000000	H'E804_3028+H'0100×m	32
	General PWM Timer Control Register	GTCR	H'00000000	H'E804_302C+H'0100×m	32
	General PWM Timer Count Direction and Duty Setting Register	GTUDDTYC	H'00000001	H'E804_3030+H'0100×m	32
	General PWM Timer I/O Control Register	GTIOR	H'00000000	H'E804_3034+H'0100×m	32
	General PWM Timer Interrupt Output Setting Register	GTINTAD	H'00000000	H'E804_3038+H'0100×m	32
	General PWM Timer Status Register	GTST	H'00008000	H'E804_303C+H'0100×m	32
	General PWM Timer Buffer Enable Register	GTBER	H'00000000	H'E804_3040+H'0100×m	32
	General PWM Timer Interrupt and A/D Converter Start Request Skipping Setting Register	GTITC	H'00000000	H'E804_3044+H'0100×m	32
	General PWM Timer Counter	GTCNT	H'00000000	H'E804_3048+H'0100×m	32
	General PWM Timer Compare Capture Register A	GTCCRA	H'FFFFFFFF	H'E804_304C+H'0100×m	32
	General PWM Timer Compare Capture Register B	GTCCRB	H'FFFFFFFF	H'E804_3050+H'0100×m	32
	General PWM Timer Compare Capture Register C	GTCCRC	H'FFFFFFFF	H'E804_3054+H'0100×m	32
	General PWM Timer Compare Capture Register E	GTCCRE	H'FFFFFFFF	H'E804_3058+H'0100×m	32
	General PWM Timer Compare Capture Register D	GTCCRD	H'FFFFFFFF	H'E804_305C+H'0100×m	32
	General PWM Timer Compare Capture Register F	GTCCRF	H'FFFFFFFF	H'E804_3060+H'0100×m	32
	General PWM Timer Cycle Setting Register	GTPR	H'FFFFFFFF	H'E804_3064+H'0100×m	32

Table 12.4 GPT registers

Module symbol	Register name	Register symbol	Reset value	Address (m = 0 to 7)	Access size
GPT32Em (m=0 to 7)	General PWM Timer Cycle Setting Buffer Register	GTPBR	H'FFFFFFFF	H'E804_3068+H'0100×m	32
	General PWM Timer Cycle Setting Double-Buffer Register	GTPDBR	H'FFFFFFFF	H'E804_306C+H'0100×m	32
	A/D Converter Start Request Timing Register A	GTADTRA	H'FFFFFFFF	H'E804_3070+H'0100×m	32
	A/D Converter Start Request Timing Buffer Register A	GTADTBRA	H'FFFFFFFF	H'E804_3074+H'0100×m	32
	A/D Converter Start Request Timing Double-Buffer Register A	GTADTDBRA	H'FFFFFFFF	H'E804_3078+H'0100×m	32
	A/D Converter Start Request Timing Register B	GTADTRB	H'FFFFFFFF	H'E804_307C+H'0100×m	32
	A/D Converter Start Request Timing Buffer Register B	GTADTBRB	H'FFFFFFFF	H'E804_3080+H'0100×m	32
	A/D Converter Start Request Timing Double-Buffer Register B	GTADTDBRB	H'FFFFFFFF	H'E804_3084+H'0100×m	32
	General PWM Timer Dead Time Control Register	GTDTCR	H'00000000	H'E804_3088+H'0100×m	32
	General PWM Timer Dead Time Value Register U	GTDVU	H'FFFFFFFF	H'E804_308C+H'0100×m	32
	General PWM Timer Dead Time Value Register D	GTDVD	H'FFFFFFFF	H'E804_3090+H'0100×m	32
	General PWM Timer Dead Time Buffer Register U	GTDBU	H'FFFFFFFF	H'E804_3094+H'0100×m	32
	General PWM Timer Dead Time Buffer Register D	GTDBD	H'FFFFFFFF	H'E804_3098+H'0100×m	32
	General PWM Timer Output Protection Function Status Register	GTSOS	H'00000000	H'E804_309C+H'0100×m	32
	General PWM Timer Output Protection Function Temporary Release Register	GTSOTR	H'00000000	H'E804_30A0+H'0100×m	32
GPT	General PWM Timer Event Control Register	GTECR	H'00	H'E804_3800	8
	General PWM Timer Event Source Setting Register 0	GTESR0	H'0000	H'E804_3810	16
	General PWM Timer Event Source Setting Register 1	GTESR1	H'0000	H'E804_3814	16
	General PWM Timer Event Source Setting Register 2	GTESR2	H'0000	H'E804_3818	16
	General PWM Timer Event Source Setting Register 3	GTESR3	H'0000	H'E804_381C	16
	General PWM Timer Event Source Setting Register 4	GTESR4	H'0000	H'E804_3820	16
	General PWM Timer Event Source Setting Register 5	GTESR5	H'0000	H'E804_3824	16
	General PWM Timer Event Source Setting Register 6	GTESR6	H'0000	H'E804_3828	16
	General PWM Timer Event Source Setting Register 7	GTESR7	H'0000	H'E804_382C	16

12.2.1 General PWM Timer Write-Protection Register (GTWP)

Address(es): GPT32Em.GTWP E804 3000h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRKEY[7:0]								—	—	—	—	—	—	—	WP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	WP	Register Write Disable	0: Enable writes to the register 1: Disable writes to the register.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b8	PRKEY[7:0]	GTWP Key Code	When A5h is written to these bits, the writes to the WP bit are permitted. These bits are read as 0.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

To prevent accidental changes, the GTWP enables or disables writing to following registers:

- GTSSR
- GTPSR
- GTCSR
- GTUPSR
- GTDNSR
- GTICASR
- GTICBSR
- GTCR
- GTUDDTYC
- GTIOR
- GTINTAD
- GTST
- GTBER
- GTITC
- GTCNT
- GTCCRA
- GTCCRB
- GTCCRC
- GTCCRD
- GTCCRE
- GTCCRF
- GTPR
- GTPBR
- GTPDBR
- GTADTRA

- GTADTBRA
- GTADTDBRA
- GTADTRB
- GTADTBRB
- GTADTDBRB
- GTDTCR
- GTDVU
- GTDVD
- GTDBU
- GTDBD
- GTSOS
- GTSOTR.

12.2.2 General PWM Timer Software Start Register (GTSTR)

Address(es): GPT32Em.GTSTR E804 3004h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CSTRT7	CSTRT6	CSTRT5	CSTRT4	CSTRT3	CSTRT2	CSTRT1	CSTRT0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The GTSTR starts the GTCNT counter operation for each channel x, where x = 0 to 7.
The GTSTR bit number represents the channel number. The GTSTR register is shared by all of the channels. The GTCNT counter starts for the channel associated with the GTSTR bit number where 1 is written. Writing 0 has no effect on the status of the GTCNT counter and the value of GTSTR register.

CSTRTx bit (channel x GTCNT Count Start) (x = 0 to 7)

The CSTRTx bit starts channel x of the GTCNT counter operation. Writing to GTSTR.CSTRTx bit has no effect unless GPTx.GTSSR.CSTRT bit is set to 1.
Read data shows the counter status of each channel (GTCR.CST bit). Zero means the counter stops and 1 means the counter is running.

12.2.3 General PWM Timer Software Stop Register (GTSTP)

Address(es): GPT32Em.GTSTP E804 3008h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CSTOP 7	CSTOP 6	CSTOP 5	CSTOP 4	CSTOP 3	CSTOP 2	CSTOP 1	CSTOP 0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

The GTSTP stops the GTCNT counter operation for each channel x, where x = 0 to 7.
The GTSTP bit number represents the channel number. The GTSTP register is shared by all of the channels. The GTCNT counter stops for the channel associated with the GTSTP bit number where 1 is written. Writing 0 has no effect on the status of GTCNT counter and the value of GTSTP register.

CSTOPx bit (channel x GTCNT Count Stop) (x = 0 to 7)

The CSTOPx bit stops channel x of the GTCNT counter operation. Writing to GTSTP.CSTOPx bit has no effect unless GPTx.GTPSR.CSTOP bit is set to 1. Read data shows the counter status of each channel (invert of GTCR.CST bit). Zero means the counter is running and 1 means the counter stops.

12.2.4 General PWM Timer Software Clear Register (GTCLR)

Address(es): GPT32Em.GTCLR E804 300Ch + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CCLR 7	CCLR 6	CCLR 5	CCLR 4	CCLR 3	CCLR 2	CCLR 1	CCLR 0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GTCLR is a write-only register that clears the GTCNT counter operation for each channel x, where x = 0 to 7.
The GTCLR bit number represents the channel number. The GTCLR register is shared by all of the channels. The GTCNT counter is cleared for the channel associated with the GTCLR bit number where 1 is written. Writing 0 has no effect on the status of the GTCNT counter.

CCLRx bit (channel x GTCNT Count Clear) (x = 0 to 7).

Channel x of the GTCNT counter value is cleared on writing 1 to the CCLRx bit. This bit is read as 0.

12.2.5 General PWM Timer Start Source Select Register (GTSSR)

Address(es): GPT32Em.GTSSR E804 3010h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CSTRT	—	—	—	—	—	—	—	SSEVTH	SSEVTG	SSEVTF	SSEVTE	SSEVTD	SSEVTC	SSEVTB	SSEVTA
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SSCBFAH	SSCBFAL	SSCBRAH	SSCBRAL	SSCAFBH	SSCAFBL	SSCARBH	SSCARBL	SSGTRGDF	SSGTRGDR	SSGTRGCF	SSGTRGCR	SSGTRGBF	SSGTRGBR	SSGTRGAF	SSGTRGAR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	SSGTRGAR	GTETRGA Pin Rising Input Source Counter Start Enable	0: Disable counter start on the rising edge of GTETRGA input 1: Enable counter start on the rising edge of GTETRGA input	R/W
b1	SSGTRGAF	GTETRGA Pin Falling Input Source Counter Start Enable	0: Disable counter start on the falling edge of GTETRGA input 1: Enable counter start on the falling edge of GTETRGA input	R/W
b2	SSGTRGBR	GTETRGB Pin Rising Input Source Counter Start Enable	0: Disable counter start on the rising edge of GTETRGB input 1: Enable counter start on the rising edge of GTETRGB input	R/W
b3	SSGTRGBF	GTETRGB Pin Falling Input Source Counter Start Enable	0: Disable counter start on the falling edge of GTETRGB input 1: Enable counter start on the falling edge of GTETRGB input	R/W
b4	SSGTRGCR	GTETRGC Pin Rising Input Source Counter Start Enable	0: Disable counter start on the rising edge of GTETRGC input 1: Enable counter start on the rising edge of GTETRGC input	R/W
b5	SSGTRGCF	GTETRGC Pin Falling Input Source Counter Start Enable	0: Disable counter start on the falling edge of GTETRGC input 1: Enable counter start on the falling edge of GTETRGC input	R/W
b6	SSGTRGDR	GTETRGD Pin Rising Input Source Counter Start Enable	0: Disable counter start on the rising edge of GTETRGD input 1: Enable counter start on the rising edge of GTETRGD input	R/W
b7	SSGTRGDF	GTETRGD Pin Falling Input Source Counter Start Enable	0: Disable counter start on the falling edge of GTETRGD input 1: Enable counter start on the falling edge of GTETRGD input	R/W
b8	SSCARBL	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Start Enable	0: Disable counter start on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter start on the rising edge of GTIOCA input when GTIOCB input is 0	R/W
b9	SSCARBH	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Start Enable	0: Disable counter start on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter start on the rising edge of GTIOCA input when GTIOCB input is 1	R/W
b10	SSCAFBL	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Start Enable	0: Disable counter start on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter start on the falling edge of GTIOCA input when GTIOCB input is 0	R/W

Bit	Symbol	Bit name	Description	R/W
b11	SSCAFBH	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Start Enable	0: Disable counter start on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter start on the falling edge of GTIOCA input when GTIOCB input is 1	R/W
b12	SSCBRAL	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Start Enable	0: Disable counter start on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter start on the rising edge of GTIOCB input when GTIOCA input is 0	R/W
b13	SSCBRAH	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Start Enable	0: Disable counter start on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter start on the rising edge of GTIOCB input when GTIOCA input is 1	R/W
b14	SSCBFAL	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Start Enable	0: Disable counter start on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter start on the falling edge of GTIOCB input when GTIOCA input is 0	R/W
b15	SSCBFAH	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Start Enable	0: Disable counter start on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter start on the falling edge of GTIOCB input when GTIOCA input is 1	R/W
b16	SSEVTA	Event Source A Counter Start Enable	0: Disable counter start on event input A 1: Enable counter start on event input A	R/W
b17	SSEVTB	Event Source B Counter Start Enable	0: Disable counter start on event input B 1: Enable counter start on event input B	R/W
b18	SSEVTC	Event Source C Counter Start Enable	0: Disable counter start on event input C 1: Enable counter start on event input C	R/W
b19	SSEVTD	Event Source D Counter Start Enable	0: Disable counter start on event input D 1: Enable counter start on event input D	R/W
b20	SSEVTE	Event Source E Counter Start Enable	0: Disable counter start on event input E 1: Enable counter start on event input E	R/W
b21	SSEVTF	Event Source F Counter Start Enable	0: Disable counter start on event input F 1: Enable counter start on event input F	R/W
b22	SSEVTG	Event Source G Counter Start Enable	0: Disable counter start on event input G 1: Enable counter start on event input G	R/W
b23	SSEVTH	Event Source H Counter Start Enable	0: Disable counter start on event input H 1: Enable counter start on event input H	R/W
b30 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31	CSTRT	Software Source Counter Start Enable	0: Disable counter start by the GTSTR register 1: Enable counter start by the GTSTR register	R/W

GTSSR sets the source to start the GTCNT counter.

SSGTRGAR bit (GTETRGA Pin Rising Input Source Counter Start Enable)

The SSGTRGAR bit enables or disables GTCNT counter start on the rising edge of GTETRGA pin input.

SSGTRGAF bit (GTETRGA Pin Falling Input Source Counter Start Enable)

The SSGTRGAF bit enables or disables GTCNT counter start on the falling edge of GTETRGA pin input.

SSGTRGBR bit (GTETRGB Pin Rising Input Source Counter Start Enable)

The SSGTRGBR bit enables or disables GTCNT counter start on the rising edge of GTETRGB pin input.

SSGTRGBF bit (GTETRGB Pin Falling Input Source Counter Start Enable)

The SSGTRGBF bit enables or disables GTCNT counter start on the falling edge of GTETRGB pin input.

SSGTRGCR bit (GTETRGC Pin Rising Input Source Counter Start Enable)

The SSGTRGCR bit enables or disables GTCNT counter start on the rising edge of GTETRGC pin input.

SSGTRGCF bit (GTETRGC Pin Falling Input Source Counter Start Enable)

The SSGTRGCF bit enables or disables GTCNT counter start on the falling edge of GTETRGC pin input.

SSGTRGDR bit (GTETRGD Pin Rising Input Source Counter Start Enable)

The SSGTRGDR bit enables or disables GTCNT counter start on the rising edge of GTETRGD pin input.

SSGTRGDF bit (GTETRGD Pin Falling Input Source Counter Start Enable)

The SSGTRGDF bit enables or disables GTCNT counter start on the falling edge of GTETRGD pin input.

SSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Start Enable)

The SSCARBL bit enables or disables GTCNT counter start on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

SSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Start Enable)

The SSCARBH bit enables or disables GTCNT counter start on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

SSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Start Enable)

The SSCAFBL bit enables or disables GTCNT counter start on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

SSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Start Enable)

The SSCAFBH bit enables or disables GTCNT counter start on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

SSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Start Enable)

The SSCBRAL bit enables or disables GTCNT counter start on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

SSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Start Enable)

The SSCBRAH bit enables or disables GTCNT counter start on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

SSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Start Enable)

The SSCBFAL bit enables or disables GTCNT counter start on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

SSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Start Enable)

The SSCBFAH bit enables or disables GTCNT counter start on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

SSEVTm bit (Event Source m Counter Start Enable) (m = A to H)

The SSEVTm bit enables or disables GTCNT counter start on the event input m.

CSTRT bit (Software Source Counter Start Enable)

The CSTRT bit enables or disables GTCNT counter start by GTSTR register.

12.2.6 General PWM Timer Stop Source Select Register (GTPSR)

Address(es): GPT32Em.GTPSR E804 3014h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CSTOP	—	—	—	—	—	—	—	PSEVTH	PSEVTG	PSEVTF	PSEVTE	PSEVTD	PSEVTC	PSEVTB	PSEVTA
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PSCBFAH	PSCBFAL	PSCBRAH	PSCBRAL	PSCAFBH	PSCAFBL	PSCARBH	PSCARBL	PSGTRGDF	PSGTRGDR	PSGTRGCF	PSGTRGCR	PSGTRGBF	PSGTRGBR	PSGTRGAF	PSGTRGAR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	PSGTRGAR	GTETRGA Pin Rising Input Source Counter Stop Enable	0: Disable counter stop on the rising edge of GTETRGA input 1: Enable counter stop on the rising edge of GTETRGA input	R/W
b1	PSGTRGAF	GTETRGA Pin Falling Input Source Counter Stop Enable	0: Disable counter stop on the falling edge of GTETRGA input 1: Enable counter stop on the falling edge of GTETRGA input	R/W
b2	PSGTRGBR	GTETRGB Pin Rising Input Source Counter Stop Enable	0: Disable counter stop on the rising edge of GTETRGB input 1: Enable counter stop on the rising edge of GTETRGB input	R/W
b3	PSGTRGBF	GTETRGB Pin Falling Input Source Counter Stop Enable	0: Disable counter stop on the falling edge of GTETRGB input 1: Enable counter stop on the falling edge of GTETRGB input	R/W
b4	PSGTRGCR	GTETRGC Pin Rising Input Source Counter Stop Enable	0: Disable counter stop on the rising edge of GTETRGC input 1: Enable counter stop on the rising edge of GTETRGC input	R/W
b5	PSGTRGCF	GTETRGC Pin Falling Input Source Counter Stop Enable	0: Disable counter stop on the falling edge of GTETRGC input 1: Enable counter stop on the falling edge of GTETRGC input	R/W
b6	PSGTRGDR	GTETRGD Pin Rising Input Source Counter Stop Enable	0: Disable counter stop on the rising edge of GTETRGD input 1: Enable counter stop on the rising edge of GTETRGD input	R/W
b7	PSGTRGDF	GTETRGD Pin Falling Input Source Counter Stop Enable	0: Disable counter stop on the falling edge of GTETRGD input 1: Enable counter stop on the falling edge of GTETRGD input	R/W
b8	PSCARBL	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Stop Enable	0: Disable counter stop on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter stop on the rising edge of GTIOCA input when GTIOCB input is 0	R/W
b9	PSCARBH	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Stop Enable	0: Disable counter stop on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter stop on the rising edge of GTIOCA input when GTIOCB input is 1	R/W
b10	PSCAFBL	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Stop Enable	0: Disable counter stop on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter stop on the falling edge of GTIOCA input when GTIOCB input is 0	R/W

Bit	Symbol	Bit name	Description	R/W
b11	PSCAFBH	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Stop Enable	0: Disable counter stop on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter stop on the falling edge of GTIOCA input when GTIOCB input is 1	R/W
b12	PSCBRAL	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Stop Enable	0: Disable counter stop on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter stop on the rising edge of GTIOCB input when GTIOCA input is 0	R/W
b13	PSCBRAH	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Stop Enable	0: Disable counter stop on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter stop on the rising edge of GTIOCB input when GTIOCA input is 1	R/W
b14	PSCBFAL	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Stop Enable	0: Disable counter stop on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter stop on the falling edge of GTIOCB input when GTIOCA input is 0	R/W
b15	PSCBFAH	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Stop Enable	0: Disable counter stop on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter stop on the falling edge of GTIOCB input when GTIOCA input is 1	R/W
b16	PSEVTA	Event Source A Counter Stop Enable	0: Disable counter stop on event input A 1: Enable counter stop on event input A	R/W
b17	PSEVTB	Event Source B Counter Stop Enable	0: Disable counter stop on event input B 1: Enable counter stop on event input B	R/W
b18	PSEVTC	Event Source C Counter Stop Enable	0: Disable counter stop on event input C 1: Enable counter stop on event input C	R/W
b19	PSEVTD	Event Source D Counter Stop Enable	0: Disable counter stop on event input D 1: Enable counter stop on event input D	R/W
b20	PSEVTE	Event Source E Counter Stop Enable	0: Disable counter stop on event input E 1: Enable counter stop on event input E	R/W
b21	PSEVTF	Event Source F Counter Stop Enable	0: Disable counter stop on event input F 1: Enable counter stop on event input F	R/W
b22	PSEVTG	Event Source G Counter Stop Enable	0: Disable counter stop on event input G 1: Enable counter stop on event input G	R/W
b23	PSEVTH	Event Source H Counter Stop Enable	0: Disable counter stop on event input H 1: Enable counter stop on event input H	R/W
b30 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31	CSTOP	Software Source Counter Stop Enable	0: Disable counter stop by the GTSTP register 1: Enable counter stop by the GTSTP register	R/W

GTPSR sets the source to stop the GTCNT counter.

PSGTRGAR bit (GTETRGA Pin Rising Input Source Counter Stop Enable)

The PSGTRGAR bit enables or disables GTCNT counter stop on the rising edge of GTETRGA pin input.

PSGTRGAF bit (GTETRGA Pin Falling Input Source Counter Stop Enable)

The PSGTRGAF bit enables or disables GTCNT counter stop on the falling edge of GTETRGA pin input.

PSGTRGBR bit (GTETRGB Pin Rising Input Source Counter Stop Enable)

The PSGTRGBR bit enables or disables GTCNT counter stop on the rising edge of GTETRGB pin input.

PSGTRGBF bit (GTETRGB Pin Falling Input Source Counter Stop Enable)

The PSGTRGBF bit enables or disables GTCNT counter stop on the falling edge of GTETRGB pin input.

PSGTRGCR bit (GTETRGC Pin Rising Input Source Counter Stop Enable)

The PSGTRGCR bit enables or disables GTCNT counter stop on the rising edge of GTETRGC pin input.

PSGTRGCF bit (GTETRGC Pin Falling Input Source Counter Stop Enable)

The PSGTRGCF bit enables or disables GTCNT counter stop on the falling edge of GTETRGC pin input.

PSGTRGDR bit (GTETRGD Pin Rising Input Source Counter Stop Enable)

The PSGTRGDR bit enables or disables GTCNT counter stop on the rising edge of GTETRGD pin input.

PSGTRGDF bit (GTETRGD Pin Falling Input Source Counter Stop Enable)

The PSGTRGDF bit enables or disables GTCNT counter stop on the falling edge of GTETRGD pin input.

PSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Stop Enable)

The PSCARBL bit enables or disables GTCNT counter stop on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

PSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Stop Enable)

The PSCARBH bit enables or disables GTCNT counter stop on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

PSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Stop Enable)

The PSCAFBL bit enables or disables GTCNT counter stop on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

PSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Stop Enable)

The PSCAFBH bit enables or disables GTCNT counter stop on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

PSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Stop Enable)

The PSCBRAL bit enables or disables GTCNT counter stop on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

PSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Stop Enable)

The PSCBRAH bit enables or disables GTCNT counter stop on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

PSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Stop Enable)

The PSCBFAL bit enables or disables GTCNT counter stop on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

PSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Stop Enable)

The PSCBFAH bit enables or disables GTCNT counter stop on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

PSEVTm bit (Event Source m Counter Stop Enable) (m = A to H)

The PSEVTm bit enables or disables GTCNT counter stop on the event input m.

CSTOP bit (Software Source Counter Stop Enable)

The CSTOP bit enables or disables GTCNT counter stop by GTSTP register.

12.2.7 General PWM Timer Clear Source Select Register (GTCSR)

Address(es): GPT32Em.GTCSR E804 3018h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CCLR	—	—	—	—	—	—	—	CSEVTH	CSEVTG	CSEVTF	CSEVTE	CSEVTD	CSEVTC	CSEVTB	CSEVTA
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CSCBFAH	CSCBFAL	CSCBRAH	CSCBRAL	CSCAFBH	CSCAFBL	CSCARBH	CSCARBL	CSGTRGDF	CSGTRGDR	CSGTRGCF	CSGTRGCR	CSGTRGBF	CSGTRGBR	CSGTRGAF	CSGTRGAR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	CSGTRGAR	GTETRGA Pin Rising Input Source Counter Clear Enable	0: Disable counter clear on the rising edge of GTETRGA input 1: Enable counter clear on the rising edge of GTETRGA input	R/W
b1	CSGTRGAF	GTETRGA Pin Falling Input Source Counter Clear Enable	0: Disable counter clear on the falling edge of GTETRGA input 1: Enable counter clear on the falling edge of GTETRGA input	R/W
b2	CSGTRGBR	GTETRGB Pin Rising Input Source Counter Clear Enable	0: Disable counter clear on the rising edge of GTETRGB input 1: Enable counter clear on the rising edge of GTETRGB input	R/W
b3	CSGTRGBF	GTETRGB Pin Falling Input Source Counter Clear Enable	0: Disable counter clear on the falling edge of GTETRGB input 1: Enable counter clear on the falling edge of GTETRGB input	R/W
b4	CSGTRGCR	GTETRGC Pin Rising Input Source Counter Clear Enable	0: Disable counter clear on the rising edge of GTETRGC input 1: Enable counter clear on the rising edge of GTETRGC input	R/W
b5	CSGTRGCF	GTETRGC Pin Falling Input Source Counter Clear Enable	0: Disable counter clear on the falling edge of GTETRGC input 1: Enable counter clear on the falling edge of GTETRGC input	R/W
b6	CSGTRGDR	GTETRGD Pin Rising Input Source Counter Clear Enable	0: Disable counter clear on the rising edge of GTETRGD input 1: Enable counter clear on the rising edge of GTETRGD input	R/W
b7	CSGTRGDF	GTETRGD Pin Falling Input Source Counter Clear Enable	0: Disable counter clear on the falling edge of GTETRGD input 1: Enable counter clear on the falling edge of GTETRGD input	R/W
b8	CSCARBL	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Clear Enable	0: Disable counter clear on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter clear on the rising edge of GTIOCA input when GTIOCB input is 0	R/W
b9	CSCARBH	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Clear Enable	0: Disable counter clear on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter clear on the rising edge of GTIOCA input when GTIOCB input is 1	R/W
b10	CSCAFBL	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Clear Enable	0: Disable counter clear on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter clear on the falling edge of GTIOCA input when GTIOCB input is 0	R/W

Bit	Symbol	Bit name	Description	R/W
b11	CSCAFBH	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Clear Enable	0: Disable counter clear on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter clear on the falling edge of GTIOCA input when GTIOCB input is 1	R/W
b12	CSCBRAL	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Clear Enable	0: Disable counter clear on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter clear on the rising edge of GTIOCB input when GTIOCA input is 0	R/W
b13	CSCBRAH	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Clear Enable	0: Disable counter clear on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter clear on the rising edge of GTIOCB input when GTIOCA input is 1	R/W
b14	CSCBFAL	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Clear Enable	0: Disable counter clear on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter clear on the falling edge of GTIOCB input when GTIOCA input is 0	R/W
b15	CSCBFAH	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Clear Enable	0: Disable counter clear on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter clear on the falling edge of GTIOCB input when GTIOCA input is 1	R/W
b16	CSEVTA	EVENT Source A Counter Clear Enable	0: Disable counter clear on event input A 1: Enable counter clear on event input A	R/W
b17	CSEVTB	EVENT Source B Counter Clear Enable	0: Disable counter clear on event input B 1: Enable counter clear on event input B	R/W
b18	CSEVTC	EVENT Source C Counter Clear Enable	0: Disable counter clear on event input C 1: Enable counter clear on event input C	R/W
b19	CSEVTD	EVENT Source D Counter Clear Enable	0: Disable counter clear on event input D 1: Enable counter clear on event input D	R/W
b20	CSEVTE	EVENT Source E Counter Clear Enable	0: Disable counter clear on event input E 1: Enable counter clear on event input E	R/W
b21	CSEVTF	EVENT Source F Counter Clear Enable	0: Disable counter clear on event input F 1: Enable counter clear on event input F	R/W
b22	CSEVTG	EVENT Source G Counter Clear Enable	0: Disable counter clear on event input G 1: Enable counter clear on event input G	R/W
b23	CSEVTH	EVENT Source H Counter Clear Enable	0: Disable counter clear on event input H 1: Enable counter clear on event input H	R/W
b30 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31	CCLR	Software Source Counter Clear Enable	0: Disable counter clear by the GTCLR register 1: Enable counter clear by the GTCLR register	R/W

GTCSR sets the source to clear the GTCNT counter.

CSGTRGAR bit (GTETRGA Pin Rising Input Source Counter Clear Enable)

The CSGTRGAR bit enables or disables GTCNT counter clear on the rising edge of GTETRGA pin input.

CSGTRGAF bit (GTETRGA Pin Falling Input Source Counter Clear Enable)

The CSGTRGAF bit enables or disables GTCNT counter clear on the falling edge of GTETRGA pin input.

CSGTRGBR bit (GTETRGB Pin Rising Input Source Counter Clear Enable)

The CSGTRGBR bit enables or disables GTCNT counter clear on the rising edge of GTETRGB pin input.

CSGTRGBF bit (GTETRGB Pin Falling Input Source Counter Clear Enable)

The CSGTRGBF bit enables or disables GTCNT counter clear on the falling edge of GTETRGB pin input.

CSGTRGCR bit (GTETRGC Pin Rising Input Source Counter Clear Enable)

The CSGTRGCR bit enables or disables GTCNT counter clear on the rising edge of GTETRGC pin input.

CSGTRGCF bit (GTETRGC Pin Falling Input Source Counter Clear Enable)

The CSGTRGCF bit enables or disables GTCNT counter clear on the falling edge of GTETRGC pin input.

CSGTRGDR bit (GTETRGD Pin Rising Input Source Counter Clear Enable)

The CSGTRGDR bit enables or disables GTCNT counter clear on the rising edge of GTETRGD pin input.

CSGTRGDF bit (GTETRGD Pin Falling Input Source Counter Clear Enable)

The CSGTRGDF bit enables or disables GTCNT counter clear on the falling edge of GTETRGD pin input.

CSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Clear Enable)

The CSCARBL bit enables or disables GTCNT counter clear on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

CSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Clear Enable)

The CSCARBH bit enables or disables GTCNT counter clear on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

CSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Clear Enable)

The CSCAFBL bit enables or disables GTCNT counter clear on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

CSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Clear Enable)

The CSCAFBH bit enables or disables GTCNT counter clear on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

CSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Clear Enable)

The CSCBRAL bit enables or disables GTCNT counter clear on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

CSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Clear Enable)

The CSCBRAH bit enables or disables GTCNT counter clear on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

CSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Clear Enable)

The CSCBFAL bit enables or disables GTCNT counter clear on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

CSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Clear Enable)

The CSCBFAH bit enables or disables GTCNT counter clear on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

CSEVTm bit (Event Source m Counter Clear Enable) (m = A to H)

The CSEVm bit enables or disables GTCNT counter clear on the event input m.

CCLR bit (Software Source Counter Clear Enable)

The CCLR bit enables or disables GTCNT counter clear by GTCLR register.

12.2.8 General PWM Timer Up Count Source Select Register (GTUPSR)

Address(es): GPT32Em.GTUPSR E804 301Ch + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	USEVTH	USEVTG	USEVTF	USEVTE	USEVTD	USEVTC	USEVTB	USEVTA
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	USCBFAH	USCBFAL	USCBRAH	USCBRAL	USCAFBH	USCAFBL	USCARBH	USCARBL	USGTRGDF	USGTRGDR	USGTRGCF	USGTRGCR	USGTRGBF	USGTRGBR	USGTRGAF	USGTRGAR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	USGTRGAR	GTETRGA Pin Rising Input Source Counter Count Up Enable	0: Disable counter count up on the rising edge of GTETRGA input 1: Enable counter count up on the rising edge of GTETRGA input.	R/W
b1	USGTRGAF	GTETRGA Pin Falling Input Source Counter Count Up Enable	0: Disable counter count up on the falling edge of GTETRGA input 1: Enable counter count up on the falling edge of GTETRGA input.	R/W
b2	USGTRGBR	GTETRGB Pin Rising Input Source Counter Count Up Enable	0: Disable counter count up on the rising edge of GTETRGB input 1: Enable counter count up on the rising edge of GTETRGB input.	R/W
b3	USGTRGBF	GTETRGB Pin Falling Input Source Counter Count Up Enable	0: Disable counter count up on the falling edge of GTETRGB input 1: Enable counter count up on the falling edge of GTETRGB input.	R/W
b4	USGTRGCR	GTETRGC Pin Rising Input Source Counter Count Up Enable	0: Disable counter count up on the rising edge of GTETRGC input 1: Enable counter count up on the rising edge of GTETRGC input.	R/W
b5	USGTRGCF	GTETRGC Pin Falling Input Source Counter Count Up Enable	0: Disable counter count up on the falling edge of GTETRGC input 1: Enable counter count up on the falling edge of GTETRGC input.	R/W
b6	USGTRGDR	GTETRGD Pin Rising Input Source Counter Count Up Enable	0: Disable counter count up on the rising edge of GTETRGD input 1: Enable counter count up on the rising edge of GTETRGD input.	R/W
b7	USGTRGDF	GTETRGD Pin Falling Input Source Counter Count Up Enable	0: Disable counter count up on the falling edge of GTETRGD input 1: Enable counter count up on the falling edge of GTETRGD input.	R/W
b8	USCARBL	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Count Up Enable	0: Disable counter count up on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter count up on the rising edge of GTIOCA input when GTIOCB input is 0.	R/W
b9	USCARBH	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Count Up Enable	0: Disable counter count up on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter count up on the rising edge of GTIOCA input when GTIOCB input is 1.	R/W
b10	USCAFBL	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Count Up Enable	0: Disable counter count up on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter count up on the falling edge of GTIOCA input when GTIOCB input is 0.	R/W

Bit	Symbol	Bit name	Description	R/W
b11	USCAFBH	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Count Up Enable	0: Disable counter count up on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter count up on the falling edge of GTIOCA input when GTIOCB input is 1.	R/W
b12	USCBRAL	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Count Up Enable	0: Disable counter count up on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter count up on the rising edge of GTIOCB input when GTIOCA input is 0.	R/W
b13	USCBRAH	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Count Up Enable	0: Disable counter count up on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter count up on the rising edge of GTIOCB input when GTIOCA input is 1.	R/W
b14	USCBFAL	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Count Up Enable	0: Disable counter count up on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter count up on the falling edge of GTIOCB input when GTIOCA input is 0.	R/W
b15	USCBFAH	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Count Up Enable	0: Disable counter count up on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter count up on the falling edge of GTIOCB input when GTIOCA input is 1.	R/W
b16	USEVTA	EVENT Source A Counter Count Up Enable	0: Disable counter count up on event input A 1: Enable counter count up on event input A.	R/W
b17	USEVTB	EVENT Source B Counter Count Up Enable	0: Disable counter count up on event input B 1: Enable counter count up on event input B.	R/W
b18	USEVTC	EVENT Source C Counter Count Up Enable	0: Disable counter count up on event input C 1: Enable counter count up on event input C.	R/W
b19	USEVTD	EVENT Source D Counter Count Up Enable	0: Disable counter count up on event input D 1: Enable counter count up on event input D.	R/W
b20	USEVTE	EVENT Source E Counter Count Up Enable	0: Disable counter count up on event input E 1: Enable counter count up on event input E.	R/W
b21	USEVTF	EVENT Source F Counter Count Up Enable	0: Disable counter count up on event input F 1: Enable counter count up on event input F.	R/W
b22	USEVTG	EVENT Source G Counter Count Up Enable	0: Disable counter count up on event input G 1: Enable counter count up on event input G.	R/W
b23	USEVTH	EVENT Source H Counter Count Up Enable	0: Disable counter count up on event input H 1: Enable counter count up on event input H.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

GTUPSR sets the source to count up the GTCNT counter.
When at least 1 bit in the GTUPSR register is set to 1, the GTCNT counter is counted up by the source that is set to 1 in this register. In such cases, the setting of GTCR.TPCS has no effect.

USGTRGAR bit (GTETRGA Pin Rising Input Source Counter Count Up Enable)

The USGTRGAR bit enables or disables GTCNT counter count up on the rising edge of GTETRGA pin input.

USGTRGAF bit (GTETRGA Pin Falling Input Source Counter Count Up Enable)

The USGTRGAF bit enables or disables GTCNT counter count up on the falling edge of GTETRGA pin input.

USGTRGBR bit (GTETRGB Pin Rising Input Source Counter Count Up Enable)

The USGTRGBR bit enables or disables GTCNT counter count up on the rising edge of GTETRGB pin input.

USGTRGBF bit (GTETRGB Pin Falling Input Source Counter Count Up Enable)

The USGTRGBF bit enables or disables GTCNT counter count up on the falling edge of GTETRGB pin input.

USGTRGCR bit (GTETRGC Pin Rising Input Source Counter Count Up Enable)

The USGTRGCR bit enables or disables GTCNT counter count up on the rising edge of GTETRGC pin input.

USGTRGCF bit (GTETRGC Pin Falling Input Source Counter Count Up Enable)

The USGTRGCF bit enables or disables GTCNT counter count up on the falling edge of GTETRGC pin input.

USGTRGDR bit (GTETRGD Pin Rising Input Source Counter Count Up Enable)

The USGTRGDR bit enables or disables GTCNT counter count up on the rising edge of GTETRGD pin input.

USGTRGDF bit (GTETRGD Pin Falling Input Source Counter Count Up Enable)

The USGTRGDF bit enables or disables GTCNT counter count up on the falling edge of GTETRGD pin input.

USCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Count Up Enable)

The USCARBL bit enables or disables GTCNT counter count up on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

USCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Count Up Enable)

The USCARBH bit enables or disables GTCNT counter count up on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

USCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Count Up Enable)

The USCAFBL bit enables or disables GTCNT counter count up on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

USCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Count Up Enable)

The USCAFBH bit enables or disables GTCNT counter count up on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

USCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Count Up Enable)

The USCBRAL bit enables or disables GTCNT counter count up on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

USCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Count Up Enable)

The USCBRAH bit enables or disables GTCNT counter count up on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

USCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Count Up Enable)

The USCBFAL bit enables or disables GTCNT counter count up on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

USCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Count Up Enable)

The USCBFAH bit enables or disables GTCNT counter count up on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

USEVTm bit (Event Source m Counter Count Up Enable) (m = A to H)

The USEVTm bit enables or disables GTCNT counter count up on the event input m.

12.2.9 General PWM Timer Down Count Source Select Register (GTDNSR)

Address(es): GPT32Em.GTDNSR E804 3020h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	DSEVTH	DSEVTG	DSEVTF	DSEVTE	DSEVTD	DSEVTC	DSEVTB	DSEVTA
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DSCBFAH	DSCBFAL	DSCBRAH	DSCBRAL	DSCAFBH	DSCAFBL	DSCARBH	DSCARBL	DSGTRGDF	DSGTRGDR	DSGTRGCF	DSGTRGCR	DSGTRGBF	DSGTRGBR	DSGTRGAF	DSGTRGAR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	DSGTRGAR	GTETRGA Pin Rising Input Source Counter Count Down Enable	0: Disable counter count down on the rising edge of GTETRGA input 1: Enable counter count down on the rising edge of GTETRGA input.	R/W
b1	DSGTRGAF	GTETRGA Pin Falling Input Source Counter Count Down Enable	0: Disable counter count down on the falling edge of GTETRGA input 1: Enable counter count down on the falling edge of GTETRGA input.	R/W
b2	DSGTRGBR	GTETRGB Pin Rising Input Source Counter Count Down Enable	0: Disable counter count down on the rising edge of GTETRGB input 1: Enable counter count down on the rising edge of GTETRGB input.	R/W
b3	DSGTRGBF	GTETRGB Pin Falling Input Source Counter Count Down Enable	0: Disable counter count down on the falling edge of GTETRGB input 1: Enable counter count down on the falling edge of GTETRGB input.	R/W
b4	DSGTRGCR	GTETRGC Pin Rising Input Source Counter Count Down Enable	0: Disable counter count down on the rising edge of GTETRGC input 1: Enable counter count down on the rising edge of GTETRGC input.	R/W
b5	DSGTRGCF	GTETRGC Pin Falling Input Source Counter Count Down Enable	0: Disable counter count down on the falling edge of GTETRGC input 1: Enable counter count down on the falling edge of GTETRGC input.	R/W
b6	DSGTRGDR	GTETRGD Pin Rising Input Source Counter Count Down Enable	0: Disable counter count down on the rising edge of GTETRGD input 1: Enable counter count down on the rising edge of GTETRGD input.	R/W
b7	DSGTRGDF	GTETRGD Pin Falling Input Source Counter Count Down Enable	0: Disable counter count down on the falling edge of GTETRGD input 1: Enable counter count down on the falling edge of GTETRGD input.	R/W
b8	DSCARBL	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Count Down Enable	0: Disable counter count down on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter count down on the rising edge of GTIOCA input when GTIOCB input is 0.	R/W
b9	DSCARBH	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Count Down Enable	0: Disable counter count down on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter count down on the rising edge of GTIOCA input when GTIOCB input is 1.	R/W
b10	DSCAFBL	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Count Down Enable	0: Disable counter count down on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter count down on the falling edge of GTIOCA input when GTIOCB input is 0.	R/W

Bit	Symbol	Bit name	Description	R/W
b11	DSCAFBH	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Count Down Enable	0: Disable counter count down on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter count down on the falling edge of GTIOCA input when GTIOCB input is 1.	R/W
b12	DSCBRAL	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Count Down Enable	0: Disable counter count down on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter count down on the rising edge of GTIOCB input when GTIOCA input is 0.	R/W
b13	DSCBRAH	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Count Down Enable	0: Disable counter count down on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter count down on the rising edge of GTIOCB input when GTIOCA input is 1.	R/W
b14	DSCBFAL	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Count Down Enable	0: Disable counter count down on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter count down on the falling edge of GTIOCB input when GTIOCA input is 0.	R/W
b15	DSCBFAH	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Count Down Enable	0: Disable counter count down on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter count down on the falling edge of GTIOCB input when GTIOCA input is 1.	R/W
b16	DSEVTA	EVENT Source A Counter Count Down Enable	0: Disable counter count down on event input A 1: Enable counter count down on event input A.	R/W
b17	DSEVTB	EVENT Source B Counter Count Down Enable	0: Disable counter count down on event input B 1: Enable counter count down on event input B.	R/W
b18	DSEVTC	EVENT Source C Counter Count Down Enable	0: Disable counter count down on event input C 1: Enable counter count down on event input C.	R/W
b19	DSEVTD	EVENT Source D Counter Count Down Enable	0: Disable counter count down on event input D 1: Enable counter count down on event input D.	R/W
b20	DSEVTE	EVENT Source E Counter Count Down Enable	0: Disable counter count down on event input E 1: Enable counter count down on event input E.	R/W
b21	DSEVTF	EVENT Source F Counter Count Down Enable	0: Disable counter count down on event input F 1: Enable counter count down on event input F.	R/W
b22	DSEVTG	EVENT Source G Counter Count Down Enable	0: Disable counter count down on event input G 1: Enable counter count down on event input G.	R/W
b23	DSEVTH	EVENT Source H Counter Count Down Enable	0: Disable counter count down on event input H 1: Enable counter count down on event input H.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

GTDNSR sets the source to count down the GTCNT counter.
When at least 1 bit in the GTDNSR register is set to 1, the GTCNT counter is counted down by the source that is set to 1 in this register. In such cases, the setting of GTCR.TPCS has no effect.

DSGTRGAR bit (GTETRGA Pin Rising Input Source Counter Count Down Enable)

The DSGTRGAR bit enables or disables GTCNT counter count down on the rising edge of GTETRGA pin input.

DSGTRGAF bit (GTETRGA Pin Falling Input Source Counter Count Down Enable)

The DSGTRGAF bit enables or disables GTCNT counter count down on the falling edge of GTETRGA pin input.

DSGTRGBR bit (GTETRGB Pin Rising Input Source Counter Count Down Enable)

The DSGTRGBR bit enables or disables GTCNT counter count down on the rising edge of GTETRGB pin input.

DSGTRGBF bit (GTETRGB Pin Falling Input Source Counter Count Down Enable)

The DSGTRGBF bit enables or disables GTCNT counter count down on the falling edge of GTETRGB pin input.

DSGTRGCR bit (GTETRGC Pin Rising Input Source Counter Count Down Enable)

The DSGTRGCR bit enables or disables GTCNT counter count down on the rising edge of GTETRGC pin input.

DSGTRGCF bit (GTETRGC Pin Falling Input Source Counter Count Down Enable)

The DSGTRGCF bit enables or disables GTCNT counter count down on the falling edge of GTETRGC pin input.

DSGTRGDR bit (GTETRGD Pin Rising Input Source Counter Count Down Enable)

The DSGTRGDR bit enables or disables GTCNT counter count down on the rising edge of GTETRGD pin input.

DSGTRGDF bit (GTETRGD Pin Falling Input Source Counter Count Down Enable)

The DSGTRGDF bit enables or disables GTCNT counter count down on the falling edge of GTETRGD pin input.

DSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Count Down Enable)

The DSCARBL bit enables or disables GTCNT counter count down on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

DSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Count Down Enable)

The DSCARBH bit enables or disables GTCNT counter count down on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

DSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Count Down Enable)

The DSCAFBL bit enables or disables GTCNT counter count down on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

DSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Count Down Enable)

The DSCAFBH bit enables or disables GTCNT counter count down on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

DSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Count Down Enable)

The DSCBRAL bit enables or disables GTCNT counter count down on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

DSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Count Down Enable)

The DSCBRAH bit enables or disables GTCNT counter count down on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

DSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Count Down Enable)

The DSCBFAL bit enables or disables GTCNT counter count down on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

DSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Count Down Enable)

The DSCBFAH bit enables or disables GTCNT counter count down on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

DSEVTm bit (Event Source m Counter Count Down Enable) (m = A to H)

The DSEVTm bit enables or disables GTCNT counter count down on the event input m.

12.2.10 General PWM Timer Input Capture Source Select Register A (GTICASR)

Address(es): GPT32Em.GTICASR E804 3024h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	ASEVTH	ASEVTG	ASEVTF	ASEVTE	ASEVTD	ASEVTC	ASEVTB	ASEVTA
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ASCBFAH	ASCBFAL	ASCBRAH	ASCBRAL	ASCAFBH	ASCAFBL	ASCARBH	ASCARBL	ASGTRGDF	ASGTRGDR	ASGTRGCF	ASGTRGCR	ASGTRGBF	ASGTRGBR	ASGTRGAF	ASGTRGAR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	ASGTRGAR	GTETRGA Pin Rising Input Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the rising edge of GTETRGA input 1: Enable GTCCRA input capture on the rising edge of GTETRGA input.	R/W
b1	ASGTRGAF	GTETRGA Pin Falling Input Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the falling edge of GTETRGA input 1: Enable GTCCRA input capture on the falling edge of GTETRGA input.	R/W
b2	ASGTRGBR	GTETRGB Pin Rising Input Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the rising edge of GTETRGB input 1: Enable GTCCRA input capture on the rising edge of GTETRGB input.	R/W
b3	ASGTRGBF	GTETRGB Pin Falling Input Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the falling edge of GTETRGB input 1: Enable GTCCRA input capture on the falling edge of GTETRGB input.	R/W
b4	ASGTRGCR	GTETRGC Pin Rising Input Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the rising edge of GTETRGC input 1: Enable GTCCRA input capture on the rising edge of GTETRGC input.	R/W
b5	ASGTRGCF	GTETRGC Pin Falling Input Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the falling edge of GTETRGC input 1: Enable GTCCRA input capture on the falling edge of GTETRGC input.	R/W
b6	ASGTRGDR	GTETRGD Pin Rising Input Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the rising edge of GTETRGD input 1: Enable GTCCRA input capture on the rising edge of GTETRGD input.	R/W
b7	ASGTRGDF	GTETRGD Pin Falling Input Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the falling edge of GTETRGD input 1: Enable GTCCRA input capture on the falling edge of GTETRGD input.	R/W
b8	ASCARBL	GTIOCA Pin Rising Input during GTIOCB Value Low Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable GTCCRA input capture on the rising edge of GTIOCA input when GTIOCB input is 0.	R/W
b9	ASCARBH	GTIOCA Pin Rising Input during GTIOCB Value High Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable GTCCRA input capture on the rising edge of GTIOCA input when GTIOCB input is 1.	R/W
b10	ASCAFBL	GTIOCA Pin Falling Input during GTIOCB Value Low Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable GTCCRA input capture on the falling edge of GTIOCA input when GTIOCB input is 0.	R/W

Bit	Symbol	Bit name	Description	R/W
b11	ASCAFBH	GTIOCA Pin Falling Input during GTIOCB Value High Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable GTCCRA input capture on the falling edge of GTIOCA input when GTIOCB input is 1.	R/W
b12	ASCBRAL	GTIOCB Pin Rising Input during GTIOCA Value Low Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable GTCCRA input capture on the rising edge of GTIOCB input when GTIOCA input is 0.	R/W
b13	ASCBRAH	GTIOCB Pin Rising Input during GTIOCA Value High Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable GTCCRA input capture on the rising edge of GTIOCB input when GTIOCA input is 1.	R/W
b14	ASCBFAL	GTIOCB Pin Falling Input during GTIOCA Value Low Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable GTCCRA input capture on the falling edge of GTIOCB input when GTIOCA input is 0.	R/W
b15	ASCBFAH	GTIOCB Pin Falling Input during GTIOCA Value High Source GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable GTCCRA input capture on the falling edge of GTIOCB input when GTIOCA input is 1.	R/W
b16	ASEVTA	EVENT Source A GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on event input A 1: Enable GTCCRA input capture on event input A.	R/W
b17	ASEVTB	EVENT Source B GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on event input B 1: Enable GTCCRA input capture on event input B.	R/W
b18	ASEVTC	EVENT Source C GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on event input C 1: Enable GTCCRA input capture on event input C.	R/W
b19	ASEVTD	EVENT Source D GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on event input D 1: Enable GTCCRA input capture on event input D.	R/W
b20	ASEVTE	EVENT Source E GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on event input E 1: Enable GTCCRA input capture on event input E.	R/W
b21	ASEVTF	EVENT Source F GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on event input F 1: Enable GTCCRA input capture on event input F.	R/W
b22	ASEVTG	EVENT Source G GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on event input G 1: Enable GTCCRA input capture on event input G.	R/W
b23	ASEVTH	EVENT Source H GTCCRA Input Capture Enable	0: Disable GTCCRA input capture on event input H 1: Enable GTCCRA input capture on event input H.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

GTICASR sets the source of input capture for GTCCRA.

ASGTRGAR bit (GTETRGA Pin Rising Input Source GTCCRA Input Capture Enable)

The ASGTRGAR bit enables or disables input capture for GTCCRA on the rising edge of GTETRGA pin input.

ASGTRGAF bit (GTETRGA Pin Falling Input Source GTCCRA Input Capture Enable)

The ASGTRGAF bit enables or disables input capture for GTCCRA on the falling edge of GTETRGA pin input.

ASGTRGBR bit (GTETRGB Pin Rising Input Source GTCCRA Input Capture Enable)

The ASGTRGBR bit enables or disables input capture for GTCCRA on the rising edge of GTETRGB pin input.

ASGTRGBF bit (GTETRGB Pin Falling Input Source GTCCRA Input Capture Enable)

The ASGTRGBF bit enables or disables input capture for GTCCRA on the falling edge of GTETRGB pin input.

ASGTRGCR bit (GTETRGC Pin Rising Input Source GTCCRA Input Capture Enable)

The ASGTRGCR bit enables or disables input capture for GTCCRA on the rising edge of GTETRGC pin input.

ASGTRGCF bit (GTETRGC Pin Falling Input Source GTCCRA Input Capture Enable)

The ASGTRGCF bit enables or disables input capture for GTCCRA on the falling edge of GTETRGC pin input.

ASGTRGDR bit (GTETRGD Pin Rising Input Source GTCCRA Input Capture Enable)

The ASGTRGDR bit enables or disables input capture for GTCCRA on the rising edge of GTETRGD pin input.

ASGTRGDF bit (GTETRGD Pin Falling Input Source GTCCRA Input Capture Enable)

The ASGTRGDF bit enables or disables input capture for GTCCRA on the falling edge of GTETRGD pin input.

ASCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source GTCCRA Input Capture Enable)

The ASCARBL bit enables or disables input capture for GTCCRA on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

ASCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source GTCCRA Input Capture Enable)

The ASCARBH bit enables or disables input capture for GTCCRA on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

ASCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source GTCCRA Input Capture Enable)

The ASCAFBL bit enables or disables input capture for GTCCRA on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

ASCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source GTCCRA Input Capture Enable)

The ASCAFBH bit enables or disables input capture for GTCCRA on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

ASCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source GTCCRA Input Capture Enable)

The ASCBRAL bit enables or disables input capture for GTCCRA on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

ASCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source GTCCRA Input Capture Enable)

The ASCBRAH bit enables or disables input capture for GTCCRA on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

ASCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source GTCCRA Input Capture Enable)

The ASCBFAL bit enables or disables input capture for GTCCRA on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

ASCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source GTCCRA Input Capture Enable)

The ASCBFAH bit enables or disables input capture for GTCCRA on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

ASEVTm bit (Event Source m Counter GTCCRA Input Capture Enable) (m = A to H)

The ASEVTm bit enables or disables input capture for GTCCRA on the event input m.

12.2.11 General PWM Timer Input Capture Source Select Register B (GTICBSR)

Address(es): GPT32Em.GTICBSR E804 3028h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	BSEVTH	BSEVTG	BSEVTF	BSEVTE	BSEVTD	BSEVTC	BSEVTB	BSEVTA
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BSCBFAH	BSCBFAL	BSCBRAH	BSCBRAL	BSCAFBH	BSCAFBL	BSCARBH	BSCARBL	BSGTRGDF	BSGTRGDR	BSGTRGCF	BSGTRGCR	BSGTRGBF	BSGTRGBR	BSGTRGAF	BSGTRGAR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	BSGTRGAR	GTETRGA Pin Rising Input Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the rising edge of GTETRGA input 1: Enable GTCCRB input capture on the rising edge of GTETRGA input.	R/W
b1	BSGTRGAF	GTETRGA Pin Falling Input Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the falling edge of GTETRGA input 1: Enable GTCCRB input capture on the falling edge of GTETRGA input.	R/W
b2	BSGTRGBR	GTETRGB Pin Rising Input Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the rising edge of GTETRGB input 1: Enable GTCCRB input capture on the rising edge of GTETRGB input.	R/W
b3	BSGTRGBF	GTETRGB Pin Falling Input Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the falling edge of GTETRGB input 1: Enable GTCCRB input capture on the falling edge of GTETRGB input.	R/W
b4	BSGTRGCR	GTETRGC Pin Rising Input Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the rising edge of GTETRGC input 1: Enable GTCCRB input capture on the rising edge of GTETRGC input.	R/W
b5	BSGTRGCF	GTETRGC Pin Falling Input Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the falling edge of GTETRGC input 1: Enable GTCCRB input capture on the falling edge of GTETRGC input.	R/W
b6	BSGTRGDR	GTETRGD Pin Rising Input Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the rising edge of GTETRGD input 1: Enable GTCCRB input capture on the rising edge of GTETRGD input.	R/W
b7	BSGTRGDF	GTETRGD Pin Falling Input Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the falling edge of GTETRGD input 1: Enable GTCCRB input capture on the falling edge of GTETRGD input.	R/W
b8	BSCARBL	GTIOCA Pin Rising Input during GTIOCB Value Low Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable GTCCRB input capture on the rising edge of GTIOCA input when GTIOCB input is 0.	R/W
b9	BSCARBH	GTIOCA Pin Rising Input during GTIOCB Value High Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable GTCCRB input capture on the rising edge of GTIOCA input when GTIOCB input is 1.	R/W
b10	BSCAFBL	GTIOCA Pin Falling Input during GTIOCB Value Low Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable GTCCRB input capture on the falling edge of GTIOCA input when GTIOCB input is 0.	R/W

Bit	Symbol	Bit name	Description	R/W
b11	BSCAFBH	GTIOCA Pin Falling Input during GTIOCB Value High Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable GTCCRB input capture on the falling edge of GTIOCA input when GTIOCB input is 1.	R/W
b12	BSCBRAL	GTIOCB Pin Rising Input during GTIOCA Value Low Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable GTCCRB input capture on the rising edge of GTIOCB input when GTIOCA input is 0.	R/W
b13	BSCBRAH	GTIOCB Pin Rising Input during GTIOCA Value High Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable GTCCRB input capture on the rising edge of GTIOCB input when GTIOCA input is 1.	R/W
b14	BSCBFAL	GTIOCB Pin Falling Input during GTIOCA Value Low Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable GTCCRB input capture on the falling edge of GTIOCB input when GTIOCA input is 0.	R/W
b15	BSCBFAH	GTIOCB Pin Falling Input during GTIOCA Value High Source GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable GTCCRB input capture on the falling edge of GTIOCB input when GTIOCA input is 1.	R/W
b16	BSEVTA	EVENT Source A GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on event input A 1: Enable GTCCRB input capture on event input A.	R/W
b17	BSEVTB	EVENT Source B GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on event input B 1: Enable GTCCRB input capture on event input B.	R/W
b18	BSEVTC	EVENT Source C GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on event input C 1: Enable GTCCRB input capture on event input C.	R/W
b19	BSEVTD	EVENT Source D GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on event input D 1: Enable GTCCRB input capture on event input D.	R/W
b20	BSEVTE	EVENT Source E GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on event input E 1: Enable GTCCRB input capture on event input E.	R/W
b21	BSEVTF	EVENT Source F GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on event input F 1: Enable GTCCRB input capture on event input F.	R/W
b22	BSEVTG	EVENT Source G GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on event input G 1: Enable GTCCRB input capture on event input G.	R/W
b23	BSEVTH	EVENT Source H GTCCRB Input Capture Enable	0: Disable GTCCRB input capture on event input H 1: Enable GTCCRB input capture on event input H.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

GTICBSR sets the source of input capture for GTCCRB.

BSGTRGAR bit (GTETRGA Pin Rising Input Source GTCCRB Input Capture Enable)

The BSGTRGAR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGA pin input.

BSGTRGAF bit (GTETRGA Pin Falling Input Source GTCCRB Input Capture Enable)

The BSGTRGAF bit enables or disables input capture for GTCCRB on the falling edge of GTETRGA pin input.

BSGTRGBR bit (GTETRGB Pin Rising Input Source GTCCRB Input Capture Enable)

The BSGTRGBR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGB pin input.

BSGTRGBF bit (GTETRGB Pin Falling Input Source GTCCRB Input Capture Enable)

The BSGTRGBF bit enables or disables input capture for GTCCRB on the falling edge of GTETRGB pin input.

BSGTRGCR bit (GTETRGC Pin Rising Input Source GTCCRB Input Capture Enable)

The BSGTRGCR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGC pin input.

BSGTRGCF bit (GTETRGC Pin Falling Input Source GTCCRB Input Capture Enable)

The BSGTRGCF bit enables or disables input capture for GTCCRB on the falling edge of GTETRGC pin input.

BSGTRGDR bit (GTETRGD Pin Rising Input Source GTCCRB Input Capture Enable)

The BSGTRGDR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGD pin input.

BSGTRGDF bit (GTETRGD Pin Falling Input Source GTCCRB Input Capture Enable)

The BSGTRGDF bit enables or disables input capture for GTCCRB on the falling edge of GTETRGD pin input.

BSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source GTCCRB Input Capture Enable)

The BSCARBL bit enables or disables input capture for GTCCRB on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

BSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source GTCCRB Input Capture Enable)

The BSCARBH bit enables or disables input capture for GTCCRB on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

BSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source GTCCRB Input Capture Enable)

The BSCAFBL bit enables or disables input capture for GTCCRB on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

BSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source GTCCRB Input Capture Enable)

The BSCAFBH bit enables or disables input capture for GTCCRB on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

BSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source GTCCRB Input Capture Enable)

The BSCBRAL bit enables or disables input capture for GTCCRB on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

BSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source GTCCRB Input Capture Enable)

The BSCBRAH bit enables or disables input capture for GTCCRB on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

BSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source GTCCRB Input Capture Enable)

The BSCBFAL bit enables or disables input capture for GTCCRB on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

BSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source GTCCRB Input Capture Enable)

The BSCBFAH bit enables or disables input capture for GTCCRB on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

BSEVTm bit (Event Source m Counter GTCCRB Input Capture Enable) (m = A to H)

The BSEVTm bit enables or disables input capture for GTCCRB on the event input m.

12.2.12 General PWM Timer Control Register (GTCR)

Address(es): GPT32Em.GTCR E804 302Ch + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	TPCS[2:0]			—	—	—	—	—	MD[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	CST	Count Start	0: Stop count operation 1: Perform count operation.	R/W
b15 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b18 to b16	MD[2:0]	Mode Select	b18 b17 b16 0 0 0: Saw-wave PWM mode (single buffer or double buffer possible) 0 0 1: Saw-wave one-shot pulse mode (fixed buffer operation) 0 1 0: Setting prohibited 0 1 1: Setting prohibited 1 0 0: Triangle-wave PWM mode 1 (32-bit transfer at trough) (single buffer or double buffer possible) 1 0 1: Triangle-wave PWM mode 2 (32-bit transfer at crest and trough) (single buffer or double buffer possible) 1 1 0: Triangle-wave PWM mode 3 (64-bit transfer at trough) fixed buffer operation) 1 1 1: Setting prohibited.	R/W
b23 to b19	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b26 to b24	TPCS[2:0]	Timer Prescaler Select	b26 b25 b24 0 0 0: P1φ/1 0 0 1: P1φ/4 0 1 0: P1φ/16 0 1 1: P1φ/64 1 0 0: P1φ/256 1 0 1: P1φ/1024.	R/W
b31 to b27	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

GTCR controls GTCNT.

CST bit (Count Start)

The CST bit controls the GTCNT counter start and stop.

[Setting conditions]

- GTSTR value where the channel number associated with the bit number is set to 1 with the GTSSR.CSTRT bit being 1.
- Any among event input or input on the GTIOCA, GTIOCB, or GTETRGA to GTETRGD pins that is enabled in the GTSSR register as a source for the start of counting
- 1 is written by software directly.

[Clearing conditions]

- GTSTP value where the channel number associated with the bit number is set to 1 with the GTPSR.CSTOP bit being 1.

- Any among event input or input on the GTIOCA, GTIOCB, or GTETRGA to GTETRGD pins that is enabled in the GTPSR register as a source for the stop of counting
- 0 is written by software directly.

MD[2:0] bits (Mode Select)

The MD[2:0] bits select the GPT operating mode. The MD[2:0] bits must be set while the GTCNT operation is stopped.

TPCS[2:0] bits (Timer Prescaler Select)

The TPCS[2:0] bits select the clock for GTCNT. A clock prescaler can be selected independently for each channel. The TPCS[2:0] bits must be set while the GTCNT operation is stopped.

12.2.13 General PWM Timer Count Direction and Duty Setting Register (GTUDDTYC)

Address(es): GPT32Em.GTUDDTYC E804 3030h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	OBDTYR	OBDTYF	OBDTY[1:0]		—	—	—	—	OADTYR	OADTYF	OADTY[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UDF	UD
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit name	Description	R/W
b0	UD	Count Direction Setting	0: Count down on GTCNT 1: Counts up on GTCNT.	R/W
b1	UDF	Forcible Count Direction Setting	0: Do not force setting 1: Force setting.	R/W
b15 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b17, b16	OADTY[1:0]	GTIOCA Output Duty Setting	b17 b16 0 x: GTIOCA pin duty depends on compare match 1 0: GTIOCA pin duty 0% 1 1: GTIOCA pin duty 100%.	R/W
b18	OADTYF	Forcible GTIOCA Output Duty Setting	0: Do not force setting 1: Force setting.	R/W
b19	OADTYR	GTIOCA Output Value Selecting after Releasing 0%/100% Duty Setting	0: Apply output value set in 0%/100% duty to GTIOA[3:2] function after releasing 0%/100% duty setting 1: Apply masked compare match output value to GTIOA[3:2] function after releasing 0%/100% duty setting.	R/W
b23 to b20	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b25, b24	OBDTY[1:0]	GTIOCB Output Duty Setting	b25 b24 0 x: GTIOCB pin duty depends on compare match 1 0: GTIOCB pin duty = 0% 1 1: GTIOCB pin duty = 100%.	R/W
b26	OBDTYF	Forcible GTIOCB Output Duty Setting	0: Do not force setting 1: Force setting.	R/W
b27	OBDTYR	GTIOCB Output Value Selecting after Releasing 0%/100% Duty Setting	0: Apply output value set in 0%/100% duty to GTIOB[3:2] function after releasing 0%/100% duty setting 1: Apply masked compare match output value to GTIOB[3:2] function after releasing 0%/100% duty setting.	R/W
b31 to b28	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

x: Don't care

GTUDDTYC sets the direction in which GTCNT counts (up-counting or down-counting) and sets the duty of GTIOCA/GTIOCB pin output.

Count direction:

- In saw-wave mode

When the UD value is set to 0 during up-counting, the count direction changes at an overflow (the timing synchronous with count clock after the GTCNT value becomes the GTPR value). When the UD value is set to 1 during down-counting, the count direction changes at an underflow (the timing synchronous with count clock after the GTCNT value becomes 0).

When the UD value changes from 1 to 0 with the UDF bit being 0 and while counting is stopped, the counter starts up-counting and the count direction changes at an overflow (the timing synchronous with count clock after the GTCNT

value becomes GTPR value). When the UD value changes from 0 to 1 with the UDF bit being 0 and while counting stops, the counter starts down-counting and the count direction changes at an underflow (the timing synchronous with count clock after the GTCNT value becomes 0).
When the UDF bit is set to 1 while counting stops, the UD bit value is reflected in the count direction when counting starts.

- In triangle-wave mode

When the UD value changes during counting, the count direction does not change. When the UD value changes while the UDF bit is 0 and counting stops, the change is not reflected in the count direction when counting starts.
When the UDF bit is set to 1 while counting stops, the UD value is reflected in the count direction when counting starts.

UD bit (Count Direction Setting)

The UD bit sets the count direction for up-counting or down-counting, for GTCNT.

UDF bit (Forcible Count Direction Setting)

The UDF bit forcibly sets the count direction when GTCNT starts operation as the UD value. Only write 0 to this bit during counter operation. When 1 is written to this bit while counting stops, return this bit to 0 before counting starts.

Output duty :

- In saw-wave mode

When the OADTY/OBDTY value changes during up-counting, the duty is reflected at an overflow (GTCNT = GTPR).
When the OADTY/OBDTY value changes during down-counting, the duty is reflected at an underflow (GTCNT = 0).
When the OADTY/OBDTY value changes to 1 with the OADTYF/OBDTYF bit being 0 and while counting is stopped, the output duty is not reflected at starting counter operation. When the count direction is up, the output duty is reflected at an overflow (GTCNT = GTPR). When the count direction is down, the output duty is reflected at an underflow (GTCNT = 0). When the OADTY/OBDTY value changes to 0 with the OADTYF/OBDTYF bit being 1 and while counting stops, the output duty is reflected at the starting counter operation.

- In triangle-wave mode

When the OADTY/OBDTY value changes during counting, the duty is reflected at an underflow. When the OADTY/OBDTY value changes to 1 with the OADTYF/OBDTYF bit being 0 and while counting stops, the output duty is not reflected at the starting counter operation, however the output duty is reflected at an underflow.
When the OADTY/OBDTY value changes to 0 with the OADTYF/OBDTYF bit being 1 and while counting stops, the output duty is reflected at starting counter operation.

OmDTY[1:0] bits (GTIOCm Output Duty Setting) (m = A, B)

The OmDTY[1:0] bits set the output duty (0%, 100% or compare match control) of the GTIOCm pin.

OmDTYF bit (Forcible GTIOCm Output Duty Setting) (m = A, B)

The OmDTYF bit forcibly sets the output duty cycle to the OmDTY setting. Set this bit to 0 during counter operation. When this bit is set to 1 while counting is stopped, return this bit to 0 until the first period ends after the counter starts.

OmDTYR bit (GTIOCm Output Value Selecting after Releasing 0%/100% Duty Setting) (m = A, B)

The OmDTYR bits select the value that is the object of output retained or toggled at cycle end, when the control changes from 0%/100% duty setting to compare match for GTIOCm pin and GTIOR.GTIOm[3:2] are set to 00b (output retained at cycle end) or GTIOR.GTIOm[3:2] are set to 11b (output toggled at cycle end).
GPT32 internally continues the compare match operation in performing 0%/100% duty operation. When the OmDTYR bit is set to 1, the GTCIOm pin outputs the value specified in the GTIOR.GTIOm[3:2] bits at the end of compare match cycle.

12.2.14 General PWM Timer I/O Control Register (GTIOR)

Address(es): GPT32Em.GTIOR E804 3034h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	NFCSB[1:0]		NFBEN	—	—	OBDF[1:0]		OBE	OBHLD	OBDFLT	—	GTIOB[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	NFCSA[1:0]		NFAEN	—	—	OADF[1:0]		OAE	OAHLD	OADFLT	—	GTIOA[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b4 to b0	GTIOA[4:0]	GTIOCA Pin Function Select	See Table 12.5.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	OADFLT	GTIOCA Pin Output Value Setting at the Count Stop	0: Output low on GTIOCA pin when counting stops 1: Output high on GTIOCA pin when counting stops.	R/W
b7	OAHLD	GTIOCA Pin Output Setting at the Start/Stop Count	0: Set GTIOCA pin output level on counting start and stop based on the register setting. 1: Retain GTIOCA pin output level on counting start and stop.	R/W
b8	OAE	GTIOCA Pin Output Enable	0: Disable output 1: Enable output.	R/W
b10, b9	OADF[1:0]	GTIOCA Pin Disable Value Setting	b10 b9 0 0: Prohibit output disable 0 1: Set GTIOCA pin to Hi-Z on output disable 1 0: Set GTIOCA pin to 0 on output disable 1 1: Set GTIOCA pin to 1 on output disable.	R/W
b12, b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b13	NFAEN	Noise Filter A Enable	0: Disable noise filter for GTIOCA pin 1: Enable noise filter for GTIOCA pin.	R/W
b15, b14	NFCSA[1:0]	Noise Filter A Sampling Clock Select	b15 b14 0 0: P1φ/1 0 1: P1φ/4 1 0: P1φ/16 1 1: P1φ/64.	R/W
b20 to b16	GTIOB[4:0]	GTIOCB Pin Function Select	See Table 12.5.	R/W
b21	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b22	OBDFLT	GTIOCB Pin Output Value Setting at the Count Stop	0: Output low on GTIOCB pin when counting stops 1: Output high on GTIOCB pin when counting stops.	R/W
b23	OBHLD	GTIOCB Pin Output Setting at the Start/Stop Count	0: Set GTIOCB pin output level on counting start and stop based on the register setting 1: Retain GTIOCB pin output level on counting start and stop.	R/W
b24	OBE	GTIOCB Pin Output Enable	0: Disable output 1: Enable output.	R/W
b26, b25	OBDF[1:0]	GTIOCB Pin Disable Value Setting	b26 b25 0 0: Prohibit output disable 0 1: Set GTIOCB pin to Hi-Z on output disable 1 0: Set GTIOCB pin to 0 on output disable 1 1: Set GTIOCB pin to 1 on output disable.	R/W
b28, b27	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b29	NFBEN	Noise Filter B Enable	0: Disable noise filter for GTIOCB pin 1: Enable noise filter for GTIOCB pin.	R/W

Bit	Symbol	Bit name	Description	R/W
b31, b30	NFCSB[1:0]	Noise Filter B Sampling Clock Select	b31 b30 0 0: P1φ/1 0 1: P1φ/4 1 0: P1φ/16 1 1: P1φ/64.	R/W

GTIOR sets the functions of the GTIOCA and GTIOCB pins.

GTIOA[4:0] bits (GTIOCA Pin Function Select)

The GTIOA[4:0] bits select the GTIOCA pin function. For details, see Table 12.5

OADFLT bit (GTIOCA Pin Output Value Setting at the Count Stop)

The OADFLT bit sets whether the GTIOCA pin outputs high or low when counting stops.

OAHLD bit (GTIOCA Pin Output Setting at the Start/Stop Count)

The OAHLD bit specifies whether the GTIOCA pin output level is retained or the level at the start/stop of counting depends on the register setting.

[When the OAHLD bit is set to 0]

- The value specified in bit [4] of the GTIOA[4:0] bits is output when counting starts
- The value specified in the OADFLT bit is output when counting stops
- If the OADFLT bit is modified while counting stops, it is immediately reflected in the output.

[When the OAHLD bit is set to 1]

- The output is retained when counting starts or stops.

OAE bit (GTIOCA Pin Output Enable)

The OAE bit disables or enables the GTIOCA pin output.

When the GTCCRA register is in use as an input capture register (at least one bit in the GTICASR register is set to 1), signals will not be output from the GTIOCA pin regardless of the setting of this bit.

OADF[1:0] bits (GTIOCA Pin Disable Value Setting)

The OADF bits select a value to be output from the GTIOCA pin in response to the output disable requests.

NFAEN bit (Noise Filter A Enable)

The NFAEN bit disables or enables the noise filter for input from the GTIOCA pin. Because changing the value of the bit might lead to internal generation of an unexpected edge, select the output compare function for the relevant pin in the GTIOR register before doing so.

NFCSA[1:0] bits (Noise Filter A Sampling Clock Select)

The NFCSA[1:0] bits set the sampling interval for the noise filter of the GTIOCA pin. When setting these bits, wait for two cycles of the selected sampling interval before setting the input-capture function.

GTIOB[4:0] bits (GTIOCB Pin Function Select)

The GTIOB[4:0] bits select the GTIOCB pin function. For details, see Table 12.5

OBDFLT bit (GTIOCB Pin Output Value Setting at the Count Stop)

The OBDFLT bit sets whether the GTIOCB pin outputs high or low when counting stops.

OBHLD bit (GTIOCB Pin Output Setting at the Start/Stop Count)

The OBHLD bit specifies whether the GTIOCB pin output level is retained or the level at the start/stop of counting depends on the register setting.

[When the OBHLD bit is set to 0]

- The value specified in bit [4] of the GTIOB[4:0] bits is output when counting starts
- The value specified in the OBDFLT bit is output when counting stops
- If the OBDFLT bit is modified while counting stops, it is immediately reflected in the output.

[When the OBHLD bit is set to 1]

- The output is retained when counting starts or stops.

OBE bit (GTIOCB Pin Output Enable)

The OBE bit disables or enables the GTIOCB pin output.

When the GTCCRB register is in use as an input capture register (at least one bit in the GTICBSR register is set to 1), signals will not be output from the GTIOCB pin regardless of the setting of this bit.

OBDF[1:0] bits (GTIOCB Pin Disable Value Setting)

The OBDF bits select a value to be output from the GTIOCB pin in response to the output disable requests.

NFBEN bit (Noise Filter B Enable)

The NFBEN bit disables or enables the noise filter for input from the GTIOCB pin. Because changing the value of the bit might lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the GTIOR register before doing so.

NFCSB[1:0] bits (Noise Filter B Sampling Clock Select)

The NFCSB[1:0] bits set the sampling interval for the noise filter of the GTIOCB pin. When setting these bits, wait for two cycles of the selected sampling interval before setting the input-capture function.

Table 12.5 Settings of GTIOA[4:0] and GTIOB[4:0] bits

GTIOA/GTIOB[4:0] bits					Function		
b4	b3	b2	b1	b0	b4	b3, b2	b1, b0
0	0	0	0	0	Set initial output low	Retain output at cycle end	Retain output at GTCCRA/GTCCRB compare match
0	0	0	0	1			Output low at GTCCRA/GTCCRB compare match
0	0	0	1	0			Output high at GTCCRA/GTCCRB compare match
0	0	0	1	1			Toggle output at GTCCRA/GTCCRB compare match
0	0	1	0	0		Output low at cycle end	Retain output at GTCCRA/GTCCRB compare match
0	0	1	0	1			Output low at GTCCRA/GTCCRB compare match
0	0	1	1	0			Output high at GTCCRA/GTCCRB compare match
0	0	1	1	1			Toggle output at GTCCRA/GTCCRB compare match
0	1	0	0	0		Output high at cycle end	Retain output at GTCCRA/GTCCRB compare match
0	1	0	0	1			Output low at GTCCRA/GTCCRB compare match
0	1	0	1	0			Output high at GTCCRA/GTCCRB compare match
0	1	0	1	1			Toggle output at GTCCRA/GTCCRB compare match
0	1	1	0	0		Toggle output at cycle end	Retain output at GTCCRA/GTCCRB compare match
0	1	1	0	1			Output low at GTCCRA/GTCCRB compare match
0	1	1	1	0			Output high at GTCCRA/GTCCRB compare match
0	1	1	1	1			Toggle output at GTCCRA/GTCCRB compare match
1	0	0	0	0	Set initial output high	Retain output at cycle end	Retain output at GTCCRA/GTCCRB compare match
1	0	0	0	1			Output low at GTCCRA/GTCCRB compare match
1	0	0	1	0			Output high at GTCCRA/GTCCRB compare match
1	0	0	1	1			Toggle output at GTCCRA/GTCCRB compare match
1	0	1	0	0		Output low at cycle end	Retain output at GTCCRA/GTCCRB compare match
1	0	1	0	1			Output low at GTCCRA/GTCCRB compare match
1	0	1	1	0			Output high at GTCCRA/GTCCRB compare match
1	0	1	1	1			Toggle output at GTCCRA/GTCCRB compare match
1	1	0	0	0		Output high at cycle end	Retain output at GTCCRA/GTCCRB compare match
1	1	0	0	1			Output low at GTCCRA/GTCCRB compare match
1	1	0	1	0			Output high at GTCCRA/GTCCRB compare match
1	1	0	1	1			Toggle output at GTCCRA/GTCCRB compare match
1	1	1	0	0		Toggle output at cycle end	Retain output at GTCCRA/GTCCRB compare match
1	1	1	0	1			Output low at GTCCRA/GTCCRB compare match
1	1	1	1	0			Output high at GTCCRA/GTCCRB compare match
1	1	1	1	1			Toggle output at GTCCRA/GTCCRB compare match

Note 1. The cycle end means an overflow (GTCNT is changed from GTPR to 0 in up-counting) or underflow (GTCNT is changed from 0 to GTPR in down-counting). In this case, the GTCNT counter is cleared for saw waves and for the trough (GTCNT is changed from 0 to 1) for triangle waves.

Note 2. When the timing of a cycle end and the timing of a GTCCRA/GTCCRB compare match are the same in a compare-match operation, the b3 and b2 settings are given priority in saw-wave PWM mode, and the b1 and b0 settings are given priority in any other mode.

Note 3. In event count operation where at least 1 bit in GTUPSR or GTDNSR is set to 1, the setting of b3 and b2 is ignored.

12.2.15 General PWM Timer Interrupt Output Setting Register (GTINTAD)

Address(es): GPT32Em.GTINTAD E804 3038h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	GRPABL	GRPABH	GRPDTE	—	—	GRP[1:0]		—	—	—	—	ADTRBDEN	ADTRBUEN	ADTRADEN	ADTRAUEN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	GTINTPR[1:0]		GTINTF	GTINTE	GTINTD	GTINTC	GTINTB	GTINTA
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	GTINTA	GTCCRA Compare Match/Input Capture Interrupt Enable	0: Disable Interrupt request 1: Enable Interrupt request.	R/W
b1	GTINTB	GTCCRB Compare Match/Input Capture Interrupt Enable	0: Disable Interrupt request 1: Enable Interrupt request.	R/W
b2	GTINTC	GTCCRC Compare Match Interrupt Enable	0: Disable Interrupt request 1: Enable Interrupt request.	R/W
b3	GTINTD	GTCCRD Compare Match Interrupt Enable	0: Disable Interrupt request 1: Enable Interrupt request.	R/W
b4	GTINTE	GTCCRE Compare Match Interrupt Enable	0: Disable Interrupt request 1: Enable Interrupt request.	R/W
b5	GTINTF	GTCCRF Compare Match Interrupt Enable	0: Disable Interrupt request 1: Enable Interrupt request.	R/W
b7, b6	GTINTPR[1:0]	GTPR Compare Match Interrupt Enable	b7 b6 0 0: Disable Interrupt request 0 1: Interrupt requests in response to overflows for saw waves and crests for triangle waves are enabled. 1 0: Interrupt requests in response to underflows for saw waves and troughs for triangle waves are enabled. 1 1: Interrupt requests in response to overflows and underflows for saw waves and crests and troughs for triangle waves are enabled.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b16	ADTRAUEN	GTADTRA Compare Match (Up-Counting) A/D Converter Start Request Enable	0: Disable A/D converter start request 1: Enable A/D converter start request.	R/W
b17	ADTRADEN	GTADTRA Compare Match (Down-Counting) A/D Converter Start Request Enable	0: Disable A/D converter start request 1: Enable A/D converter start request.	R/W
b18	ADTRBUEN	GTADTRB Compare Match (Up-Counting) A/D Converter Start Request Enable	0: Disable A/D converter start request 1: Enable A/D converter start request.	R/W
b19	ADTRBDEN	GTADTRB Compare Match (Down-Counting) A/D Converter Start Request Enable	0: Disable A/D converter start request 1: Enable A/D converter start request.	R/W
b23 to b20	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b25, b24	GRP[1:0]	Output Disable Source Select	b25 b24 0 0: Select Group A output disable request 0 1: Select Group B output disable request 1 0: Select Group C output disable request 1 1: Select Group D output disable request.	R/W
b27, b26	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Bit	Symbol	Bit name	Description	R/W
b28	GRPDTE	Dead Time Error Output Disable Request Enable	0: Disable dead time error output disable request 1: Enable dead time error output disable request.	R/W
b29	GRPABH	Same Time Output Level High Disable Request Enable	0: Disable same time output level high disable request 1: Enable same time output level high disable request.	R/W
b30	GRPABL	Same Time Output Level Low Disable Request Enable	0: Disable same time output level low disable request 1: Enable same time output level low disable request.	R/W
b31	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

GTINTAD enables or disables interrupt requests, A/D converter start requests and output disable requests.

GTINTA bit (GTCCRA Compare Match/Input Capture Interrupt Enable)

The GTINTA bit enables and disables the interrupt request (CCMPAn) in response to compare matches with or input capture by the GTCCRA register.

GTINTB bit (GTCCRB Compare Match/Input Capture Interrupt Enable)

The GTINTB bit enables and disables the interrupt request (CCMPBn) in response to compare matches with or input capture by the GTCCRB register.

GTINTC bit (GTCCRC Compare Match Interrupt Enable)

The GTINTC bit enables and disables the interrupt request (CMPCn) in response to compare matches with the GTCCRC register.

GTINTD bit (GTCCRD Compare Match Interrupt Enable)

The GTINTD bit enables and disables the interrupt request (CMPDn) in response to compare matches with the GTCCRD register.

GTINTE bit (GTCCRE Compare Match Interrupt Enable)

The GTINTE bit enables and disables the interrupt request (CMPEn) in response to compare matches with the GTCCRE register.

GTINTF bit (GTCCRF Compare Match Interrupt Enable)

The GTINTF bit enables and disables the interrupt request (CMPFn) in response to compare matches with the GTCCRF register.

GTINTPR[1:0] bits (GTPR Compare Match Interrupt Enable)

The GTINTPR[1:0] bits enable and disable the interrupt requests (OVFn and UNFn) in response to compare matches with the GTPR register (and overflows of the GTCNT counter) and underflows of the GTCNT counter.

ADTRAUEN bit (GTADTRA Compare Match (Up-Counting) A/D Converter Start Request Enable)

The ADTRAUEN bit enables or disables A/D converter start requests generated by GTADTRA compare matches during GTCNT up-counting.

ADTRADEN bit (GTADTRA Compare Match (Down-Counting) A/D Converter Start Request Enable)

The ADTRADEN bit enables or disables A/D converter start requests generated by GTADTRA compare matches during GTCNT down-counting.

ADTRBUEN bit (GTADTRB Compare Match (Up-Counting) A/D Converter Start Request Enable)

The ADTRBUEN bit enables or disables A/D converter start requests generated by GTADTRB compare matches during GTCNT up-counting.

ADTRBDEN bit (GTADTRB Compare Match (Down-Counting) A/D Converter Start Request Enable)

The ADTRBDEN bit enables or disables A/D converter start requests generated by GTADTRB compare matches during GTCNT down-counting.

GRP[1:0] bits (Output Disable Source Select)

The GRP[1:0] bits select GTIOCA pin and GTIOCB pin output disable source. The output disable requests to the POEG are output to the group selected by the GRP[1:0] bits. Each of the output disable sources, a dead time error, simultaneous output of high-level, and low-level signal, is also output in accord with the setting of its enabling bit. GTST.ODF shows the request of output disable source group that is selected with the GRP[1:0] bits. Set the GRP[1:0] bits when GTIOR.OAE and OBE bits are both 0.

GRPDTE bit (Dead Time Error Output Disable Request Enable)

The GRPDTE bit enables or disables dead time error output disable request.

GRPABH bit (Same Time Output Level High Disable Request Enable)

The GRPABH bit enables or disables output disable request when GTIOCA pin and GTIOCB pin output 1 at the same time.

GRPABL bit (Same Time Output Level Low Disable Request Enable)

The GRPABL bit enables or disables output disable request when GTIOCA pin and GTIOCB pin output 0 at the same time.

12.2.16 General PWM Timer Status Register (GTST)

Address(es): GPT32Em.GTST E804 303Ch + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	OABLF	OABHF	DTEF	—	—	—	ODF	—	—	—	—	ADTRBDF	ADTRBUF	ADTRADF	ADTRAUF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TUCF	—	—	—	—	ITCNT[2:0]			TCFPU	TCFPO	TCFF	TCFE	TCFD	TCFC	TCFB	TCFA
Value after reset:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	TCFA	Input Capture/Compare Match Flag A	0: No input capture/compare match of GTCCRA occurred 1: Input capture/compare match of GTCCRA occurred.	R/(W)*1
b1	TCFB	Input Capture/Compare Match Flag B	0: No input capture/compare match of GTCCRB occurred 1: Input capture/compare match of GTCCRB occurred.	R/(W)*1
b2	TCFC	Compare Match Flag C	0: No compare match of GTCCRC occurred 1: Compare match of GTCCRC occurred.	R/(W)*1
b3	TCFD	Compare Match Flag D	0: No compare match of GTCCRD occurred 1: Compare match of GTCCRD occurred.	R/(W)*1
b4	TCFE	Compare Match Flag E	0: No compare match of GTCCRE occurred 1: Compare match of GTCCRE occurred.	R/(W)*1
b5	TCFF	Compare Match Flag F	0: No compare match of GTCCRF occurred 1: Compare match of GTCCRF occurred.	R/(W)*1
b6	TCFPO	Overflow Flag	0: No overflow (crest) occurred 1: Overflow (crest) occurred.	R/(W)*1
b7	TCFPU	Underflow Flag	0: No underflow (trough) occurred 1: Underflow (trough) occurred.	R/(W)*1
b10 to b8	ITCNT[2:0]	OVFn/UNFn Interrupt Skipping Count Counter	Counter for counting the number of times a timer interrupt is skipped.	R
b14 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	TUCF	Count Direction Flag	0: GTCNT counter is counting down 1: GTCNT counter is counting up.	R
b16	ADTRAUF	GTADTRA Compare Match (Counting Up) A/D Converter Start Request Flag	0: A compare match with the GTADTRA register has not occurred during counting up. 1: A compare match with the GTADTRA register has occurred during counting up.	R/(W)*1
b17	ADTRADF	GTADTRA Compare Match (Counting Down) A/D Converter Start Request Flag	0: A compare match with the GTADTRA register has not occurred during counting down. 1: A compare match with the GTADTRA register has occurred during counting down.	R/(W)*1
b18	ADTRBUF	GTADTRB Compare Match (Counting Up) A/D Converter Start Request Flag	0: A compare match with the GTADTRB register has not occurred during counting up. 1: A compare match with the GTADTRB register has occurred during counting up.	R/(W)*1
b19	ADTRBDF	GTADTRB Compare Match (Counting Down) A/D Converter Start Request Flag	0: A compare match with the GTADTRB register has not occurred during counting down. 1: A compare match with the GTADTRB register has occurred during counting down.	R/(W)*1
b23 to b20	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b24	ODF	Output Disable Flag	0: No output disable request occurred 1: Output disable request occurred.	R
b27 to b25	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Bit	Symbol	Bit name	Description	R/W
b28	DTEF	Dead Time Error Flag	0: No dead time error occurred 1: Dead time error occurred.	R
b29	OABHF	Same Time Output Level High Flag	0: GTIOCA pin and GTIOCB pin did not output 1 at the same time 1: GTIOCA pin and GTIOCB pin output 1 at the same time.	R
b30	OABLF	Same Time Output Level Low Flag	0: GTIOCA pin and GTIOCB pin did not output 0 at the same time 1: GTIOCA pin and GTIOCB pin output 0 at the same time.	R
b31	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Note 1. Only 0 can be written to this bit. Do not write 1.

GTST indicates the status of the GPT.

TCFA flag (Input Capture/Compare Match Flag A)

The TCFS flag indicates the status for the input capture or compare match of GTCCRA.

[Setting conditions]

- GTCNT = GTCCRA, when the GTCCRA register functions as a compare match register
- GTCNT counter value is transferred to GTCCRA by the input capture signal when the GTCCRA register functions as an input capture register.

[Clearing condition]

- 0 is written to this flag.

TCFB flag (Input Capture/Compare Match Flag B)

The TCFB flag indicates the status for the input capture or compare match of GTCCRB.

[Setting conditions]

- GTCNT = GTCCRB, when the GTCCRB register functions as a compare match register
- GTCNT counter value is transferred to GTCCRB by the input capture signal when the GTCCRB register function as an input capture register.

[Clearing condition]

- 0 is written to this flag.

TCFC flag (Compare Match Flag C)

The TCFC flag indicates the status for the compare match of GTCCRC.

[Setting condition]

- GTCNT = GTCCRC

[Clearing condition]

- 0 is written to this flag.

[Not comparing condition]

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 01b, 10b, 11b (GTCCRC performs buffer operation).

TCFD flag (Compare Match Flag D)

The TCFD flag indicates the status for the compare match of GTCCRD.

[Setting condition]

- GTCNT = GTCCRD.

[Clearing condition]

- 0 is written to this flag.

[Not comparing condition]

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (Triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 10b, 11b (GTCCRD performs buffer operation).

TCFE flag (Compare Match Flag E)

The TCFE flag indicates the status for the compare match of GTCCRE.

[Setting condition]

- GTCNT = GTCCRE.

[Clearing condition]

- 0 is written to this flag.

[Not comparing condition]

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (Triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 01b, 10b, 11b (GTCCRE performs buffer operation).

TCFF flag (Compare Match Flag F)

The TCFF flag indicates the status for the compare match of GTCCRF.

[Setting condition]

- GTCNT = GTCCRF.

[Clearing condition]

- 0 is written to this flag.

[Not comparing condition]

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (Triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 10b, 11b (GTCCRF performs buffer operation).

TCFPO flag (Overflow Flag)

The TCFPO flag indicates when an overflow or a crest has occurred.

[Setting conditions]

- In saw-wave mode, an overflow (GTCNT changes from GTPR to 0 in up-counting) has occurred
- In triangle-wave mode, a crest (GTCNT changes from GTPR to GTPR-1) has occurred
- In counting by hardware sources, an overflow (GTCNT changes from GTPR to 0 in up-counting) has occurred.

[Clearing condition]

- 0 is written to this flag.

TCFPU flag (Underflow Flag)

The TCFPU flag indicates when an underflow or a trough has occurred.

[Setting conditions]

- In saw-wave mode, an underflow (GTCNT changes from 0 to GTPR in down-counting) has occurred
- In triangle-wave mode, a crest (GTCNT changes from 0 to 1) has occurred
- In counting by hardware sources, an underflow (GTCNT changes from 0 to GTPR in down-counting) has occurred.

[Clearing condition]

- 0 is written to this flag.

ITCNT[2:0] bits (OVFn/UNFn Interrupt Skipping Count Counter)

When the OVFn/UNFn (n=0 to 7) interrupt skipping function is used (the GTITC.IVTC[1:0] bits are set to a value other than 00b), the counter in the ITCNT[2:0] bits increments by 1 every time the OVFn/UNFn interrupt source that is selected in GTITC.IVTC[1:0] is generated.

[Clearing conditions]

- The OVFn/UNFn interrupt skipping function is not used (GTITC.IVTT[2:0] is 000b when GTITC.IVTC[1:0] is 00b)
- The OVFn/UNFn interrupt skipping count matches the specified count (ITCNT[2:0] matches the skipping count specified in GTITC.IVTT[2:0]).

TUCF flag (Count Direction Flag)

The TUCF flag indicates the count direction of GTCNT. In event count operation, this flag is set to 1 in up-counting and is set to 0 in down-counting.

ADTRAUF flag (GTADTRA Compare Match (Counting Up) A/D Converter Start Request Flag)

The ADTRAUF flag indicates whether or not a compare match with the GTADTRA register has occurred during counting up.

[Setting condition]

- GTCNT = GTADTRA during counting up

[Clearing condition]

- 0 is written to this flag.

ADTRADF flag (GTADTRA Compare Match (Counting Down) A/D Converter Start Request Flag)

The ADTRADF flag indicates whether or not a compare match with the GTADTRA register has occurred during counting down.

[Setting condition]

- GTCNT = GTADTRA during counting down

[Clearing condition]

- 0 is written to this flag.

ADTRBUF flag (GTADTRB Compare Match (Counting Up) A/D Converter Start Request Flag)

The ADTRBUF flag indicates whether or not a compare match with the GTADTRB register has occurred during counting up.

[Setting condition]

- GTCNT = GTADTRB during counting up

[Clearing condition]

- 0 is written to this flag.

ADTRBDF flag (GTADTRB Compare Match (Counting Down) A/D Converter Start Request Flag)

The ADTRBDF flag indicates whether or not a compare match with the GTADTRB register has occurred during counting down.

[Setting condition]

- GTCNT = GTADTRB during counting down

[Clearing condition]

- 0 is written to this flag.

ODF flag (Output Disable Flag)

The ODF flag shows the request of the output disable source group that is selected by GRP[1:0] bits. When output is disabled, an output disable control is not released within the same cycle in which an output disable request is negated. It is released in the next cycle.

DTEF flag (Dead Time Error Flag)

The DTEF flag indicates that the timer output toggle point after the automatic addition of dead time has exceeded the timer cycle.
This flag returns to 0 when the timer output toggle point after the automatic addition of dead time is returned to the cycle.
This flag is read only. Writing 0 to clear the flag is not allowed.
When an interrupt by the DTEF flag is enabled (GTINTAD.GRPDTE = 1), the DTEF flag is output to POEG as an output disable request each time the DTEF flag changes from 0 to 1.
[Setting condition]

- The timer output toggle point after the automatic addition of dead time has exceeded the timer cycle.
  For triangle wave in up-counting: GTCCRA – GTDVU ≤ 0
  For triangle wave in down-counting: GTCCRA – GTDVD < 0
  For saw wave 1 shot pulse mode in up-counting:
  GTCCRA – GTDVU < 0 or GTCCRA + GTDVD > GTPR
  For saw wave 1 shot pulse mode in down-counting:
  GTCCRA + GTDVU > GTPR or GTCCRA - GTDVD < 0

[Clearing condition]

- The timer output toggle point after the automatic addition of dead time is within the timer cycle.

OABHF flag (Same Time Output Level High Flag)

The OABHF flag indicates that GTIOCA pin and GTIOCB pin output 1 at the same time.
When GTIOCA pin or GTIOCB pin output 0, this flag returns to 0. This flag is read only. Writing 0 to clear the flag is not allowed. When an interrupt by the OABHF flag is enabled (GTINTAD.GRPABH = 1), the OABHF flag is output to POEG as an output disable request.
[Setting condition]

- GTIOCA pin and GTIOCB pin output 1 at the same time when both the OAE and OBE bits are set to 1.

[Clearing conditions]

- GTIOCA pin output value is different from GTIOCB pin output value when both the OAE and OBE bits are set to 1
- GTIOCA pin and GTIOCB pin output 0 at the same time when both the OAE and OBE bits are set to 1
- Either OAE bit or OBE bit is set to 0.

OABLF flag (Same Time Output Level Low Flag)

The OABLF flag indicates that GTIOCA pin and GTIOCB pin output 0 at the same time.
When GTIOCA pin or GTIOCB pin output 1, this flag returns to 0. This flag is read only. Writing 0 to clear the flag is not allowed. When an interrupt by the OABLF flag is enabled (GTINTAD.GRPABL = 1), the OABLF flag is output to POEG as an output disable request.
[Setting condition]

- GTIOCA pin and GTIOCB pin output 0 at the same time when both the OAE and OBE bits are set to 1.

[Clearing conditions]

- GTIOCA pin output value is different from GTIOCB pin output value when both the OAE and OBE bits are set to 1
- GTIOCA pin and GTIOCB pin output 1 at the same time when both the OAE and OBE bits are set to 1
- At least either OAE bit or OBE bit is set to 0.

The compare-target signals to generate the OABHF/OABLF flag are the compare match outputs (PWM outputs) signals before masked by the output disable function. When the output disable state is performed, a compare match also

performs continuously in the GPT and the OABHF/OABLF flag is updated in association with the result of the compared values.

12.2.17 General PWM Timer Buffer Enable Register (GTBER)

Address(es): GPT32Em.GTBER E804 3040h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	ADTDB	ADTTB[1:0]		—	ADTDA	ADTTA[1:0]		—	CCRSWT	PR[1:0]		CCRB[1:0]		CCRA[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	BD[3]	BD[2]	BD[1]	BD[0]
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	BD[0]	GTCCR Buffer Operation Disable	0: Enable buffer operation 1: Disable buffer operation.	R/W
b1	BD[1]	GTPR Buffer Operation Disable		R/W
b2	BD[2]	GTADTR Buffer Operation Disable		R/W
b3	BD[3]	GTDV Buffer Operation Disable		R/W
b15 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b17, b16	CCRA[1:0]	GTCCRA Buffer Operation	b17 b16 0 0: No buffer operation 0 1: Single buffer operation (GTCCRA ⇔ GTCCRC) 1 x: Double buffer operation (GTCCRA ⇔ GTCCRC ⇔ GTCCRD).	R/W
b19, b18	CCRB[1:0]	GTCCRB Buffer Operation	b19 b18 0 0: No buffer operation 0 1: Single buffer operation (GTCCRB ⇔ GTCCRE) 1 x: Double buffer operation (GTCCRB ⇔ GTCCRE ⇔ GTCCRF).	R/W
b21, b20	PR[1:0]	GTPR Buffer Operation	b21 b20 0 0: No buffer operation 0 1: Single buffer operation (GTPBR ⇔ GTPR) 1 x: Double buffer operation (GTPDBR ⇔ GTPBR ⇔ GTPR).	R/W
b22	CCRSWT	GTCCRA and GTCCRB Forcible Buffer Operation	Writing 1 to this bit forces a buffer transfer of GTCCRA and GTCCRB. This bit automatically returns to 0 after 1 is written. This bit is read as 0.	R/W
b23	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b25, b24	ADTTA[1:0]	GTADTRA Buffer Transfer Timing Select	• Triangle waves b25 b24 0 0: No transfer 0 1: Transfer at crest 1 0: Transfer at trough 1 1: Transfer at both crest and trough. • Saw waves b25 b24 0 0: No transfer Values other than 0 0: Transfer is triggered when the counter underflows (in down-counting), overflows (in up-counting), or is cleared.	R/W
b26	ADTDA	GTADTRA Double Buffer Operation	0: Single buffer operation (GTADTBRA ⇔ GTADTRA) 1: Double buffer operation (GTADTDBRA ⇔ GTADTBRA ⇔ GTADTDRA).	R/W
b27	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Bit	Symbol	Bit name	Description	R/W
b29, b28	ADTTB[1:0]	GTADTRB Buffer Transfer Timing Select	• Triangle waves b29 b28 0 0: No transfer 0 1: Transfer at crest 1 0: Transfer at trough 1 1: Transfer at both crest and trough. • Saw waves b29 b28 0 0: No transfer Values other than 0 0: Transfer is triggered when the counter underflows (in down-counting), overflows (in up-counting), or is cleared.	R/W
b30	ADTDB	GTADTRB Double Buffer Operation	0: Single buffer operation (GTADTBRB ⇔ GTADTRB) 1: Double buffer operation (GTADTDBRB ⇔ GTADTBRB ⇔ GTADTRB).	R/W
b31	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

GTBER provides settings for the buffer operation and must be set while the GTCNT operation stops.

BD[0] bit (GTCCR Buffer Operation Disable)

The BD[0] bit disables buffer operation using GTCCRA, GTCCRB, GTCCRC, GTCCRD, GTCCRE, and GTCCRF of the combined GPT.
When GTDTCR.TDE is 1 and when BD[0] is set to 0, GTCCRB does not perform buffer operation and the GTCCRB register is automatically set to a compare match value for a negative-phase waveform with dead time.

BD[1] bit (GTPR Buffer Operation Disable)

The BD[1] bit disables buffer operation using GTPR, GTPBR, and GTPDBR of the combined GPT.

BD[2] bit (GTADTR Buffer Operation Disable)

The BD[2] bit disables buffer operation using GTADTRA, GTADTRB, GTADTBRA, GTADTBRB, GTADTDBRA, and GTADTDBRB of the combined GPT. In event count operation, this bit is not available and the GTADTR buffer operation is not performed.

BD[3] bit (GTDV Buffer Operation Disable)

The BD[3] bit disables buffer operation using GTDVU, GTDVD, GTDBD, and GTDBU of the combined GPT.
When the GTDTCR.TDFER bit is set to 1 and when BD[3] is set to 0, buffer operation is not performed and the GTDVD value is set as a value of GTDVU automatically. In event count operation, this bit is not available and the GTDV buffer operation is not performed.

CCRA[1:0] bits (GTCCRA Buffer Operation)

The CCRA[1:0] bits set buffer operation using GTCCRA, GTCCRC, and GTCCRD of the combined GPT. When buffer operation is restricted by the operating mode set in GTCR, the GTCR setting is given priority.*1

CCRB[1:0] bits (GTCCRB Buffer Operation)

The CCRB[1:0] bits set buffer operation using GTCCRB, GTCCRE, and GTCCRF of the combined GPT. When buffer operation is restricted by the operating mode set in GTCR, the GTCR setting is given priority.*1

PR[1:0] bits (GTPR Buffer Operation)

The PR[1:0] bits set buffer operation using GTPR, GTPBR, and GTPDBR of the combined GPT.

CCRSWT bit (GTCCRA and GTCCRB Forcible Buffer Operation)

Writing 1 to the CCRSWT bit forcibly performs a buffer transfer of GTCCRA and GTCCRB. This bit automatically returns to 0 after 1 is written. This bit is read as 0 and is only valid when counting is stopped with a specified compare match operation.

ADTTA[1:0] bits (GTADTRA Buffer Transfer Timing Select)

The ADTTA[1:0] bits set the transfer timing for buffer operation of GTADTRA, GTADTBRA, and GTADTDBRA. These bits are not available in event count operation.

ADTDA bit (GTADTRA Double Buffer Operation)

The ADTDA bit sets buffer operation with GTADTRA, GTADTBRA, and GTADTDBRA combined. This bit is not available in event count operation.

ADTTB[1:0] bits (GTADTRB Buffer Transfer Timing Select)

The ADTTB[1:0] bits set the transfer timing for buffer operation of GTADTRB, GTADTBRB, and GTADTDBRB. This bit is not available in event count operation.

ADTDB bit (GTADTRB Double Buffer Operation)

The ADTDB bit sets buffer operation using GTADTRB, GTADTBRB, and GTADTDBRB of the combined GPT. This bit is not available in event count operation.

Note 1. The buffer operation mode is fixed in saw-wave one-shot pulse mode, or triangle-wave PWM mode 3 (64-bit transfer at trough).

12.2.18 General PWM Timer Interrupt and A/D Converter Start Request Skipping Setting Register (GTITC)

Address(es): GPT32Em.GTITC E804 3044h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	ADTBL	—	ADTAL	—	IVTT[2:0]			IVTC[1:0]		ITLF	ITLE	ITLD	ITLC	ITLB	ITLA
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	ITLA	GTCCRA Compare Match/Input Capture Interrupt Link	0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.	R/W
b1	ITLB	GTCCRB Compare Match/Input Capture Interrupt Link	0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.	R/W
b2	ITLC	GTCCRC Compare Match Interrupt Link	0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.	R/W
b3	ITLD	GTCCRD Compare Match Interrupt Link	0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.	R/W
b4	ITLE	GTCCRE Compare Match Interrupt Link	0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.	R/W
b5	ITLF	GTCCRF Compare Match Interrupt Link	0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.	R/W
b7, b6	IVTC[1:0]	OVFn/UNFn Interrupt Skipping Function Select	b7 b6 0 0: Do not perform skipping 0 1: Count and skip both overflow and underflow for saw waves and crest for triangle waves 1 0: Count and skip both overflow and underflow for saw waves and trough for triangle waves 1 1: Count and skip both overflow and underflow for saw waves and both crest and trough for triangle waves.	R/W
b10 to b8	IVTT[2:0]	OVFn/UNFn Interrupt Skipping Count Select	b10 b8 0 0 0: No skipping 0 0 1: Skipping count of 1 0 1 0: Skipping count of 2 0 1 1: Skipping count of 3 1 0 0: Skipping count of 4 1 0 1: Skipping count of 5 1 1 0: Skipping count of 6 1 1 1: Skipping count of 7.	R/W
b11	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b12	ADTAL	GTADTRA A/D Converter Start Request Link	0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.	R/W
b13	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b14	ADTBL	GTADTRB A/D Converter Start Request Link	0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.	R/W
b31 to b15	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

GTITC sets the skipping function for the GTCNT counter overflow (GTPR compare match) interrupt (OVFn) and underflow interrupt (UNFn). It also specifies whether to link other interrupts and A/D converter start requests with the OVFn/UNFn interrupt skipping function. An output disable request to the POEG cannot be linked with the OVFn/UNFn interrupt skipping function. This bit is not available in event count operation.

ITLA bit (GTCCRA Compare Match/Input Capture Interrupt Link)

The ITLA bit specifies whether to link the GTCCRA compare match/input capture interrupt (GTCIA) with the OVFn/UNFn interrupt skipping function.

ITLB bit (GTCCRB Compare Match/Input Capture Interrupt Link)

The ITLB bit specifies whether to link the GTCCRB compare match/input capture interrupt (GTCIB) with the OVFn/UNFn interrupt skipping function.

ITLC bit (GTCCRC Compare Match Interrupt Link)

The ITLC bit specifies whether to link the GTCCRC compare match interrupt (GTCIC) with the OVFn/UNFn interrupt skipping function.

ITLD bit (GTCCRD Compare Match Interrupt Link)

The ITLD bit specifies whether to link the GTCCRD compare match interrupt (GTCID) with the OVFn/UNFn interrupt skipping function.

ITLE bit (GTCCRE Compare Match Interrupt Link)

The ITLE bit specifies whether to link the GTCCRE compare match interrupt (GTCIE) with the OVFn/UNFn interrupt skipping function.

ITLF bit (GTCCRF Compare Match Interrupt Link)

The ITLF bit specifies whether to link the GTCCRF compare match interrupt (GTCIF) with the OVFn/UNFn interrupt skipping function.

IVTC[1:0] bits (OVFn/UNFn Interrupt Skipping Function Select)

The IVTC[1:0] bits set the skipping function for the GTPR compare match (GTCNT overflow) interrupt (OVFn) and GTCNT counter underflow interrupt (UNFn).

IVTT[2:0] bits (OVFn/UNFn Interrupt Skipping Count Select)

The IVTT[2:0] bits set the skipping count for the GTPR compare match (GTCNT overflow) interrupt (OVFn) and GTCNT counter underflow interrupt (UNFn). When modifying the IVTT[2:0] bits, first set the IVTC[1:0] bits to 00b.

ADTAL bit (GTADTRA A/D Converter Start Request Link)

The ADTAL bit specifies whether to link the GTADTRA A/D converter start request with OVFn/UNFn interrupt skipping function.

ADTBL bit (GTADTRB A/D Converter Start Request Link)

The ADTBL bit specifies whether to link the GTADTRB A/D converter start request with OVFn/UNFn interrupt skipping function.

12.2.19 General PWM Timer Counter (GTCNT)

Address(es): GPT32Em.GTCNT E804 3048h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GTCNT is a 32-bit read/write counter and can only be written to after counting stops. GTCNT must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited. GTCNT must be set within the range of 0 ≤ GTCNT ≤ GTPR.

12.2.20 General PWM Timer Compare Capture Register n (GTCCRn) (n = A to F)

Address(es): GPT32Em.GTCCRA E804 304Ch + 0100h × m (m = 0 to 7)
GPT32Em.GTCCRB E804 3050h + 0100h × m (m = 0 to 7)
GPT32Em.GTCCRC E804 3054h + 0100h × m (m = 0 to 7)
GPT32Em.GTCCRE E804 3058h + 0100h × m (m = 0 to 7)
GPT32Em.GTCCRD E804 305Ch + 0100h × m (m = 0 to 7)
GPT32Em.GTCCRF E804 3060h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

GTCCRn registers are read/write registers.
GTCCRA and GTCCRB are registers used for both output compare and input capture.
GTCCRC and GTCCRE are compare match registers that can also function as buffer registers for GTCCRA and GTCCRB.
GTCCRD and GTCCRF are compare match registers that can also function as buffer registers for GTCCRC and GTCCRE (double-buffer registers for GTCCRA and GTCCRB).

12.2.21 General PWM Timer Cycle Setting Register (GTPR)

Address(es): GPT32Em.GTPR E804 3064h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

GTPR is a read/write register that sets the maximum count value of GTCNT. For saw waves, the value of (GTPR + 1) is the cycle. For triangle waves, the value of (GTPR value × 2) is the cycle.

12.2.22 General PWM Timer Cycle Setting Buffer Register (GTPBR)

Address(es): GPT32Em.GTPBR E804 3068h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

GTPBR is a read/write register that functions as a buffer register for GTPR.

12.2.23 General PWM Timer Cycle Setting Double-Buffer Register (GTPDBR)

Address(es): GPT32Em.GTPDBR E804 306Ch + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

GTPDBR is a 32-bit read/write register that functions as a buffer register for GTPBR (double-buffer register for GTPR).

12.2.24 A/D Converter Start Request Timing Register n (GTADTRn) (n = A, B)

Address(es): GPT32Em.GTADTRA E804 3070h + 0100h × m (m = 0 to 7)
GPT32Em.GTADTRB E804 307Ch + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

GTADTRn registers are 32-bit read/write registers that set the timing of A/D converter start request generation. When the GTADTRn value matches the GTCNT counter value, an A/D converter start request is generated. GTADTRn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited.

12.2.25 A/D Converter Start Request Timing Buffer Register n (GTADTBRn) (n = A, B)

Address(es): GPT32Em.GTADTBRA E804 3074h + 0100h × m (m = 0 to 7)
GPT32Em.GTADTBRB E804 3080h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

GTADTBRn registers are 32-bit read/write registers that function as buffer registers for GTADTRn. GTADTBRn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited.

12.2.26 A/D Converter Start Request Timing Double-Buffer Register n (GTADTDBRn) (n = A, B)

Address(es): GPT32Em.GTADTDBRA E804 3078h + 0100h × m (m = 0 to 7)
GPT32Em.GTADTDBRB E804 3084h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

GTADTDBRn registers are 32-bit read/write registers that function as buffer registers for GTADTBRn (double-buffer registers for GTADTR). GTADTDBRn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited.

12.2.27 General PWM Timer Dead Time Control Register (GTDTCR)

Address(es): GPT32Em.GTDTCR E804 3088h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TDFER	—	—	TDBDE	TDBUE	—	—	—	TDE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	TDE	Negative-Phase Waveform Setting	0: Set GTCCRB without using GTDVU and GTDVD. 1: Use GTDVU and GTDVD to set the compare match value for negative-phase waveform with automatic dead time in GTCCRB.	R/W
b3 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b4	TDBUE	GTDVU Buffer Operation Enable	0: Disable GTDVU buffer operation 1: Enable GTDVU buffer operation.	R/W
b5	TDBDE	GTDVD Buffer Operation Enable	0: Disable GTDVD buffer operation 1: Enable GTDVD buffer operation.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	TDFER	GTDVD Setting	0: Set GTDVU and GTDVD separately 1: Automatically set the value written to GTDVU to GTDVD.	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

GTDTCR enables automatic setting of a compare match value for negative-phase waveform with dead time.

TDE bit (Negative-Phase Waveform Setting)

The TDE bit specifies whether to use GTDVU and GTDVD. When GTDVU and GTDVD are used, the compare match value for a negative-phase waveform with dead time obtained by the compare match value of a positive-phase waveform (GTCCRA) and the dead time value (GTDVU and GTDVD), is automatically set in GTCCRB.
The TDE bit setting is ignored in saw-wave PWM mode, and automatic setting does not take place.
The GTCCRB value is automatically set and has the following upper and lower limit values. If the obtained GTCCRB value is not within the upper or lower limit, the following limit value is set in GTCCRB and the GTST.DTEF flag is set to 1. However, in triangle waves, when the obtained GTCCRB value exceeds the upper limit value, the GTST.DTEF flag is set to 0.

- Triangle waves
  Upper limit value: GTPR – 1
  Lower limit value: 1 in up-counting, 0 in down-counting
- Saw-wave one-shot pulse mode
  Upper limit value: GTPR
  Lower limit value: 0

TDBUE bit (GTDVU Buffer Operation Enable)

The TDBUE bit enables buffer operation with GTDVU and GTDBU combined. The buffer transfer timing is the trough for triangle waves, and an overflow or underflow for saw waves.

TDBDE bit (GTDVD Buffer Operation Enable)

The TDBDE bit enables buffer operation with GTDVD and GTDBD combined. The buffer transfer timing is the trough for triangle waves, and an overflow or underflow for saw waves. When this bit and the TDFER bit are set to 1 simultaneously, the TDFER bit setting is given priority.

TDFER bit (GTDVD Setting)

The TDFER bits sets whether or not the value written to GTDVU is also set to GTDVD automatically.

12.2.28 General PWM Timer Dead Time Value Register n (GTDVn) (n = U, D)

Address(es): GPT32Em.GTDVU E804 308Ch + 0100h x m (m = 0 to 7)
GPT32Em.GTDVD E804 3090h + 0100h x m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

GTDVn is a 32-bit read/write register that sets the dead time for generating PWM waveforms with dead time. GTDVU is used for up-counting and GTDVD is used for down-counting.
It is prohibited to set the GTDVn register with a value larger than or equal to that of the GTPR register.
Do not set the change point of waveform beyond the count cycle period, when the automatic dead time setting function is used. By reading the GTCCRB register, can be seen the change point for the reverse-phase waveform after the dead time added, which is set in the automatic dead time setting function.
When GTDVn is used, writing to GTCCRB is not allowed. When this register is set to 0, waveforms without dead time are output. GTDVn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited.
When GTDVn buffer operation is enabled, GTDBn can be written anytime. The value of GTDBn is transferred to GTDVn at the end of the count cycle period. When the GTDVn buffer operation is disabled, set the CST bit in the GTCR register to stop the GPT with before changing GTDVn to a new value. When GTDVn buffer operation is disabled, to change GTDVn to a new value, stop the GPT with the CST bit in the GTCR register.

12.2.29 General PWM Timer Dead Time Buffer Register n (GTDBn) (n = U, D)

Address(es): GPT32Em.GTDBU E804 3094h + 0100h × m (m = 0 to 7)
GPT32Em.GTDBD E804 3098h + 0100h × m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

GTDBn is a 32-bit read/write register that functions as a buffer register for GTDVn.

12.2.30 General PWM Timer Output Protection Function Status Register (GTSOS)

Address(es): GPT32Em.GTSOS E804 309Ch + 0100h x m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b1, b0	SOS[1:0]	Output Protection Function Status	b1 b0 0 0: Normal operation 0 1: Protected state (set GTCCRA = 0 during transfer at trough or crest) 1 0: Protected state (set GTCCRA ≥ GTPR during transfer at trough) 1 1: Protected state (set GTCCRA ≥ GTPR during transfer at crest).	R
b7 to b2	—	Reserved	These bits are read as 0.	R
b9, b8	—	Reserved	The read value is undefined.	R
b31 to b10	—	Reserved	These bits are read as 0.	R

GTSOS is a status register that indicates the status of the output protection function. The output protection function is enabled only when the dead time is automatically set (GTDTCR.TDE bit = 1) in triangle-wave mode.

SOS[1:0] bits (Output Protection Function Status)

The SOS[1:0] bits indicate the status of the output protection function in triangle-wave PWM mode.

12.2.31 General PWM Timer Output Protection Function Temporary Release Register (GTSOTR)

Address(es): GPT32Em.GTSOTR E804 30A0h + 0100h x m (m = 0 to 7)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOTR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	SOTR	Output Protection Function Temporary Release	0: Do not release protected state 1: Release protected state.	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

GTSOTR temporarily releases the protected state of GTIOCB pin output when output protection is set. The protected state can be released only when GTSOS.SOS[1:0] bits = 10b (protected state in which GTCCRA ≥ GTPR has occurred during transfer at trough). The protected state cannot be released in any other case.

SOTR bit (Output Protection Function Temporary Release)

The SOTR bit specifies whether to temporarily release the protected state of the GTIOCB pin output in an output protected state. When the SOTR bit is set to 1, the output protection function is canceled from the first trough. When the SOTR bit is set to 0, output protection resumes from the first trough.

12.2.32 General PWM Timer Event Control Register (GTECR)

Address(es): GPT.GTECR E804 3800h

	b7	b6	b5	b4	b3	b2	b1	b0
	EVCON	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b6 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	EVCON	All Event cooperation function Enable	0: Disable Event cooperation function 1: Enable Event cooperation function	R/W

The GTECR register controls the event cooperation function.

12.2.33 General PWM Timer Event Source Setting Register (GTESRn) (n=0 to 7)

Address(es): GPT.GTESR0 H'E804_3810, GPT.GTESR1 H'E804_3814, GPT.GTESR2 H'E804_3818, GPT.GTESR3 H'E804_381C
GPT.GTESR4 H'E804_3820, GPT.GTESR5 H'E804_3824, GPT.GTESR6 H'E804_3828, GPT.GTESR7 H'E804_382C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	EVS[6:0]						
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b6 to b0	EVS[6:0]	Event Source Select	0000000: Disable output of the associated event signal 0000001 to 1000000b: Number setting for the event signal to be selected Other settings are prohibited.	R/W
b15 to b7	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The GTESRn register specifies an event signal to be selected for each event input.
Table 12.6 shows the associations between the GTESRn registers and the event input name.

Table 12.6 Associations between the GTESRn registers and Event input name

Register name	Event name
GTESR0	Event Input A
GTESR1	Event Input B
GTESR2	Event Input C
GTESR3	Event Input D
GTESR4	Event Input E
GTESR5	Event Input F
GTESR6	Event Input G
GTESR7	Event Input H

12.3 Operation

12.3.1 Basic Operation

Each channel has a 32-bit timer that performs a periodic count operation using the count clock and hardware sources. The count function provides both up-counting and down-counting. The GTPR controls the count cycle. When the GTCNT counter value matches the value in GTCCRA or GTCCRB, the output from the associated pin GTIOCA or GTIOCB can be changed. GTCCRA or GTCCRB can be used as an input capture register with hardware resources. GTCCRC and GTCCRD can function as buffer registers for GTCCRA. GTCCRE and GTCCRF can function as buffer registers for GTCCRB.

12.3.1.1 Counter Operation

(1) Counter start and stop

The counter of each channel starts the count operation by setting GTCR.CST to 1. The GTCR.CST bit value is changed by following sources.

- Writing to GTCR register
- Writing 1 to the bit in GTSTR associated with the GPT channel number when the GTSSR.CSTRT bit set to 1
- Writing 1 to the bit in GTSTP associated with the GPT channel number when the GTPSR.CSTOP bit set to 1
- The hardware source selected in the GTSSR register
- The hardware source selected in the GTPSR register.

(2) Periodic count operation in up-counting by count clock

The GTCNT counter in each channel starts up-counting when the associated GTCR.CST bit is set to 1 with GTUPSR and GTDNSR registers set to 00000000h. When the GTCNT value changes from the GTPR value to 0 (overflow), the GTST.TCFPO flag is set to 1. When GTCNT overflows, up-counting resumes from 00000000h.
Figure 12.2 shows an example of a periodic count operation in up-counting.

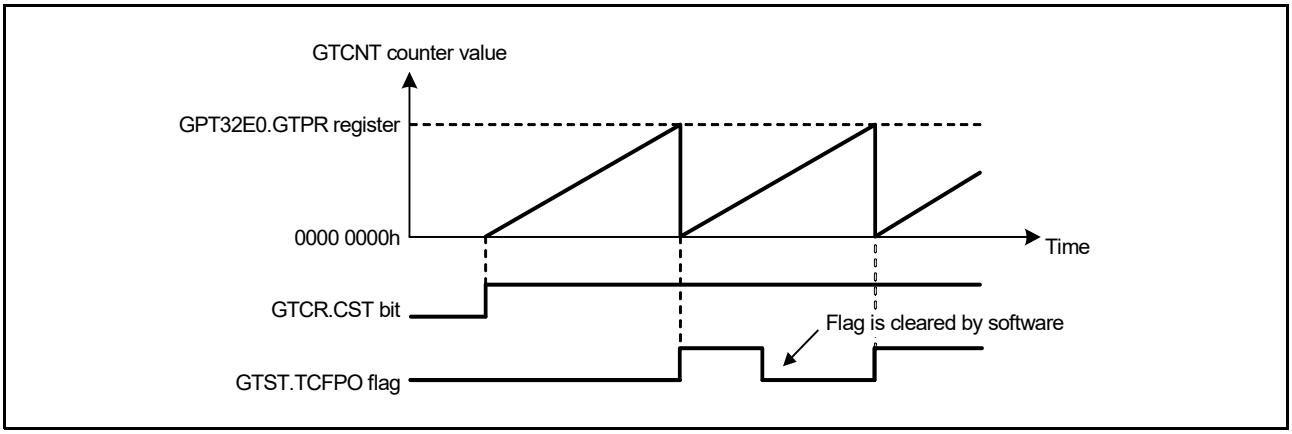


Figure 12.2 Example of periodic count operation in up-counting by the count clock

Figure 12.3 shows an example setting for periodic count operation in up-counting.

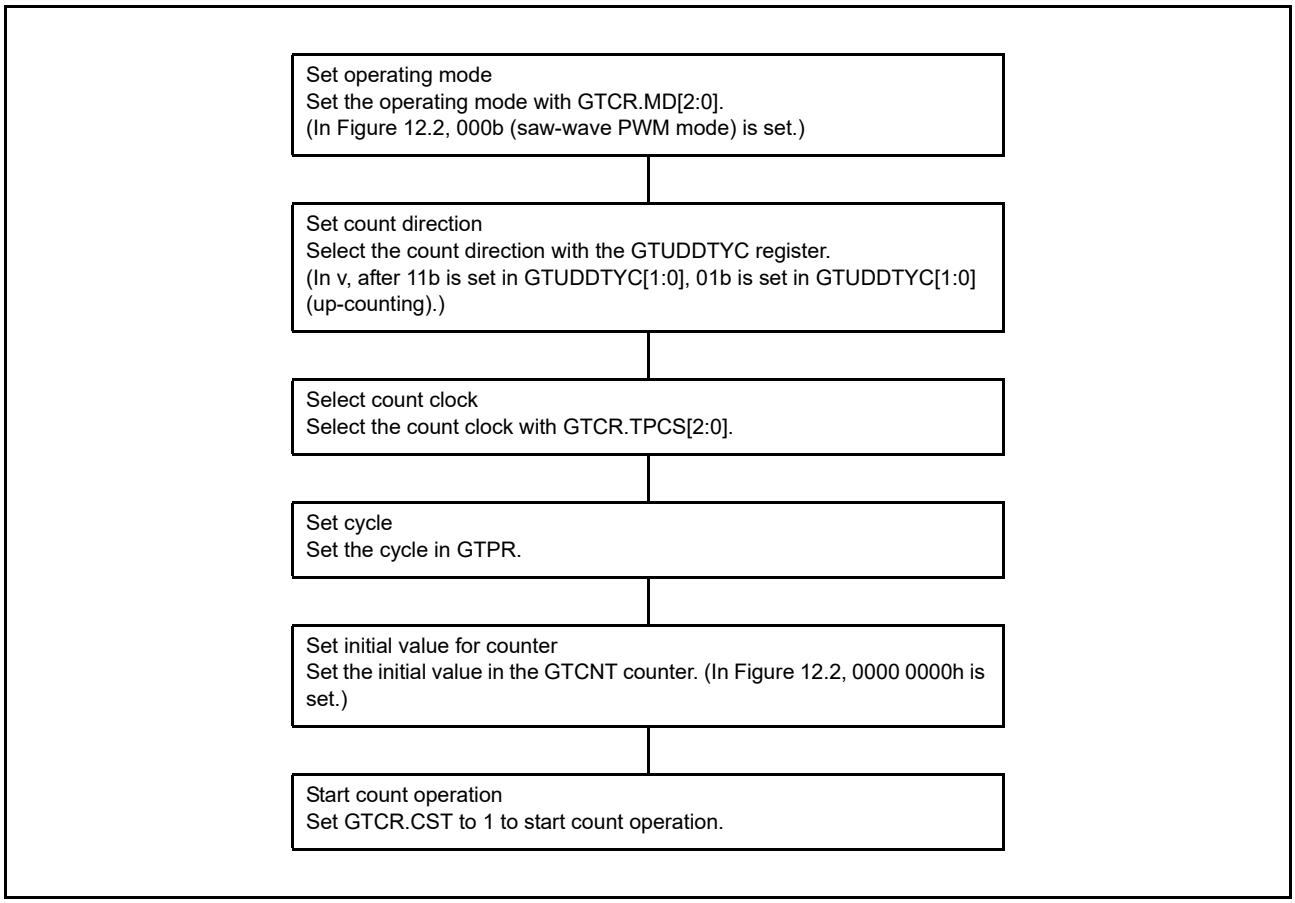


Figure 12.3 Example setting for a periodic count operation in up-counting by the count clock

(3) Periodic count operation in down-counting by count clock

The GTCNT counter in each channel can perform down-counting by setting GTUDDTYC.UD with GTUPSR and GTDNSR registers set to 00000000h. When GTCNT changes from 0 to the GTPR value (underflow), GTST.TCFPU is set to 1. When the GTCNT counter underflows, down-counting resumes from the GTPR value.
Figure 12.4 shows an example of periodic count operation in down-counting by the count clock.

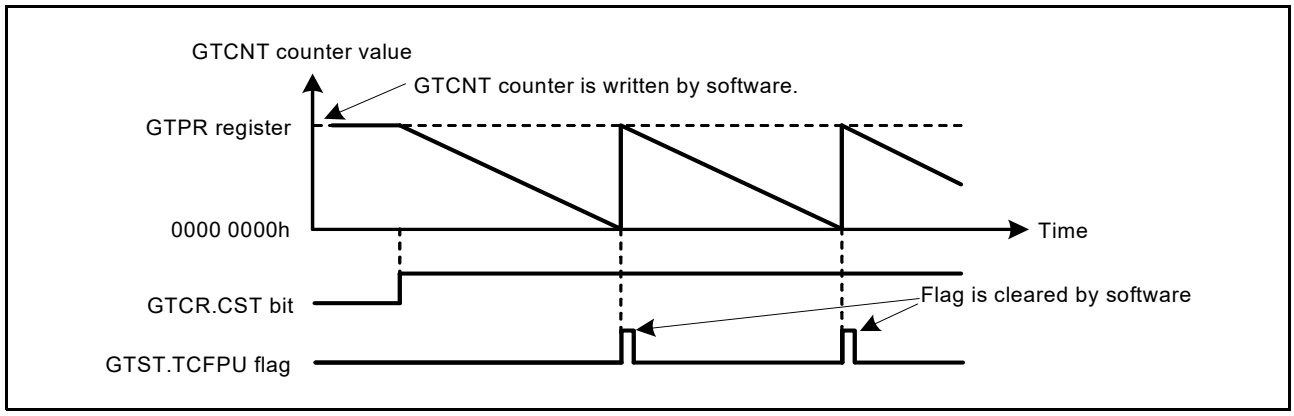


Figure 12.4 Example of periodic count operation in down-counting by the count clock

Figure 12.5 shows an example setting for periodic count operation in down-counting by the count clock.

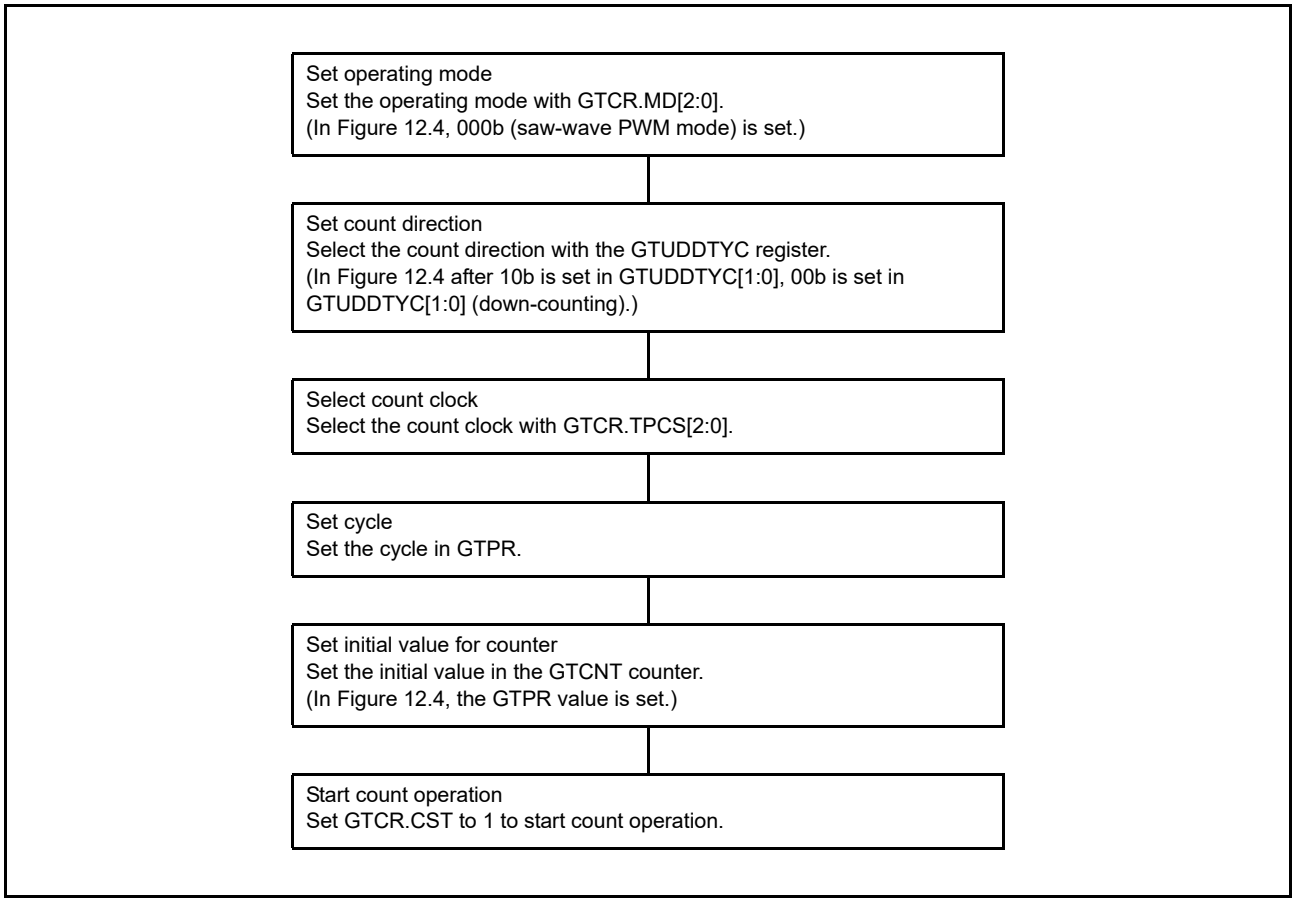


Figure 12.5 Example setting for periodic count operation in down-counting by count clock

(4) Event count operation in up-counting using hardware sources

The GTCNT counter in each channel can perform up-counting using hardware sources as set in GTUPSR.
When GTUPSR is set to enable, the count clock selected in GTCR.TPCS[2:0] and the count direction selected in GTUDDTYC.UD are ignored. If up-counting and down-counting using hardware sources occur at the same time, the GTCNT counter value does not change. The overflow behavior for up-counting using hardware sources is the same as for up-counting by the count clock.
When GTCR.CST bit is set to 1 to count up using hardware sources, the count operation is enabled. After GTCR.CST is set to 1, the counter cannot count up for 1 clock cycle as specified in GTCR.TPCS[2:0] because the count operation is synchronized by the count clock selected by GTCR.TPCS[2:0]. Set the GTCR.TPCS[2:0] bits to 000b to select counting up of single cycles of the P1φ clock with a delay of one clock cycle after the GTCR.CST bit has been set to 1.
Figure 12.6 shows an example of a periodic count operation in up-counting by a hardware resource (rising edge of GTETRGA pin).

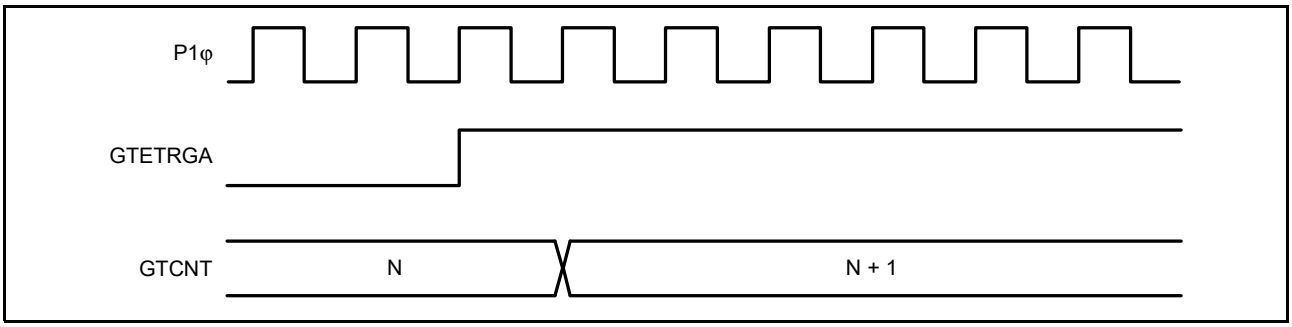


Figure 12.6 Example of periodic count operation in up-counting using hardware sources

Figure 12.7 shows an example setting for periodic count operation in up-counting by the count clock.

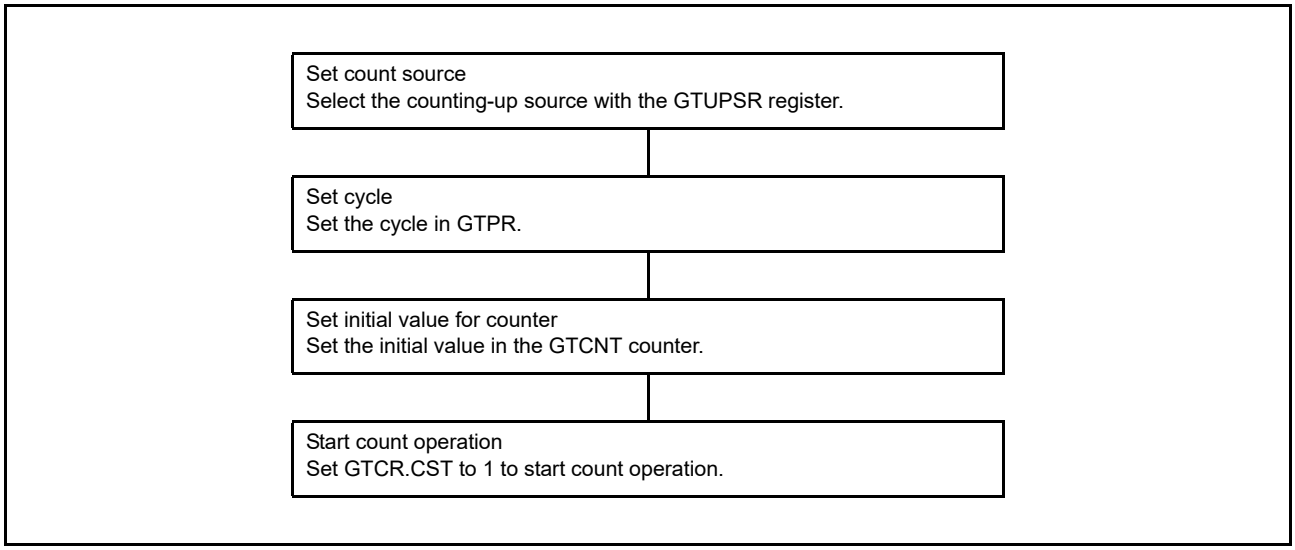


Figure 12.7 Example setting for an event count operation in up-counting using hardware sources

(5) Event count operation in down-counting using hardware sources

The GTCNT counter in each channel can perform down-counting using hardware sources set in the GTDNSR. When GTDNSR is set to enable, the count clock selected in GTCR.TPCS[2:0] and the count direction selected in GTUDDTYC.UD are ignored. If up-counting and down-counting using hardware sources occur at the same time, GTCNT counter value does not change. The underflow behavior for down-counting using hardware sources is the same as for down-counting by the count clock.

When GTCR.CST bit is set to 1 to count down using hardware sources, the count operation is enabled. After GTCR.CST is set to 1, the counter cannot count down for 1 clock cycle as specified in GTCR.TPCS[2:0] because the count operation is synchronized with the count clock selected by GTCR.TPCS[2:0]. Set the GTCR.TPCS[2:0] bits to 000b to select counting up of single cycles of the P1φ clock with a delay of one clock cycle after the GTCR.CST bit has been set to 1.

Figure 12.8 shows an example of a periodic count operation in down-counting by a hardware resource (rising edge of GTETRGA pin).

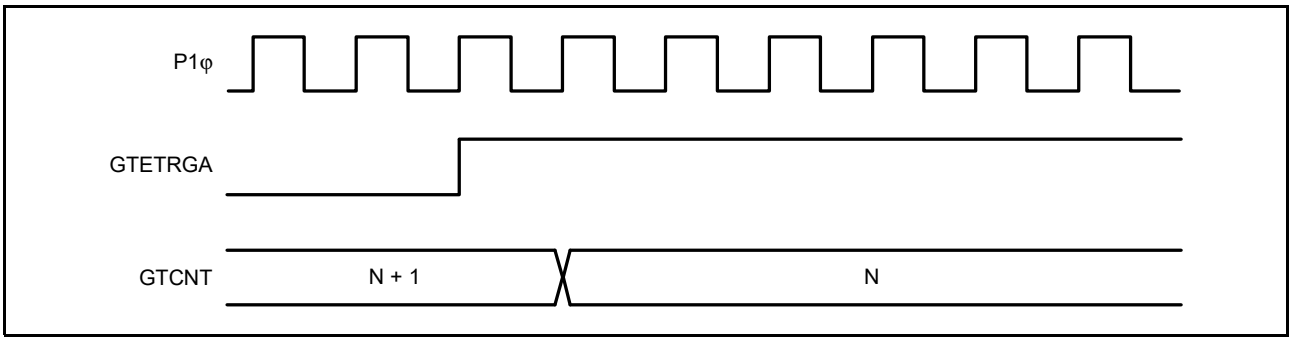


Figure 12.8 Example of event count operation in down-counting using hardware sources

Figure 12.9 shows an example setting for a periodic count operation in down-counting using a hardware resource.

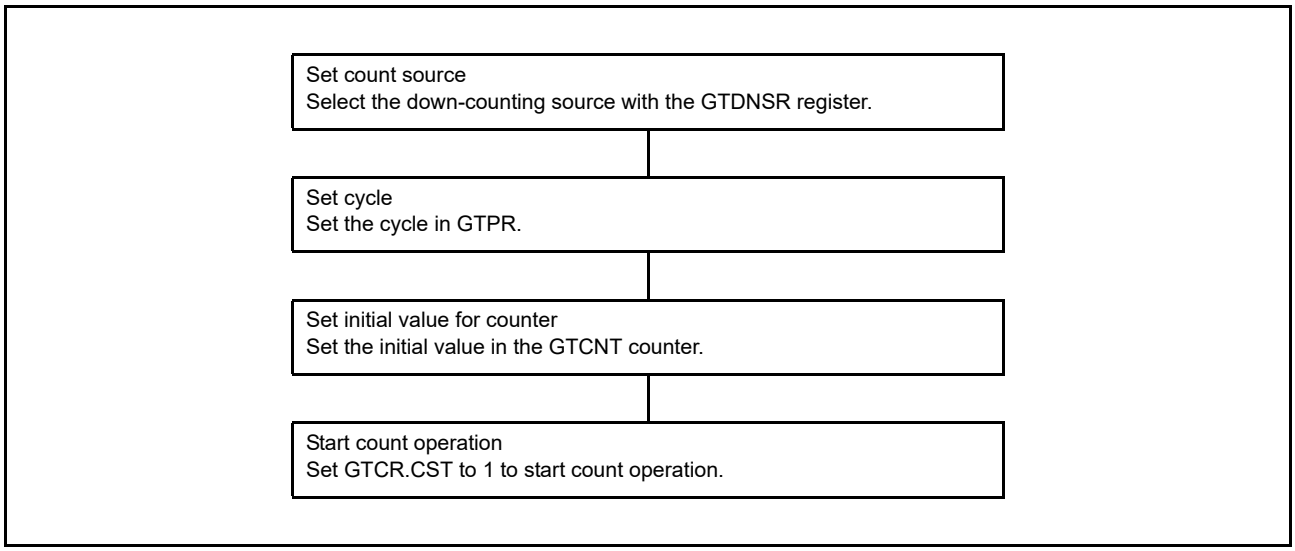


Figure 12.9 Example setting for an event count operation in down-counting using hardware sources

(6) Counter clear operation

The counter of each channel is cleared by either of the following sources.

- Writing 0 to GTCNT register
- Writing 1 to the bit in GTCLR associated with the GPT channel number when the GTCSR.CCLR bit set to 1
- The hardware source selected in GTCSR register.

Writing to the GTCNT register is prohibited during count operation. The GTCNT counter can be cleared both by writing 1 to the GTCLR and by the clear request of hardware sources, whether GTCNT is counting (GTCR.CST = 1) or not (GTCR.CST = 0).

For saw waves selected by setting GTCR.MD[2:0] and the count direction flag showing down-counting (GTST.TUCF = 0), the GTCNT register is set to the value of the GTPR register when the counter is cleared by writing 1 to the GTCLR register or by the hardware source specified in the GTCSR register. When not in saw waves mode and down-counting, the GTCNT register is set to 0 when writing 1 to the GTCLR register and when clearing by hardware sources are performed.

In event count operation when at least 1 bit in GTUPSR or GTDNSR is set to 1, after clear sources occur, both writing to GTCLR register and clearing by hardware sources are performed immediately to synchronize with P1φ. If other settings are used, clear is synchronized with the counter clock selected by GTCR.TPCS[2:0].

12.3.1.2 Waveform Output by Compare Match

Compare match means that the GTCNT counter value matches the value of GTCCRA or GTCCRB. When a compare match occurs, the compare match flag is generated synchronously with the count clock including the event count. At the same time the GPT can output low, high, or toggle output from the associated GTIOCA or GTIOCB output pin. In addition, the GTIOCA or GTIOCB pin output can be low, high, or toggle at the cycle end which is determined by GTPR. The cycle end is:

- For saw waves in up-counting – when GTCNT changes from the GTPR value to 0 (overflow)
- For saw waves in down-counting – when GTCNT changes from 0 to GTPR value (underflow)
- For saw waves – when the GTCNT counter is cleared
- For triangle waves – when the GTCNT changes from 0 to 1 (trough).

(1) Low and high output

Figure 12.10 shows an example of low and high output operation by a compare match of GTCCRA and GTCCRB. In this example, the GPT32E0.GTCNT counter performs up-counting, and settings are made so that high is output from the GTIOC0A pin by a GPT32E0.GTCCRA compare match, and low is output from the GTOC0B pin by a GPT32E0.GTCCRB compare match. The pin level does not change when the specified level and pin level match.

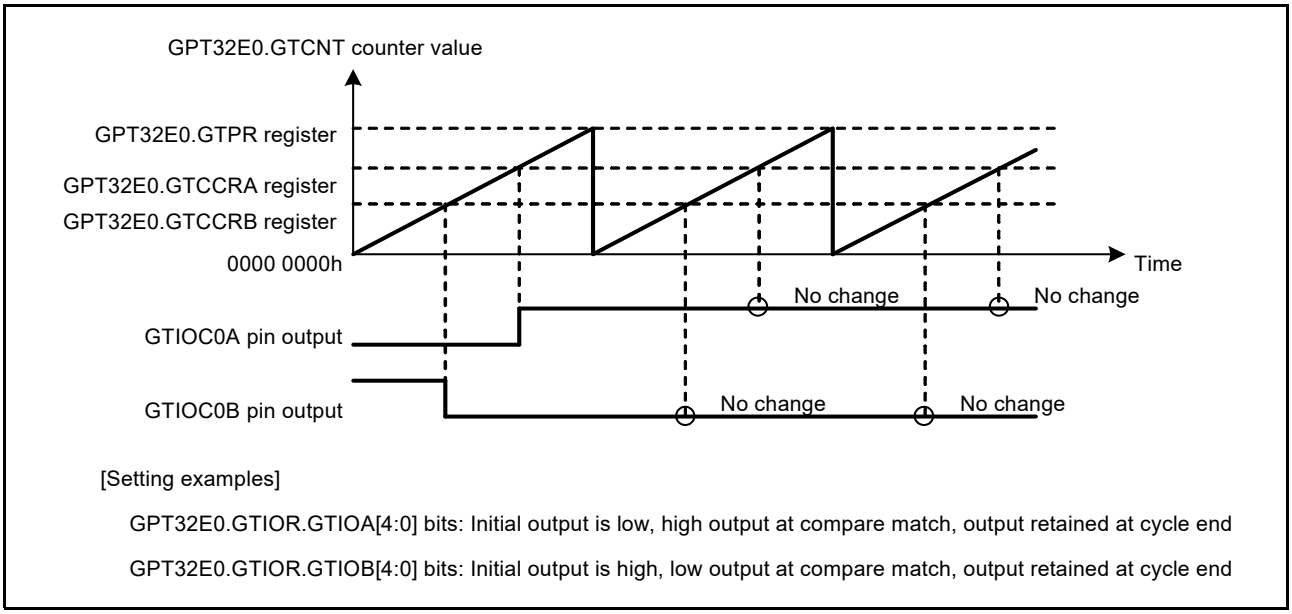


Figure 12.10 Example of low and high output operation

Figure 12.11 shows an example setting for low output and high output operation.

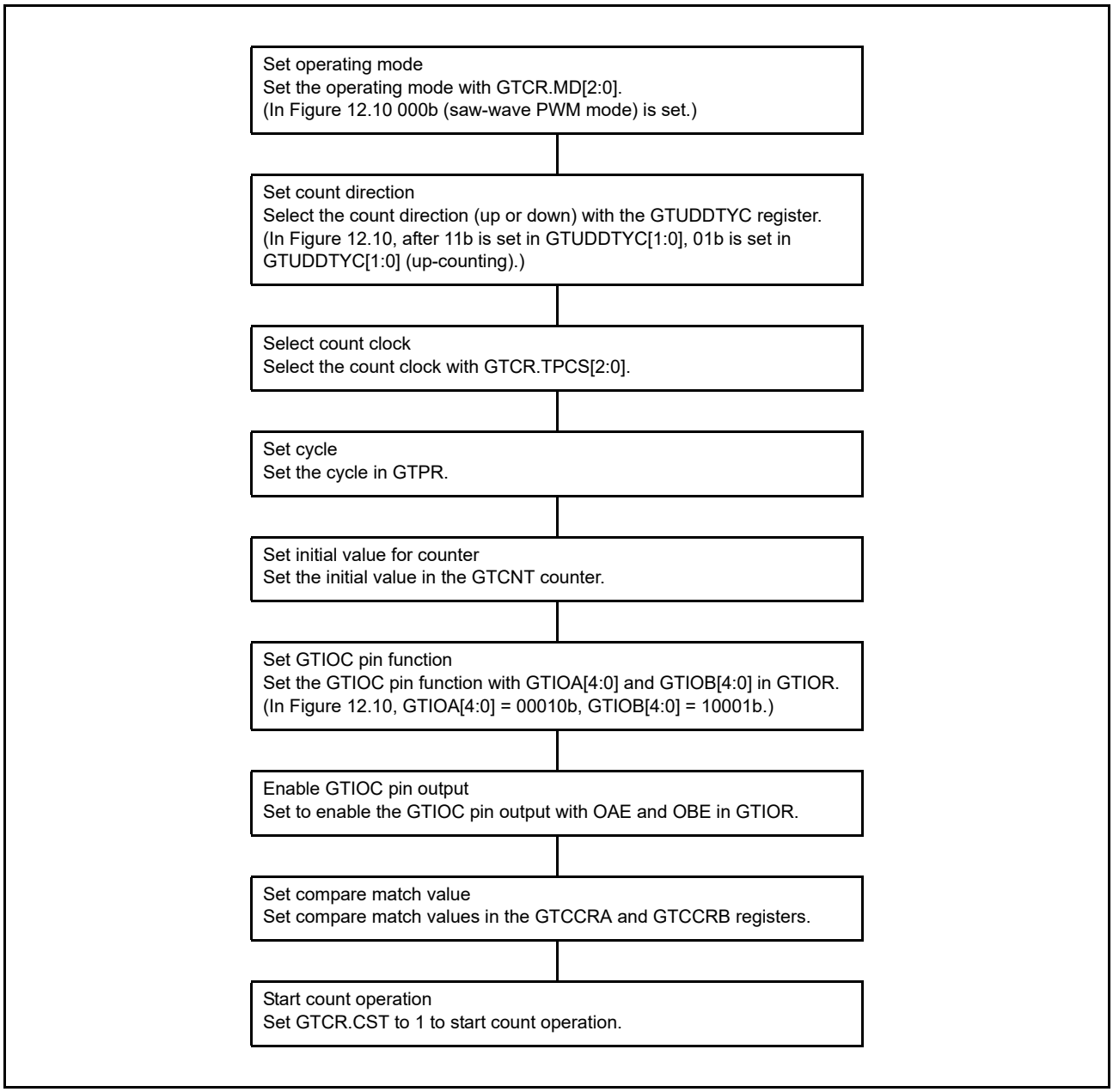


Figure 12.11 Example setting for low output and high output operation

(2) Toggled output

Figure 12.12 and Figure 12.13 show examples of toggled output operation by compare matches of GTCCRA and GTCCRB. In Figure 12.12, the GPT32E0.GTCNT counter performs up-counting, and settings are made so that the GTIOC0A pin output by a GPT32E0.GTCCRA compare match and GTIOC0B pin output by a GPT32E0.GTCCRB compare match are toggled.

Figure 12.13, the GPT32E0.GTCNT counter performs up-counting, and settings are made so that the GTIOC0A output is toggled by a compare match of GPT32E0.GTCCRA and the GTIOC0B output is toggled at the cycle end.

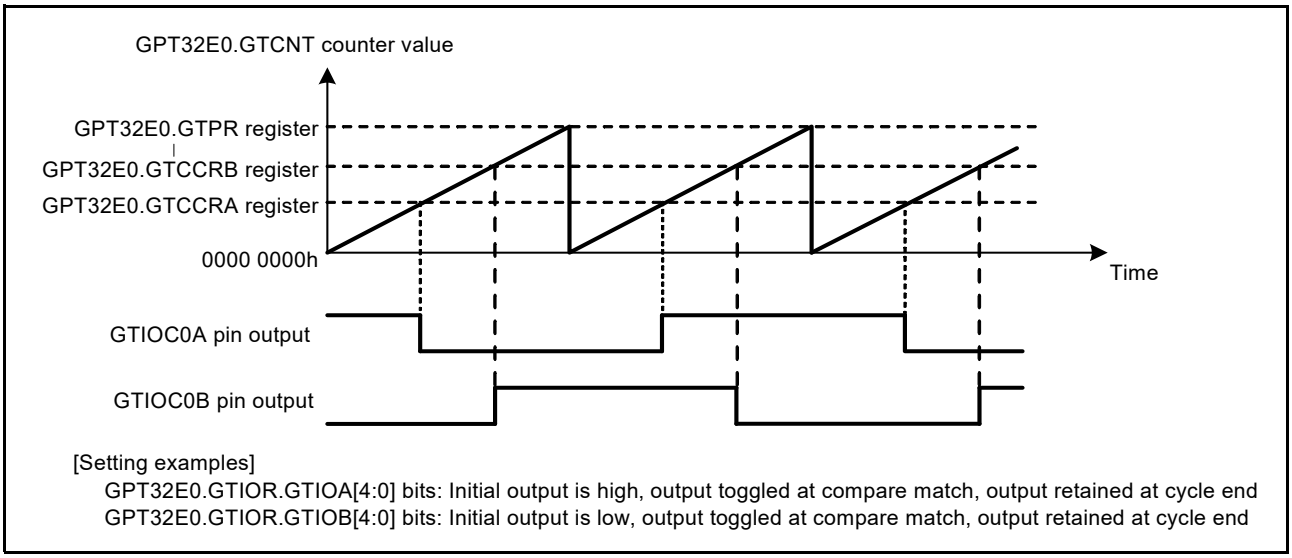


Figure 12.12 Example of toggled output operation (1)

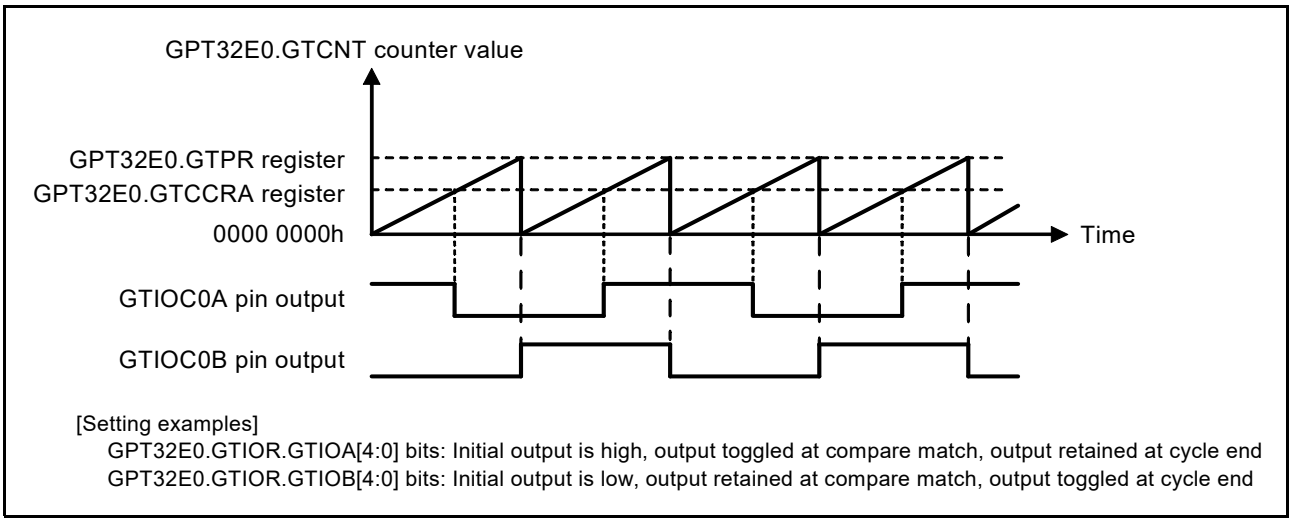


Figure 12.13 Example of toggled output operation (2)

Figure 12.14 shows an example setting for toggled output operation.

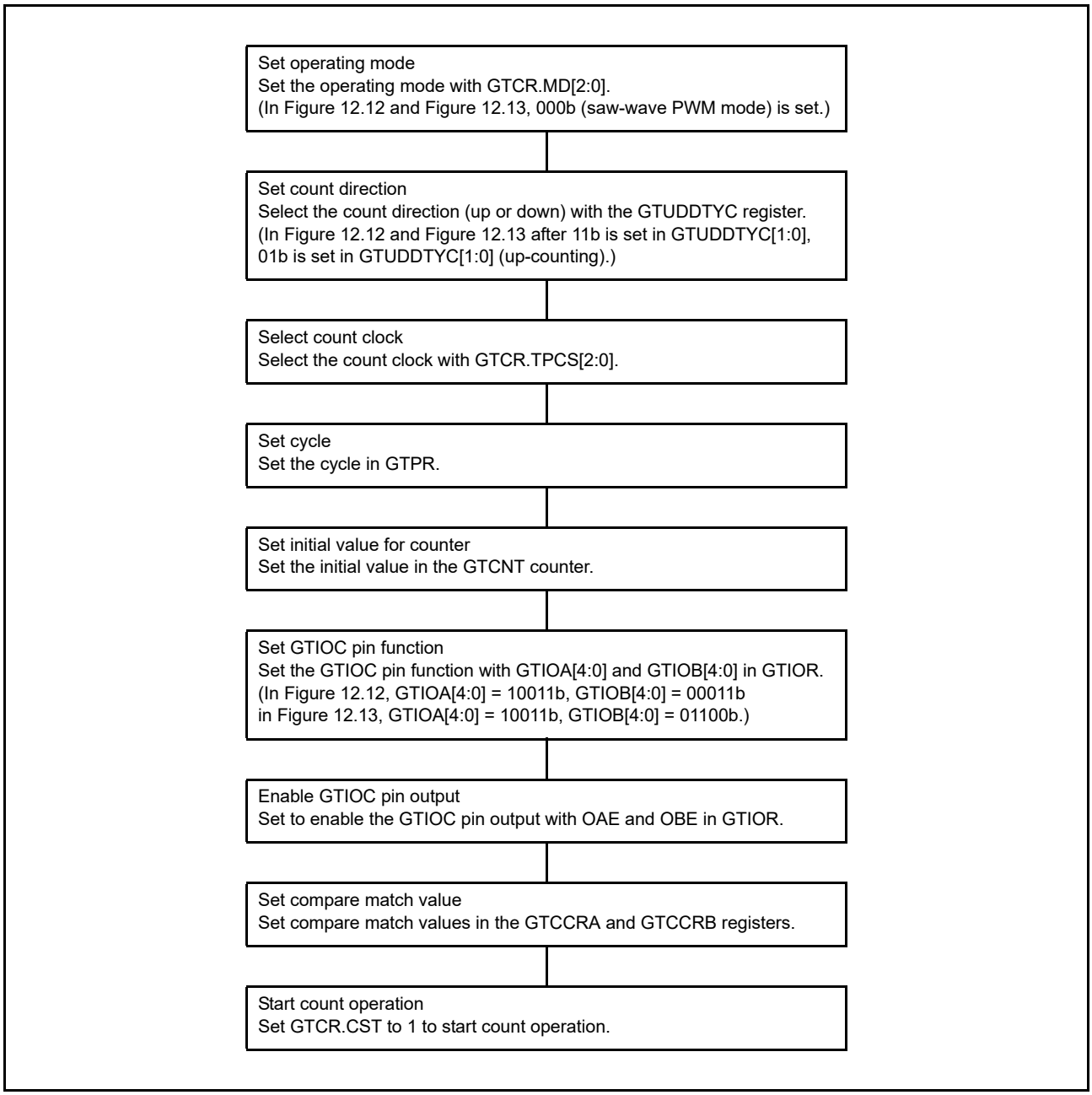


Figure 12.14 Example setting for toggled output operation

12.3.1.3 Input Capture Function

The GTCNT counter value can be transferred to either GTCCRA or GTCCRB on detection of the hardware source that is set in GTICASR and GTICBSR.

Figure 12.15 shows an example of the input capture function.

In this example, the GPT32E0.GTCNT counter performs up-counting by the count clock, and settings are made so that an input capture is performed to GTICCRA at both edges of the GTIOC0A input pin and to GTICCRB on the rising edge of the GTIOC0B input pin.

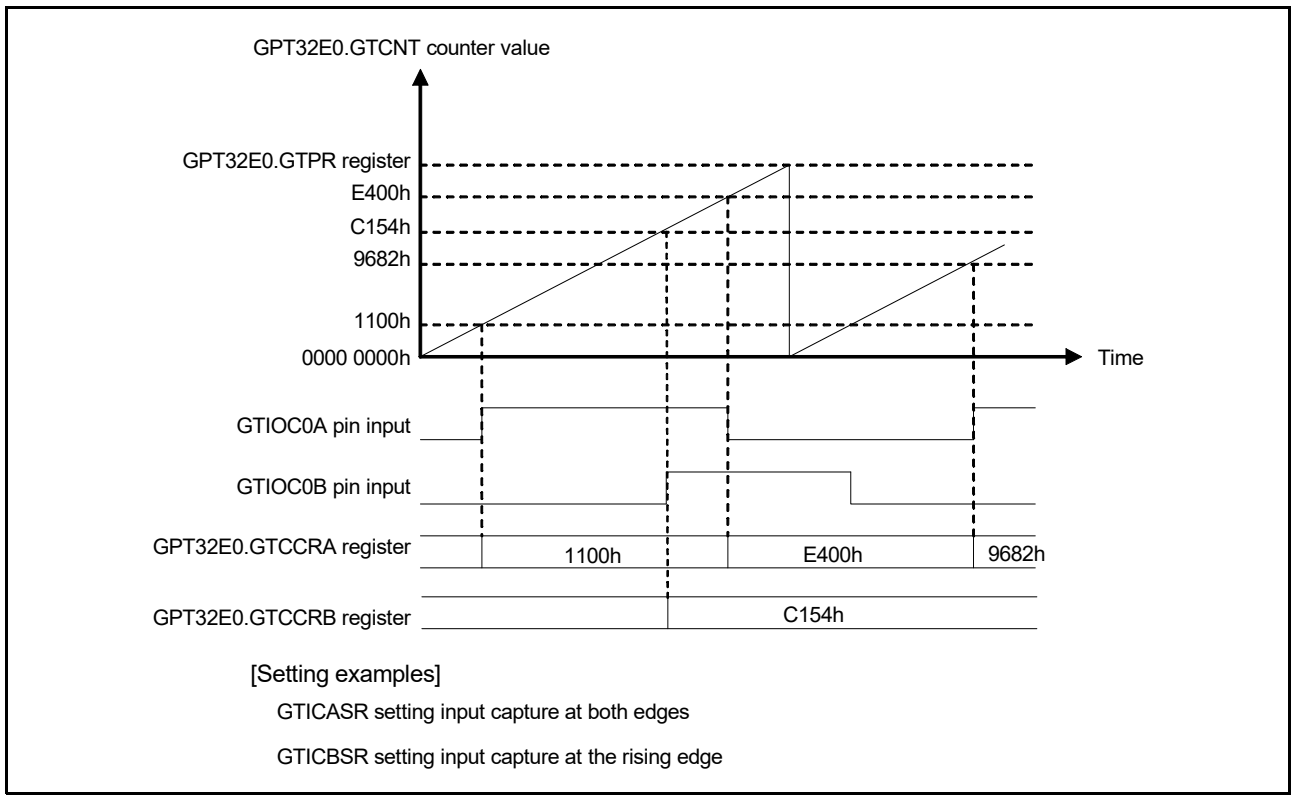


Figure 12.15 Example of input capture operation

Figure 12.16 shows an example setting for an input capture operation with count operation by the count clock.

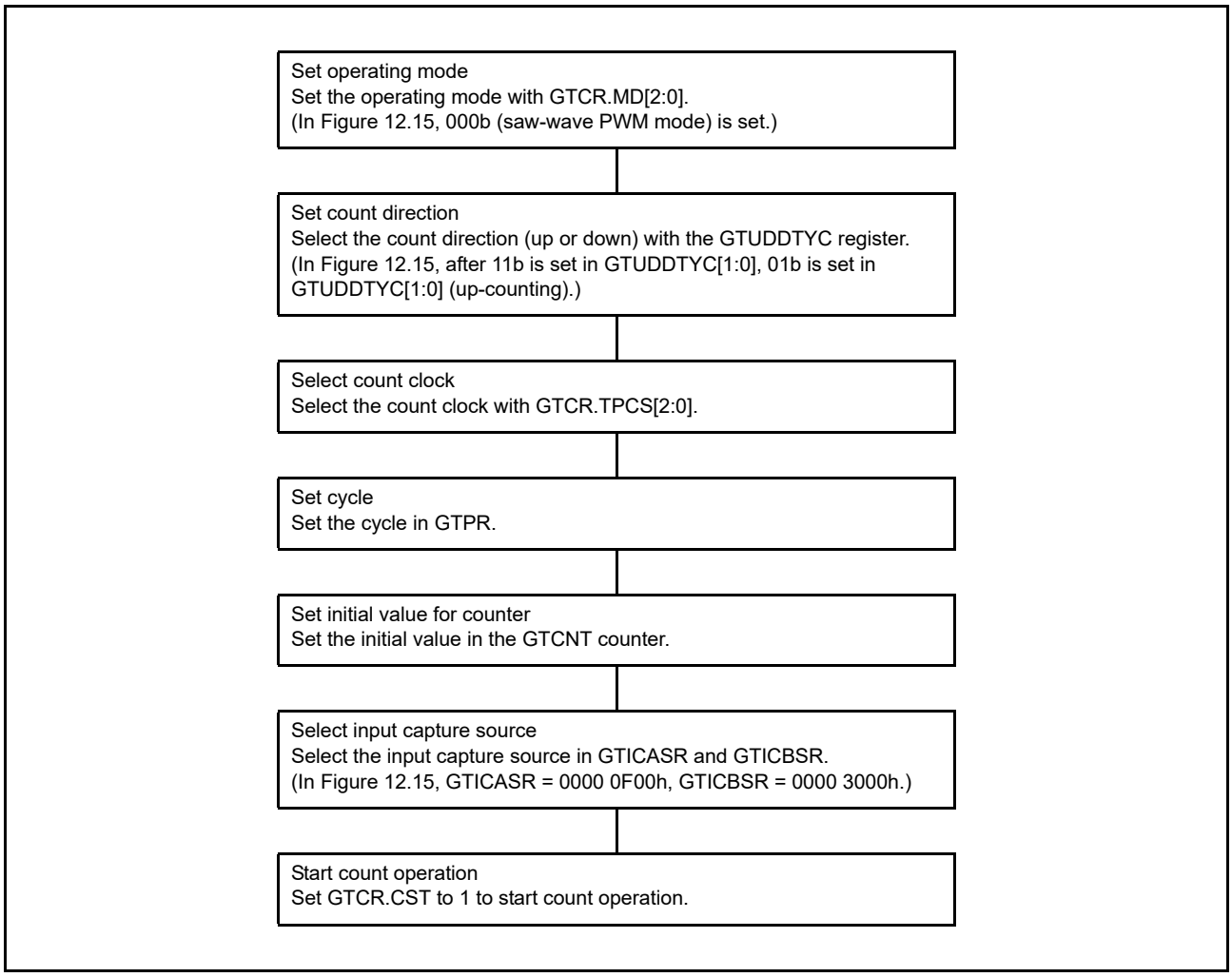


Figure 12.16 Example setting for input capture operation

12.3.2 Buffer Operation

The following buffer operations can be set with GTBER:

- GTPR, GTPBR, and GTPDBR
- GTCCRA, GTCCRC, and GTCCRD
- GTCCRB, GTCCRE, and GTCCRF
- GTADTRA, GTADTBRA, and GTADTDBRA
- GTADTRB, GTADTBRB, and GTADTDBRB.

The following buffer operations can be set with GTDTCR:

- GTDVU and GTDBU
- GTDVD and GTDBD.

12.3.2.1 GTPR Register Buffer Operation

GTPBR can function as a buffer register for GTPR, and GTPDBR can function as a buffer register for GTPBR (double-buffer register for GTPR). The buffer transfer is performed at an overflow during up-counting or an underflow during down-counting in saw-wave mode or in event count, and at a trough in triangle-wave mode.

In saw-wave mode or in event count, the buffer transfer is performed when the following counter clear operations occur during counting:

- Clear by hardware sources (the clear source is selected in GTCSR[23:0])
- Clear by software (when GTCSR.CCLR bit is 1 and GTCLR[n] bit is set to 1, n = channel number).

To set GTPR to function as double buffer, set GTBER.PR[1:0] to 10b or 11b. To set GTPR to not function as a buffer, set GTBER.PR[1:0] to 00b.

Figure 12.17 to Figure 12.19 show examples of GTPR buffer operation and Figure 12.20 shows an example setting for GTPR buffer operation.

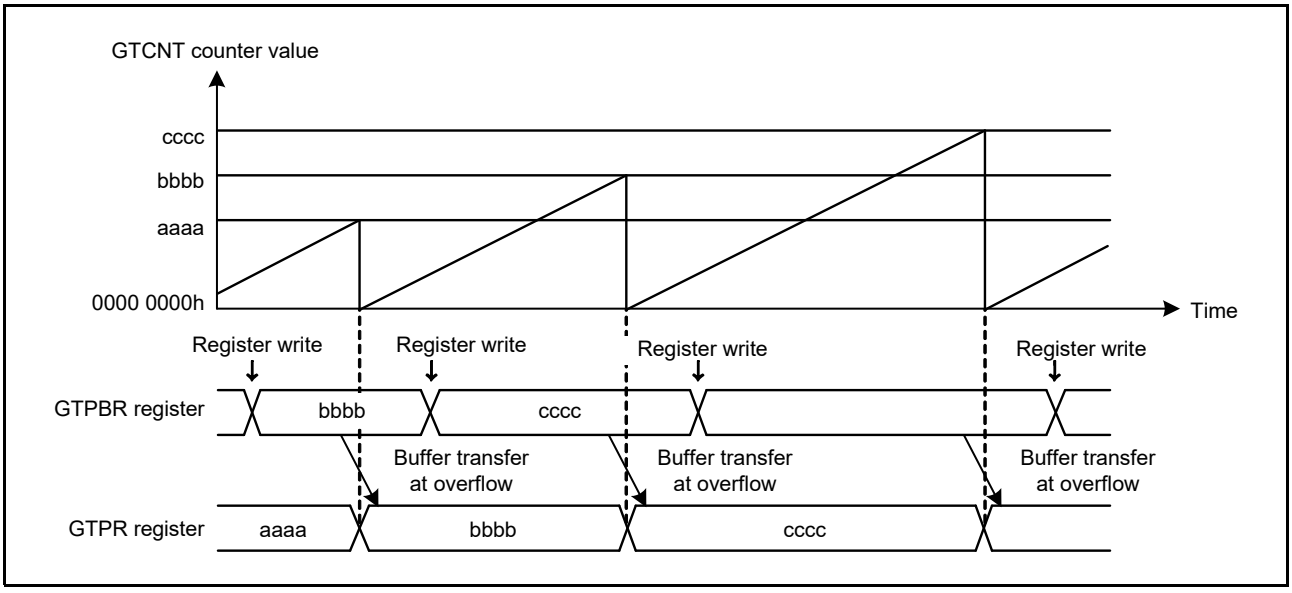


Figure 12.17 Example of GTPR buffer operation with saw waves in up-counting

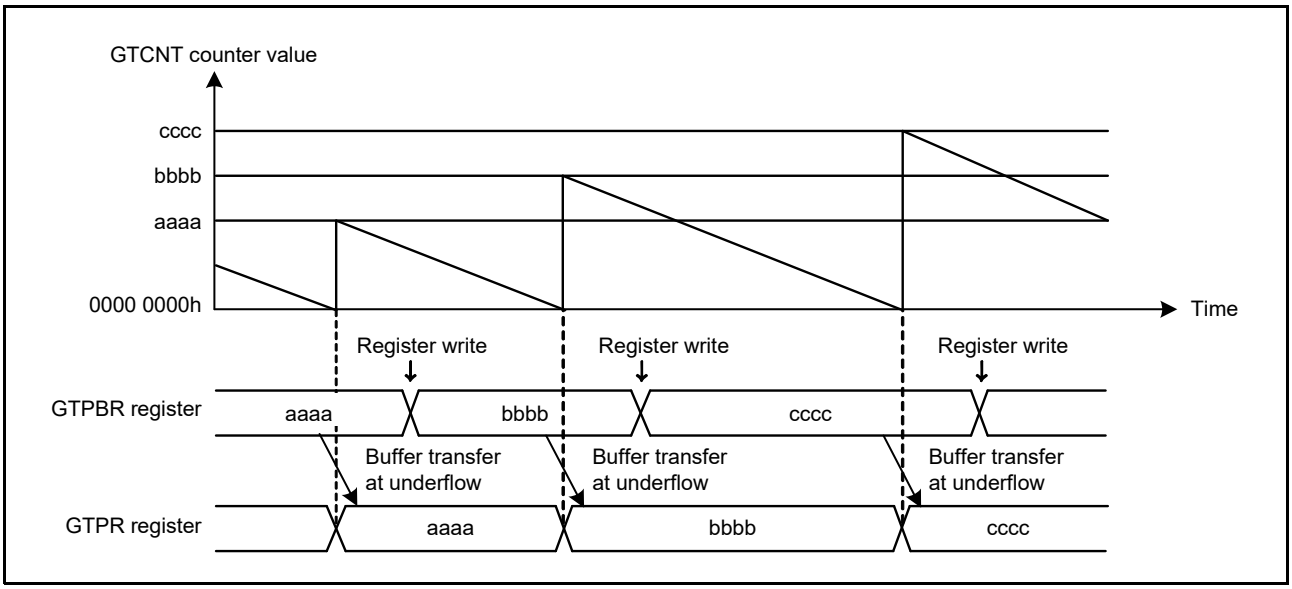


Figure 12.18 Example of GTPR buffer operation with saw waves in down-counting

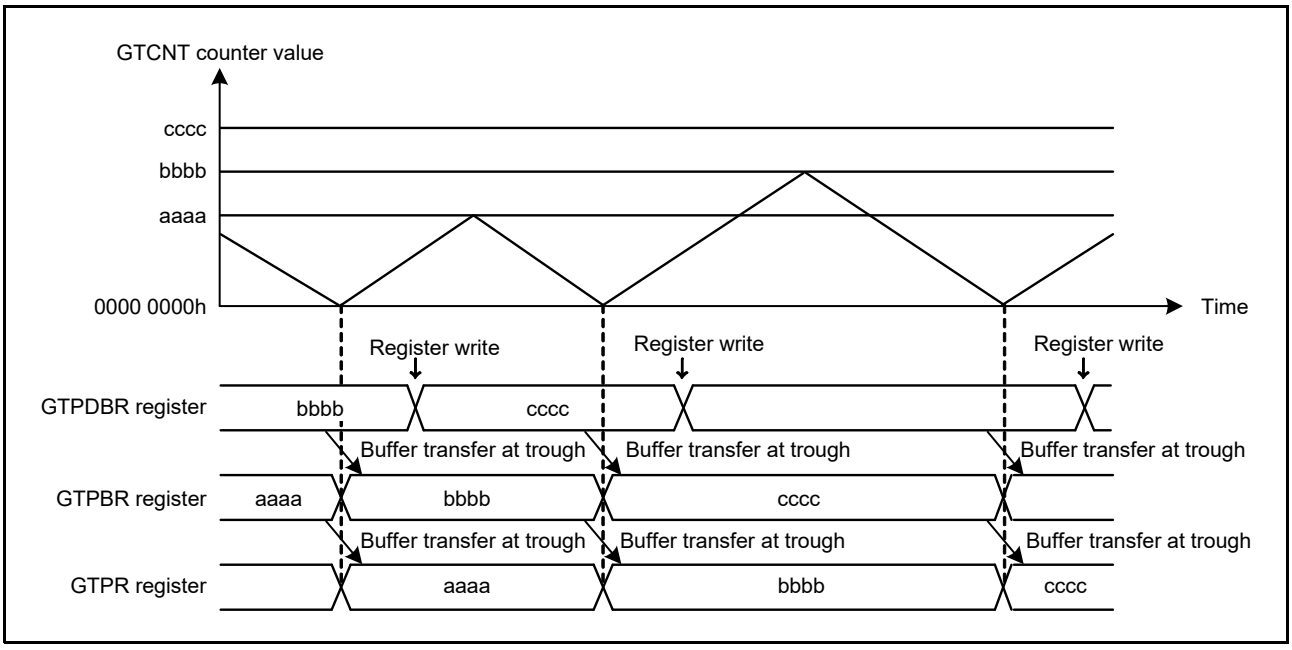


Figure 12.19 Example of GTPR double buffer operation with triangle waves

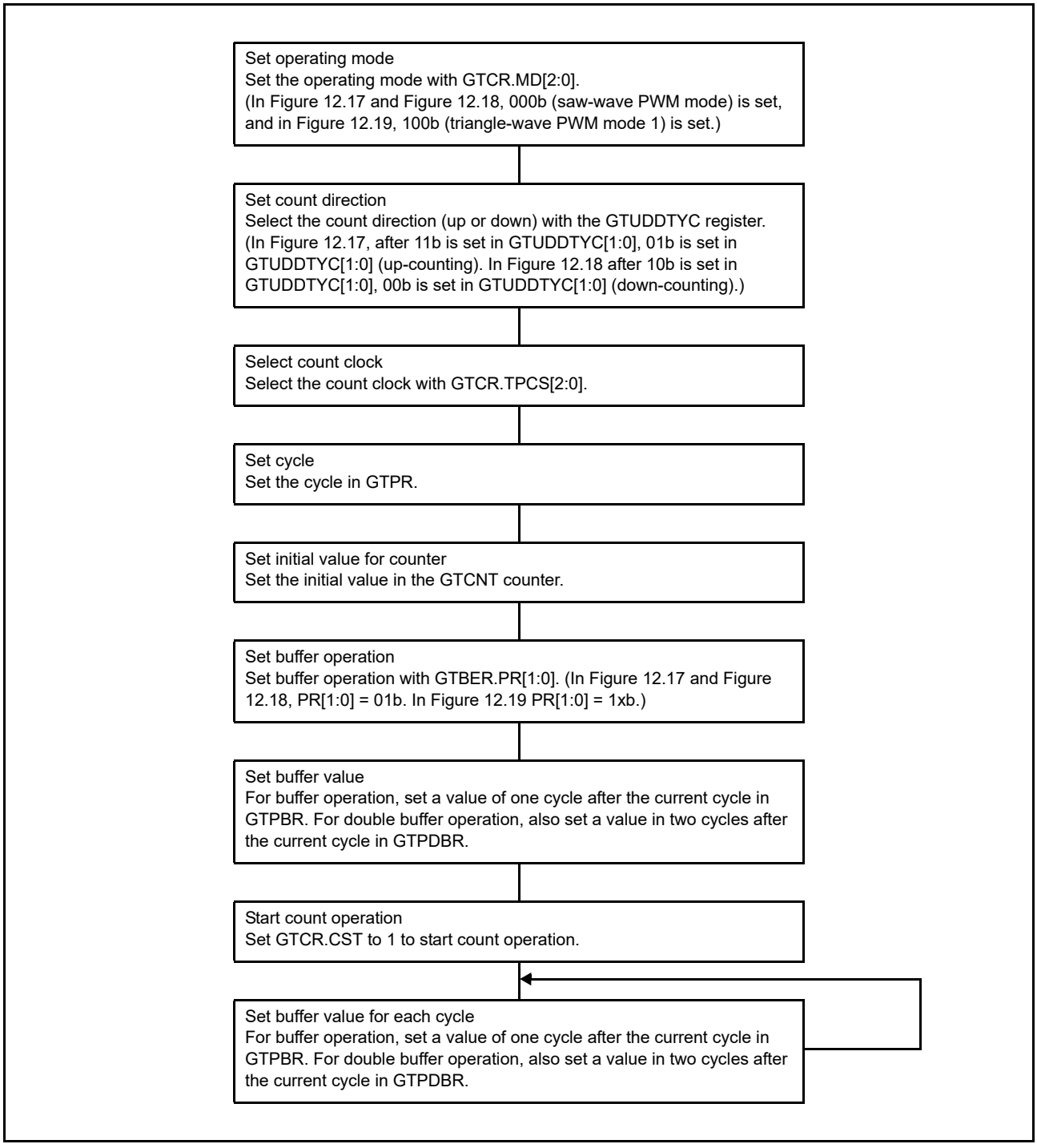


Figure 12.20 Example setting for GTPR buffer operation

12.3.2.2 Buffer Operation for GTCCRA and GTCCRB

GTCCRC can function as the GTCCRA buffer register and GTCCRD can function as the GTCCRC buffer register (double-buffer register for GTCCRA). Similarly, GTCCRE can function as the GTCCRB buffer register and GTCCRF can function as the GTCCRE buffer register (double-buffer register for GTCCRB).
To set GTCCRA or GTCCRB to function as a double buffer, set GTBER.CCRA[1:0] or GTBER.CCRB[1:0] to 10b or 11b. For single buffer operation, set GTBER.CCRA[1:0] or GTBER.CCRB[1:0] to 01b. To set GTCCRA or GTCCRB to not function as a buffer, set GTBER.CCRA[1:0] or GTBER.CCRB[1:0] to 00b.

(1) When GTCCRA or GTCCRB functions as an output compare register

Buffer transfer occurs in the following situations:

- Buffer transfer by overflow or underflow
  Buffer transfer is performed at an overflow (during up-counting) or an underflow (during down-counting) in saw-wave mode or in event count operation. In triangle-wave mode, buffer transfer is performed at a trough (triangle-wave PWM mode 1) or a crest and trough (triangle-wave PWM mode 2).
- Buffer transfer by counter clear
  In saw-wave mode or in event count operation, during counting, buffer transfer (which is the same as an overflow during up-counting or an underflow during down-counting) is performed by the counter clear sources the same as shown in section 12.3.2.1, GTPR Register Buffer Operation. In triangle-wave mode, buffer transfer is not performed by the counter clear.
- Forcible buffer transfer
  When 1 is written to the GTBER.CCRSWT bit while counting is stopped, the GTCCRA and the GTCCRB register buffer transfer are performed forcibly in saw-wave mode, in event count operation and in triangle-wave mode. Additionally, buffer transfer from the GTCCRD register to temporary register A and from the GTCCRF register to temporary register B are performed in saw-wave 1 shot pulse mode or triangle-wave PWM mode 3.

Figure 12.21 to Figure 12.23 show examples of GTCCRA and GTCCRB buffer operation and Figure 12.24 shows an example setting for GTCCRA and GTCCRB buffer operation.

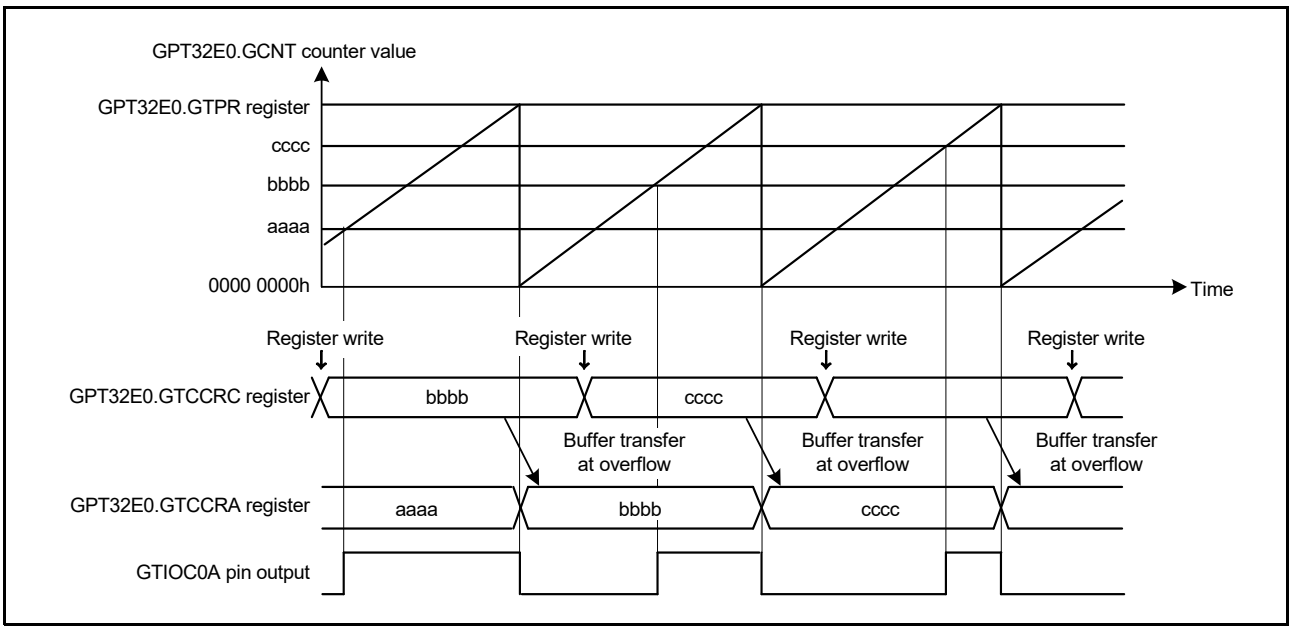


Figure 12.21 Example of GTCCRA and GTCCRB buffer operation with output compare, saw waves in up-counting, high output at GTCCRA compare match, and low output at cycle end

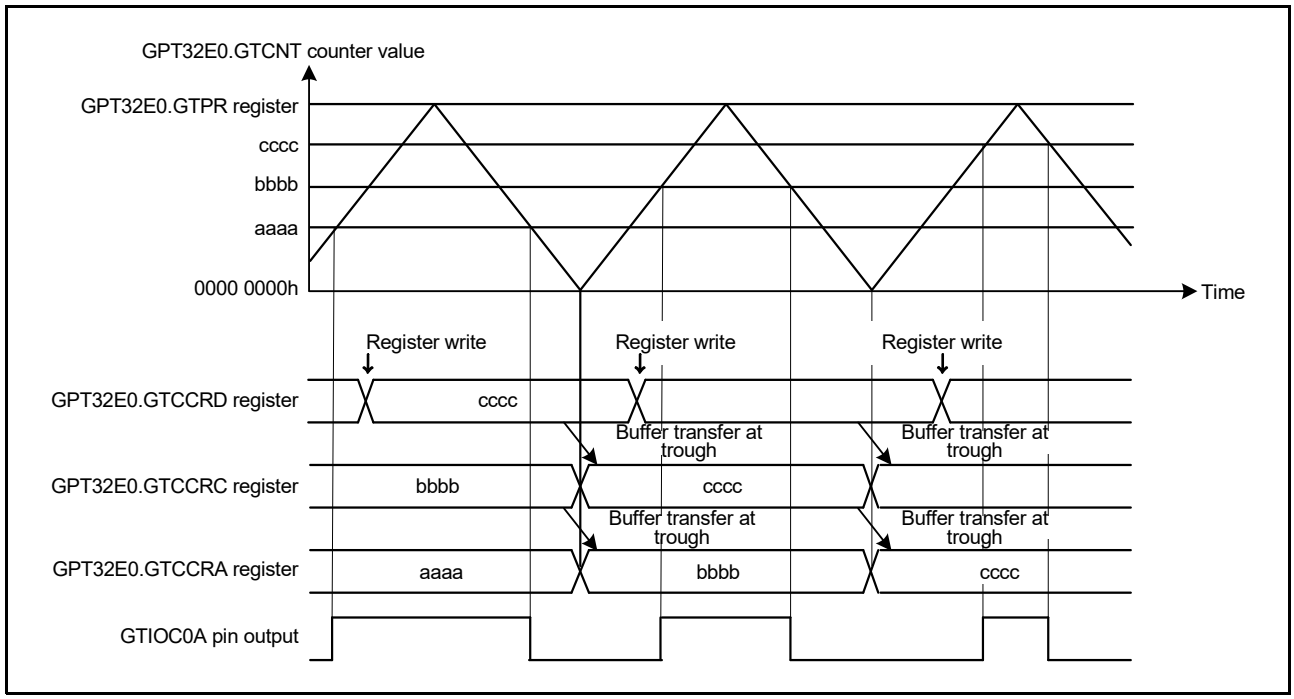


Figure 12.22 Example of GTCCRA and GTCCRB double buffer operation with output compare, triangle waves, buffer operation at trough, output toggled at GTCCRA compare match, and output retained at cycle end

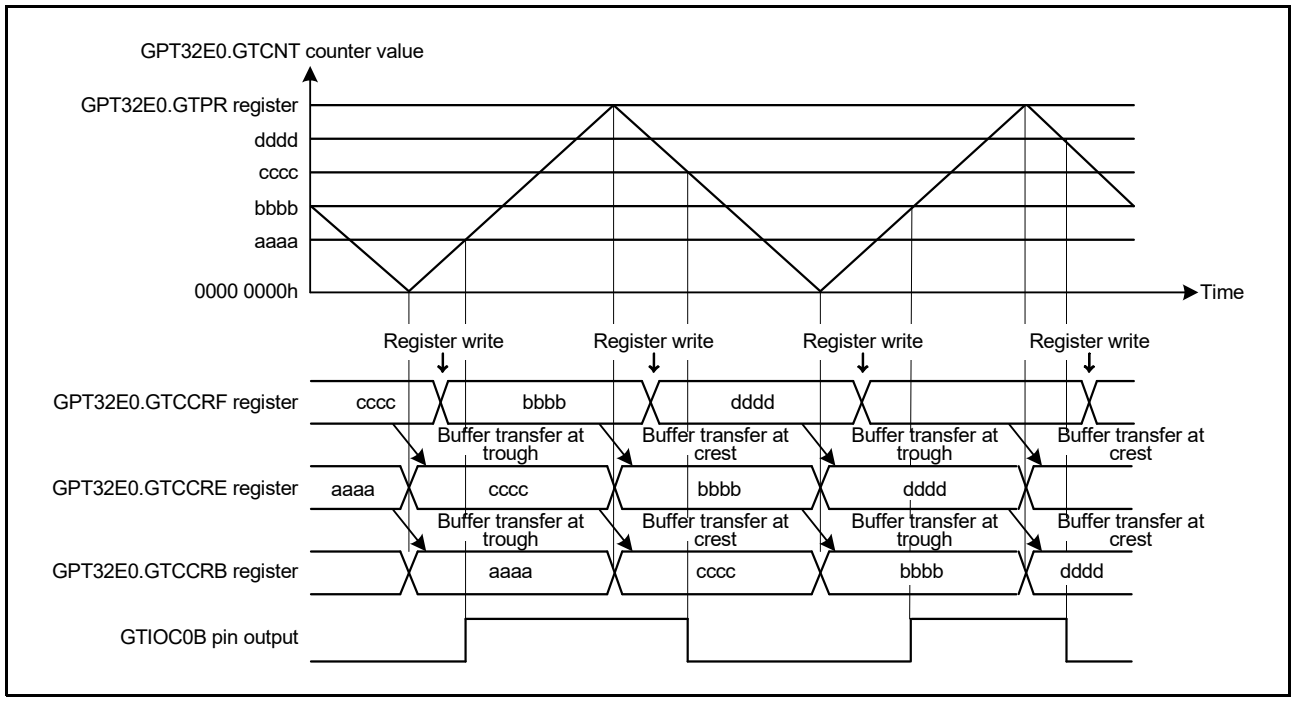


Figure 12.23 Example of GTCCRA and GTCCRB double buffer operation with output compare, triangle waves, buffer operation at both troughs and crests, output toggled at GTCCRB compare match, and output retained at cycle end

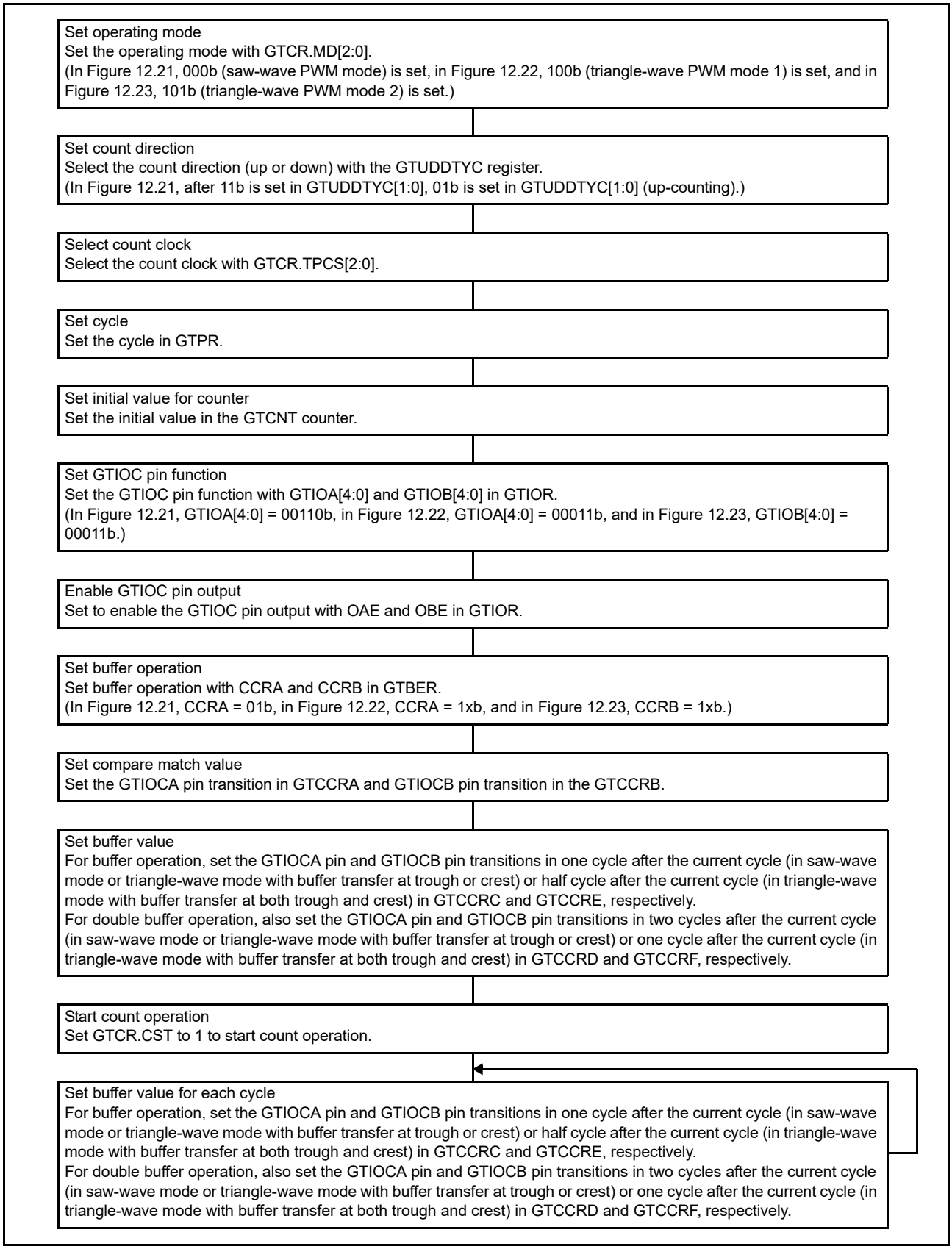


Figure 12.24 Example setting for GTCCRA and GTCCRB buffer operation with output compare

(2) When GTCCRA or GTCCRB functions as an input capture register

When an input capture is generated, the GTCNT counter value is transferred to GTCCRA and GTCCRB and the stored GTCCRA and GTCCRB register values are transferred to the buffer registers. In input capture operation, the buffer transfer is not performed by the counter clear.

Figure 12.25 and Figure 12.26 show examples of GTCCRA and GTCCRB buffer operation and Figure 12.27 shows an example setting for GTCCRA and GTCCRB buffer operation.

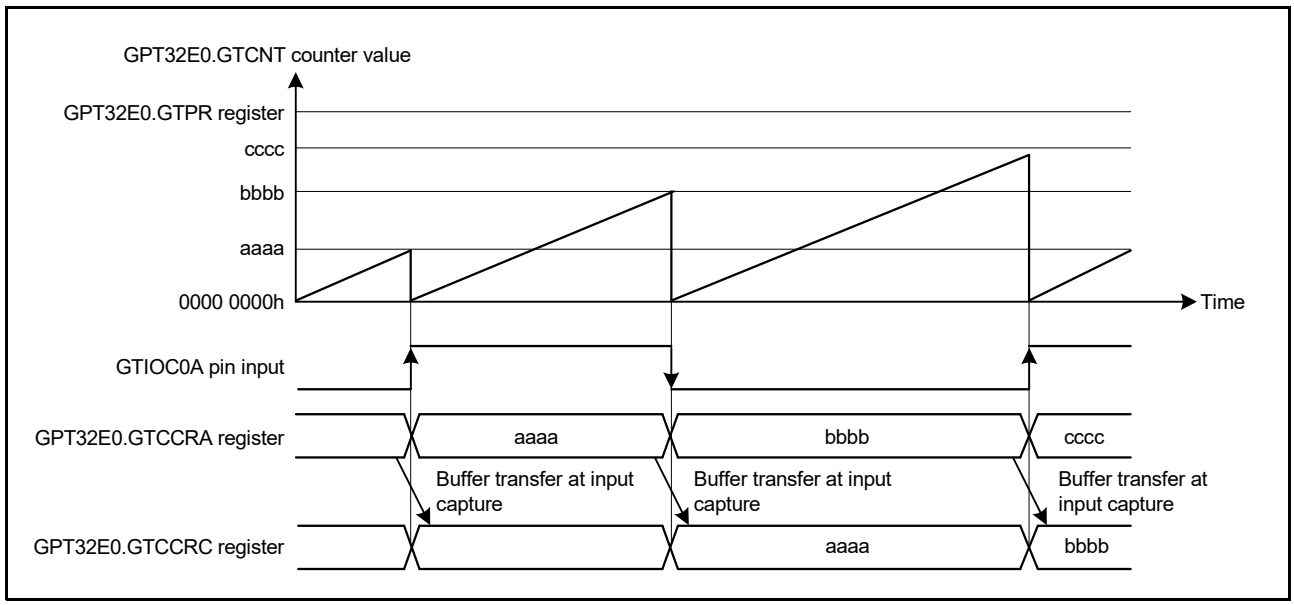


Figure 12.25 Example of GTCCRA and GTCCRB buffer operation with input capture at both edges of GTIOC0A input, saw waves in up-counting, and GTCNT counter cleared at both edges of GTIOC0A input

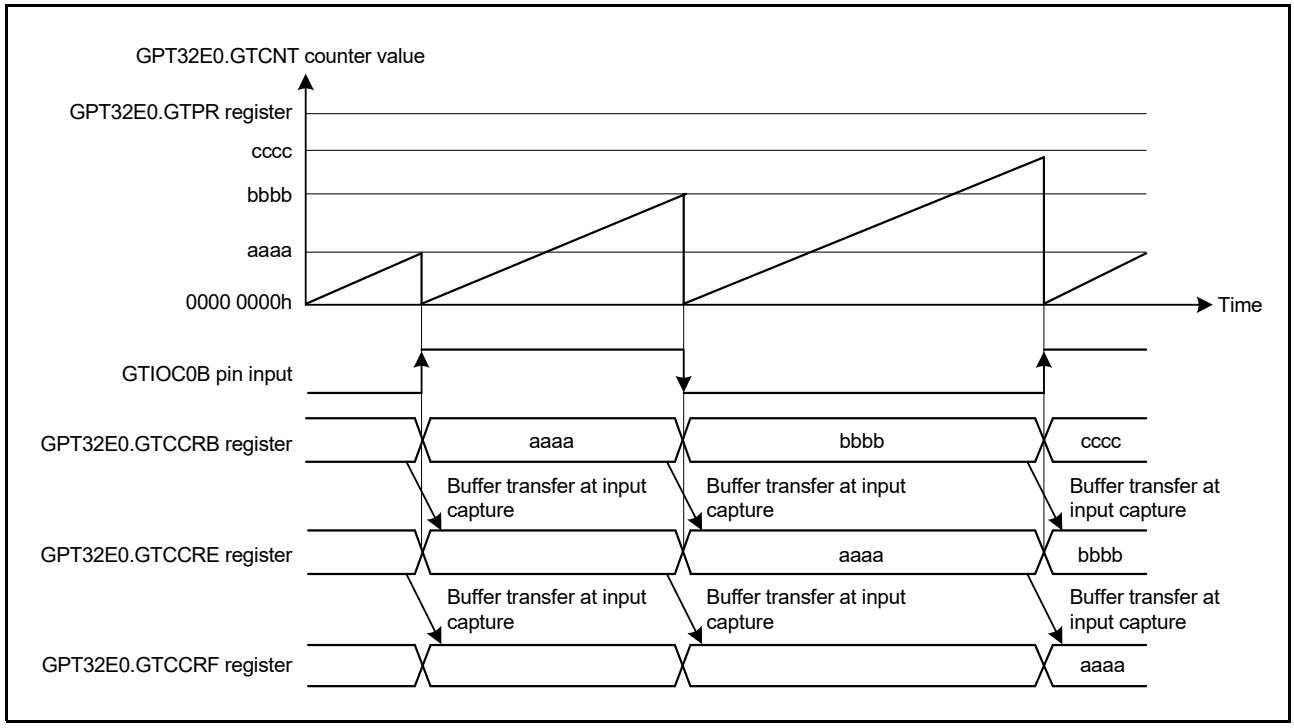


Figure 12.26 Example of GTCCRA and GTCCRB double buffer operation with input capture at both edges of GTIOC0B input, saw waves in up-counting, and GTCNT counter cleared at both edges of GTIOC0B input

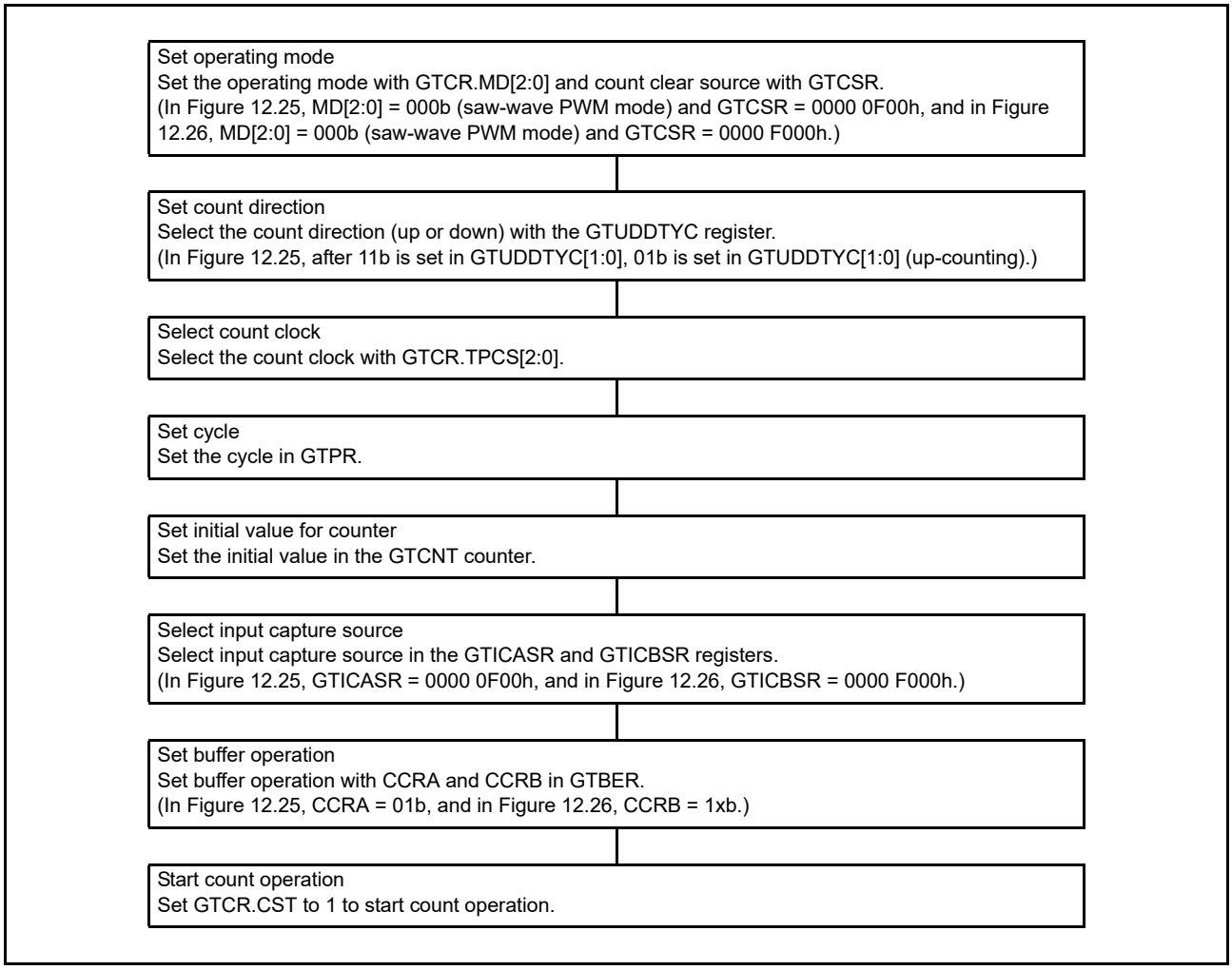


Figure 12.27 Example setting for GTCCRA and GTCCRB buffer operation with input capture

12.3.2.3 Buffer Operation for GTADTRA and GTADTRB

GTADTBRA can function as the GTADTRA buffer register and GTADTDBRA can function as the GTADTBRA buffer register (double-buffer register for GTADTRA). Similarly, GTADTBRB can function as the GTADTRB buffer register and GTADTDBRB can function as the GTADTBRB buffer register (double-buffer register for GTADTRB).
To set GTADTRA or GTADTRB to function as a double buffer, set GTBER.ADTDA or GTBER.ADTDB to 1. For single buffer operation, set GTBER.ADTDA or GTBER.ADTDB to 0. To set GTADTRA or GTADTRB to not function as a buffer, set GTBER.ADTTA[1:0] or GTBER.ADTTB[1:0] to 00b.
The buffer transfer timing can be set with the GTBER.ADTTA[1:0] bits. For saw waves, overflows (during up-counting) or underflows (during down-counting) can be selected. For triangle waves, crests are selected when GTBER.ADTTA[1:0] = 01b, troughs are selected when GTBER.ADTTA[1:0] = 10b, and both crests and troughs are selected when GTBER.ADTTA[1:0] = 11b.
Figure 12.28 to Figure 12.30 show examples of GTADTRA and GTADTRB buffer operation and Figure 12.31 shows an example setting for GTDTRA and GTADTRB buffer operation.

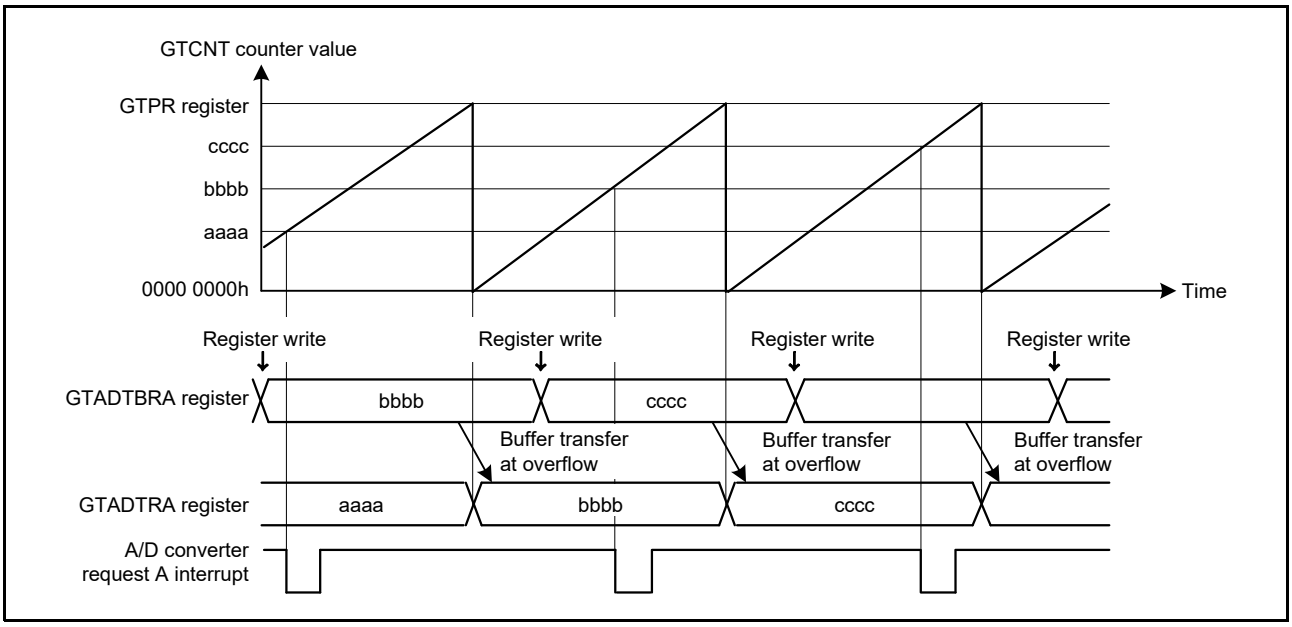


Figure 12.28 Example of GTADTRA and GTADTRB buffer operation with saw waves in up-counting and A/D converter start request interrupt generated by up-counting

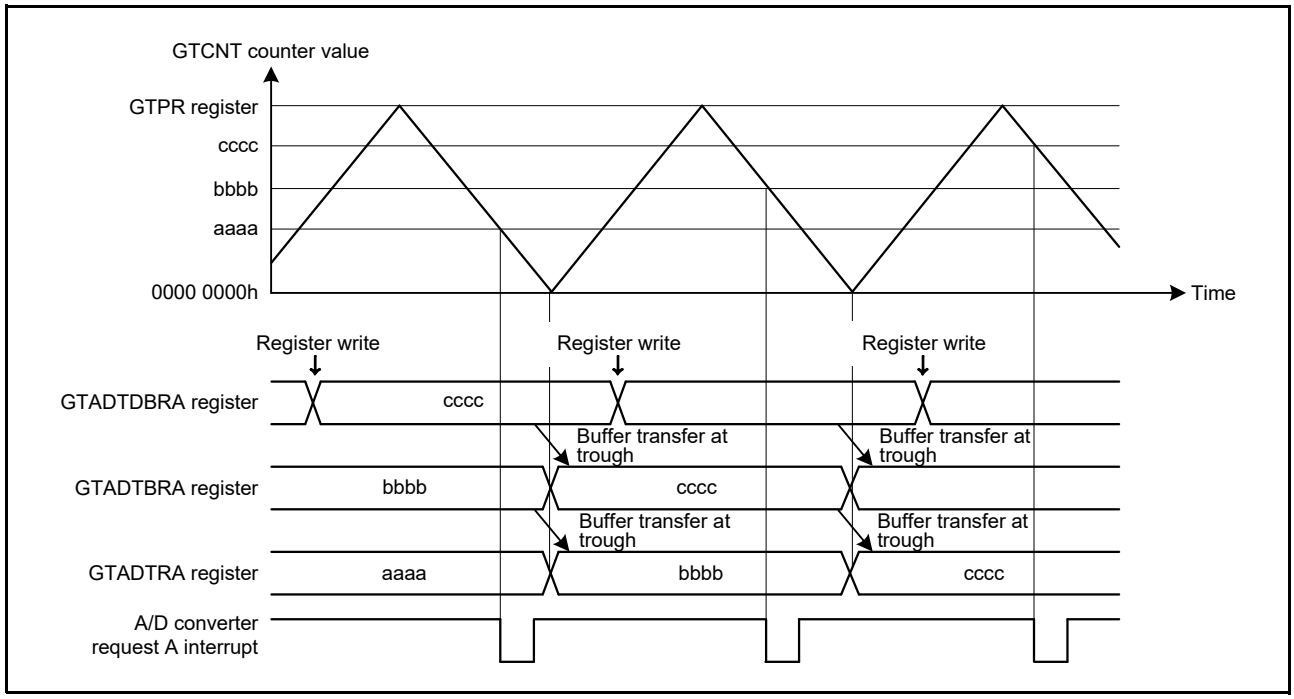


Figure 12.29 Example of GTADTRA and GTADTRB double buffer operation with triangle waves, buffer transfer at troughs, and A/D converter start request interrupt generated by down-counting

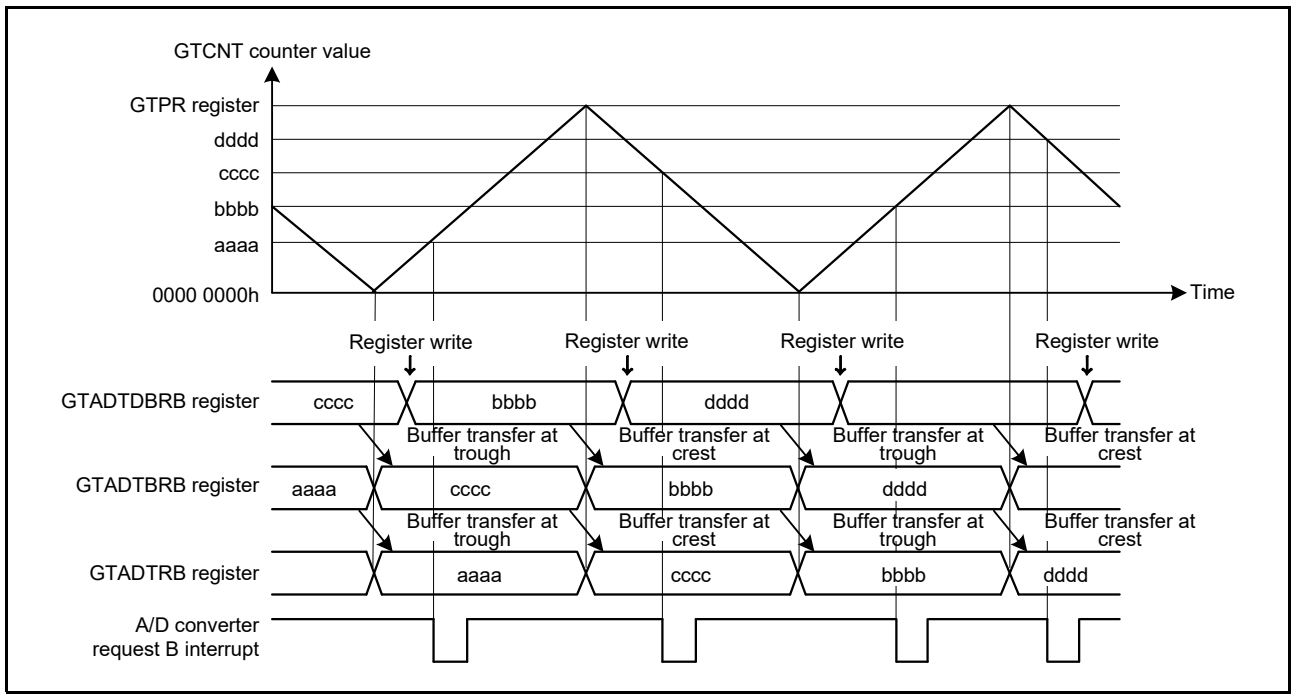


Figure 12.30 Example of GTADTRA and GTADTRB double buffer operation with triangle waves, buffer transfer at both troughs and crests, and A/D converter start request interrupt generated by both up- and down-counting

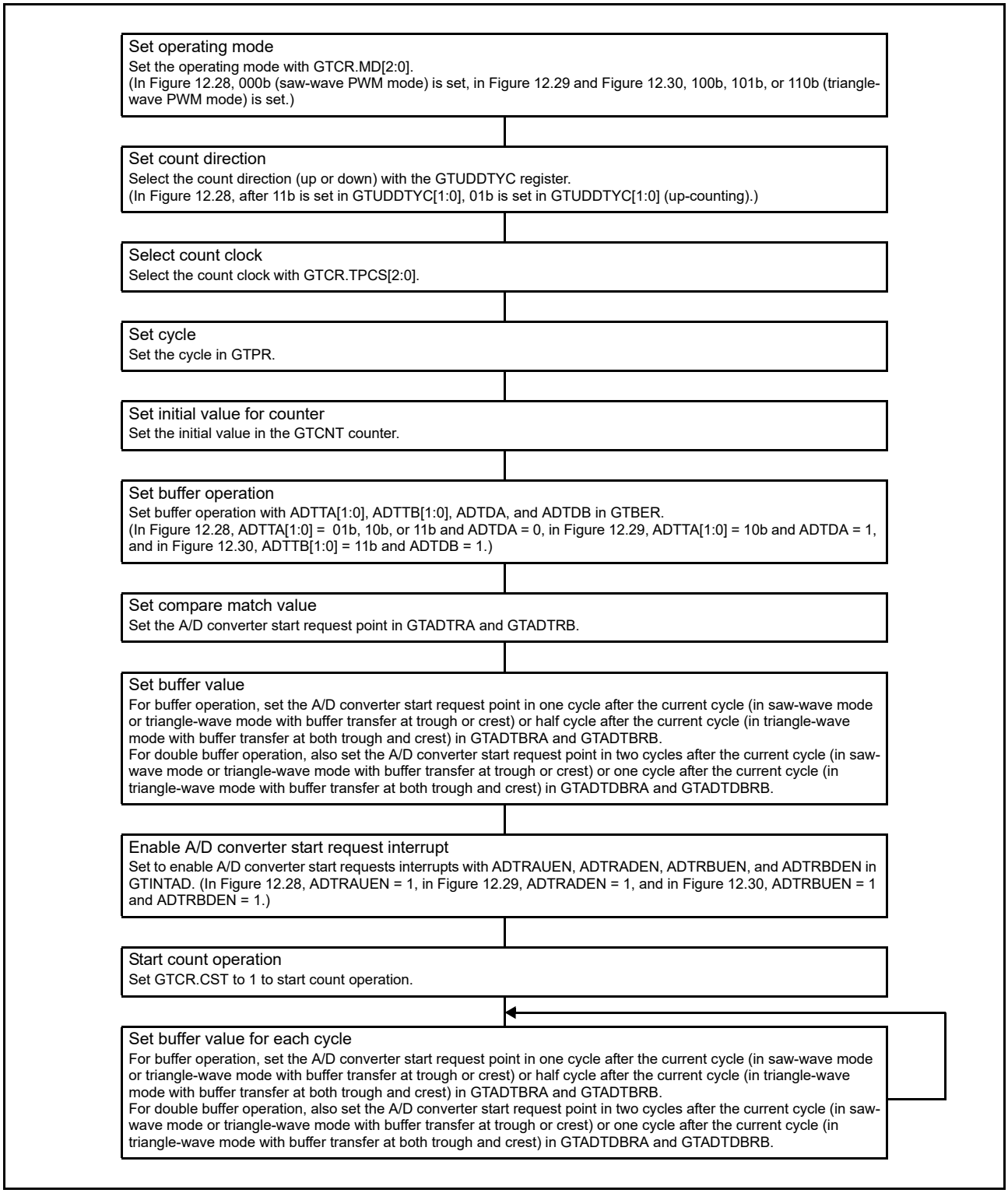


Figure 12.31 Example setting for GTADTRA and GTADTRB buffer operation

12.3.3 PWM Output Operating Mode

The GPT can output PWM waveforms to the GTIOCA or GTIOCB pin by a compare match between the GTCNT counter and GTCCRA or GTCCRB.
By setting GTDTCR, GTDVU, and GTDVD, the compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

12.3.3.1 Saw-Wave PWM Mode

In saw-wave PWM mode, GTCNT performs saw-wave (half-wave) operation by setting the cycle in GTPR.A PWM waveform is output to the GTIOCA or GTIOCB pin when a GTCCRA or GTCCRB compare match occurs. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the cycle end based on the GTIOR setting.
Figure 12.32 shows an example of saw-wave PWM mode operation, and Figure 12.33 shows an example setting for saw-wave PWM mode.

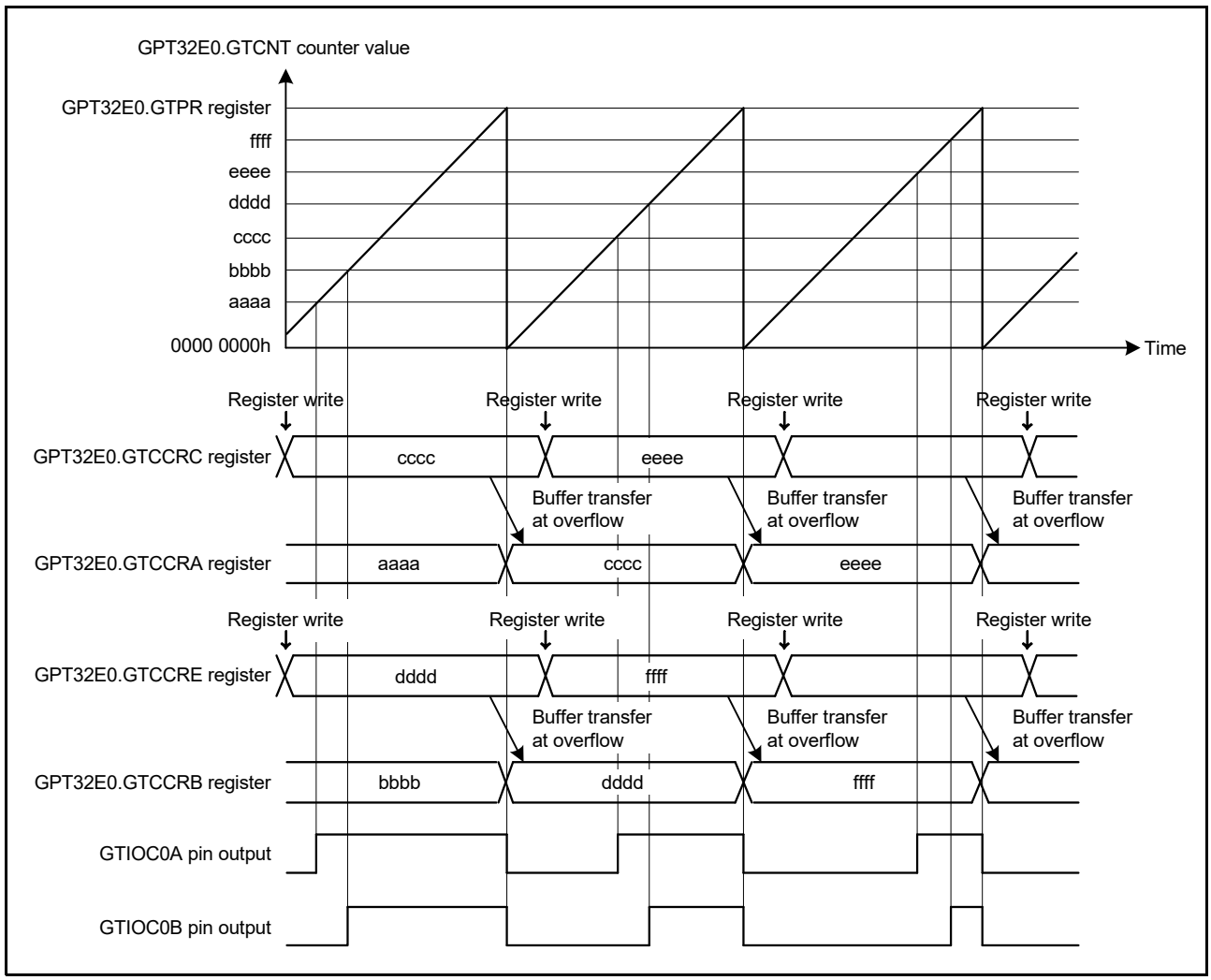


Figure 12.32 Example of saw-wave PWM mode operation with up-counting, buffer operation, high output at GTCCRA/GTCCRB compare match, and low output at cycle end

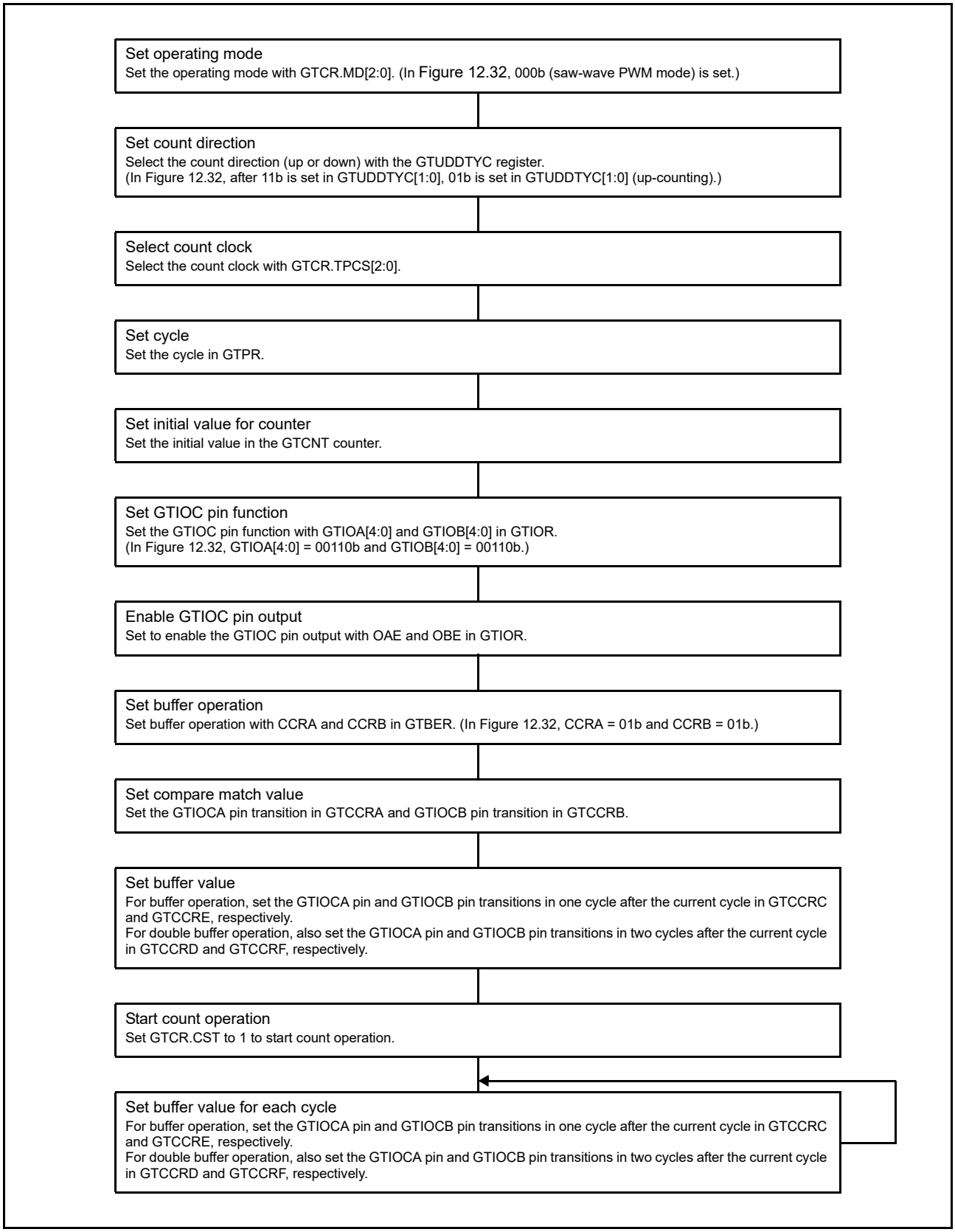


Figure 12.33 Example setting for saw-wave PWM mode

12.3.3.2 Saw-Wave One-Shot Pulse Mode

The saw-wave one-shot pulse mode is a mode in which the cycle is set in GTPR. The GTCNT counter performs saw-wave (half-wave) operation and a PWM waveform is output to the GTIOCA or GTIOCB pin at a compare match of GTCCRA or GTCCRB with buffer operation fixed.

Buffer operation in saw-wave one-shot pulse mode is different from the usual buffer operation. Buffer transfer is performed from the following:

- GTCCRC to GTCCRA at the cycle end
- GTCCRE to GTCCRB at the cycle end
- GTCCRD to temporary register A at the cycle end
- GTCCRF to temporary register B at the cycle end
- Temporary register A to GTCCRA at a GTCCRA compare match
- Temporary register B to GTCCRB at a GTCCRB compare match.

The pin output value can be selected from low output, high output, or toggle output separately for a compare match and the cycle end based on the GTIOR setting.

When 1 is written to the GTBER.CCRSWT bit while counting is stopped, the values of the GTCCRD and GTCCRF registers are forcibly transferred to the temporary registers A and B, respectively. By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

Figure 12.34 shows an example of saw-wave one-shot pulse mode operation, and Figure 12.35 shows an example setting for saw-wave one-shot pulse mode.

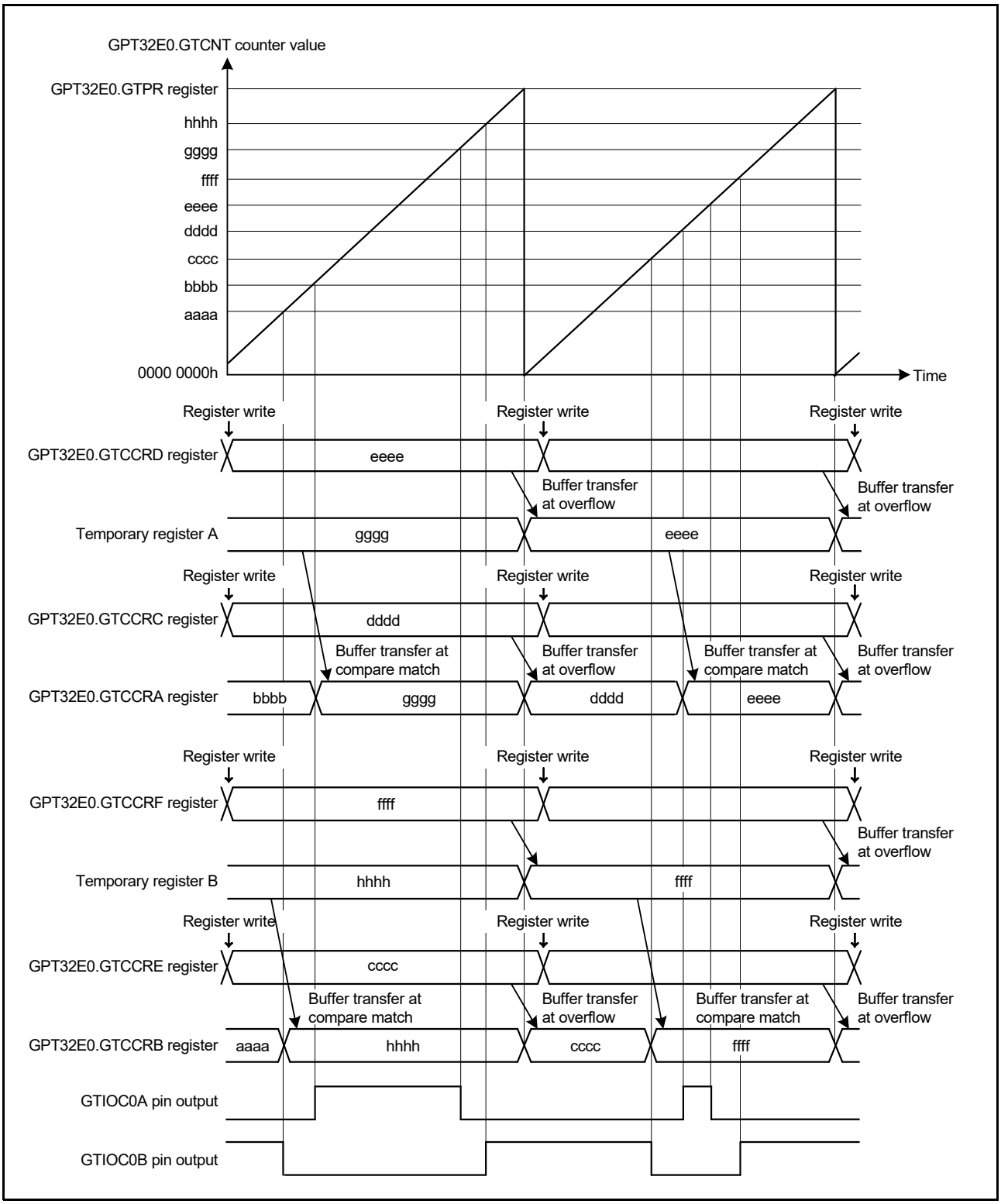


Figure 12.34 Example of saw-wave one-shot pulse mode operation with up-counting, low output from the GTIOC0A pin and high output from the GTIOC0B pin at count start, output toggled at GTCCRA/GTCCRB compare match, and output retained at cycle end

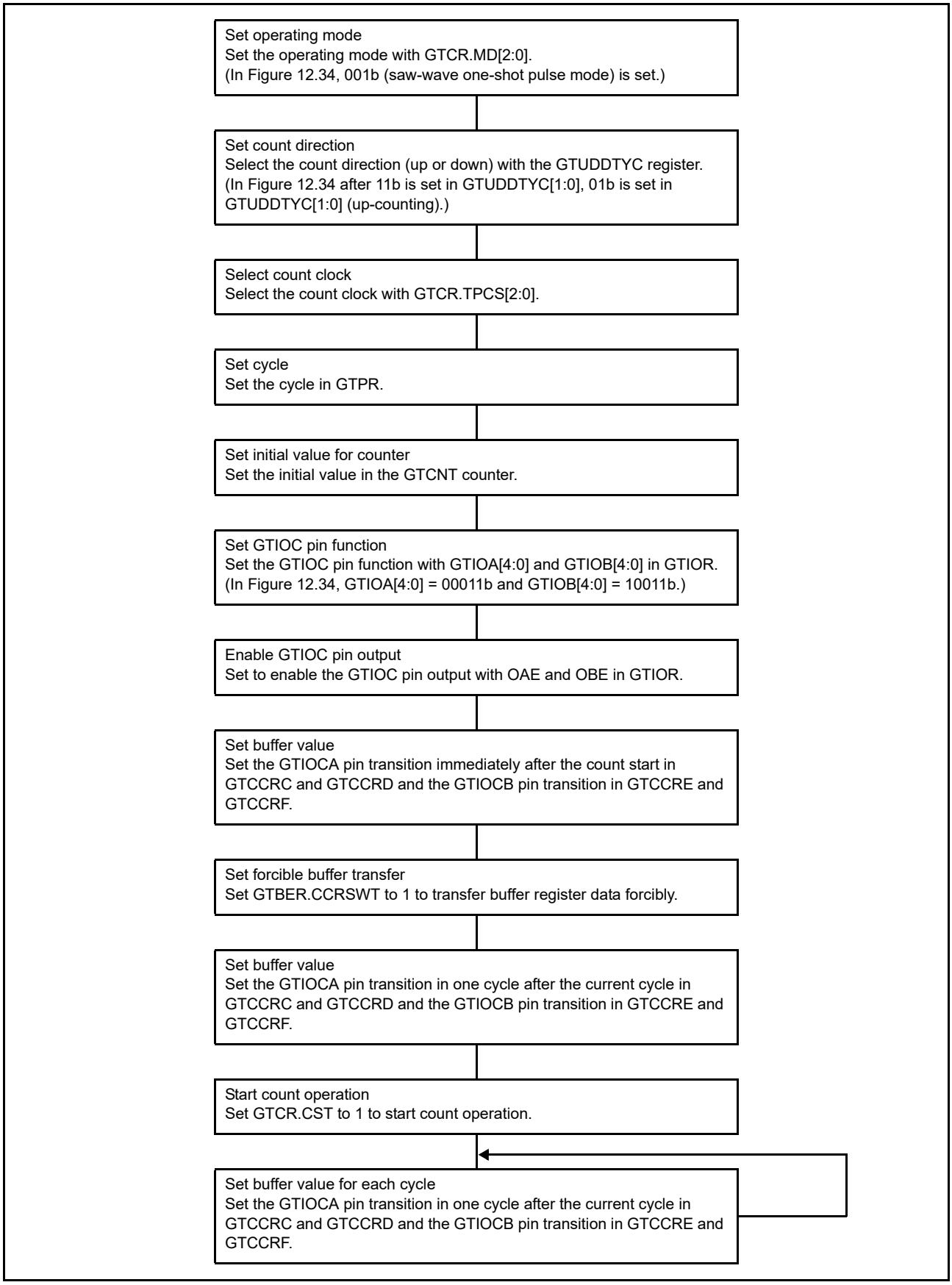


Figure 12.35 Example setting for saw-wave one-shot pulse mode

12.3.3.3 Triangle-Wave PWM Mode 1 (32-bit transfer at trough)

The triangle-wave PWM mode 1 is a mode in which the cycle is set in GTPR. The GTCNT counter performs triangle-wave (full-wave) operation, and a PWM waveform is output to the GTIOCA or GTIOCB pin when a GTCCRA or GTCCRB compare match occurs. Buffer transfer is performed at the trough. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the cycle end based on the GTIOR setting.

By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

Figure 12.36 shows an example of a triangle-wave PWM mode 1 operation, and Figure 12.37 shows an example setting for a triangle-wave PWM mode 1.

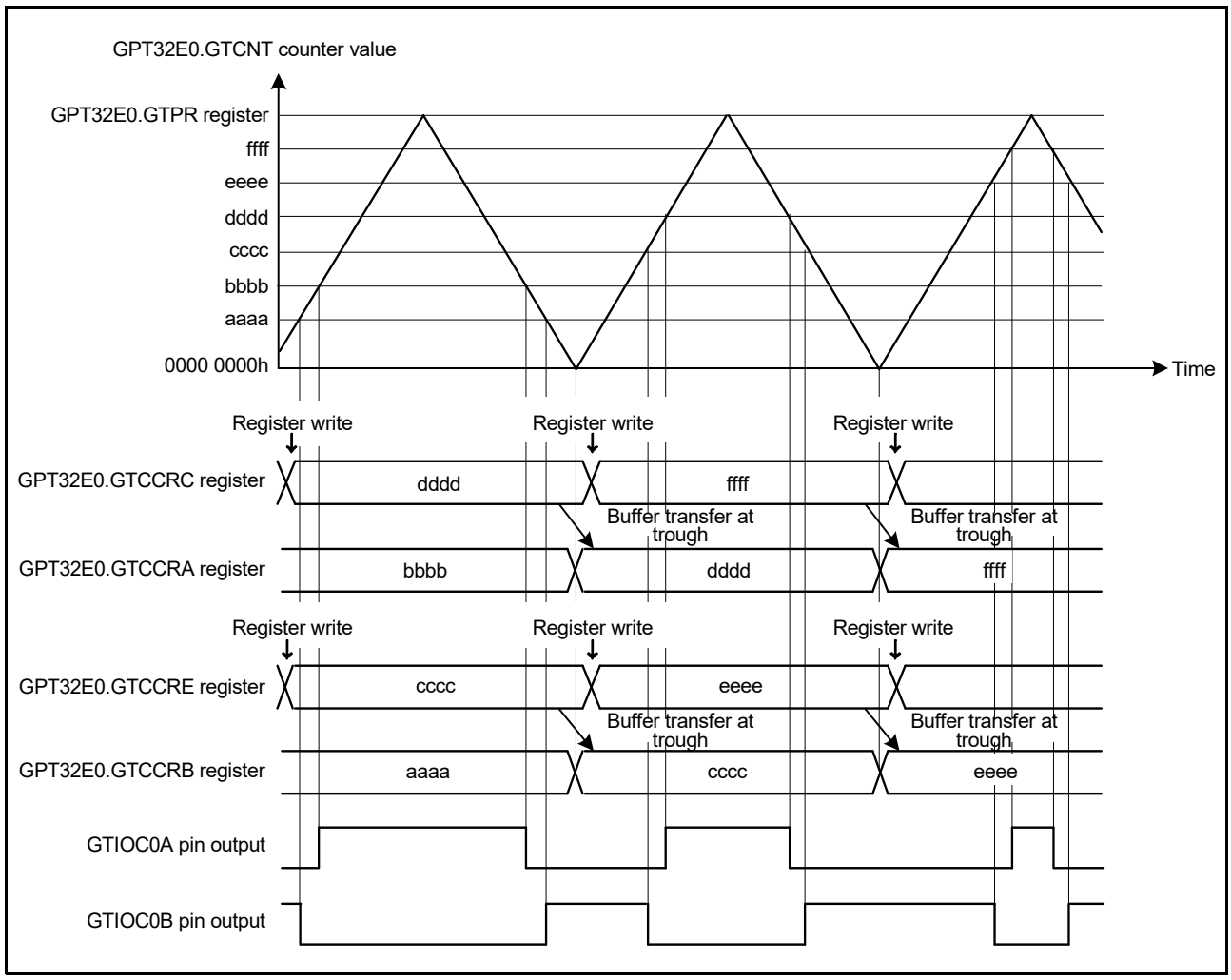


Figure 12.36 Example of triangle-wave PWM mode 1 operation with buffer operation, low output from the GTIOC0A pin and high output from the GTIOC0B pin at count start, output toggled at GTCCRA/GTCCRB register compare match, and output retained at cycle end

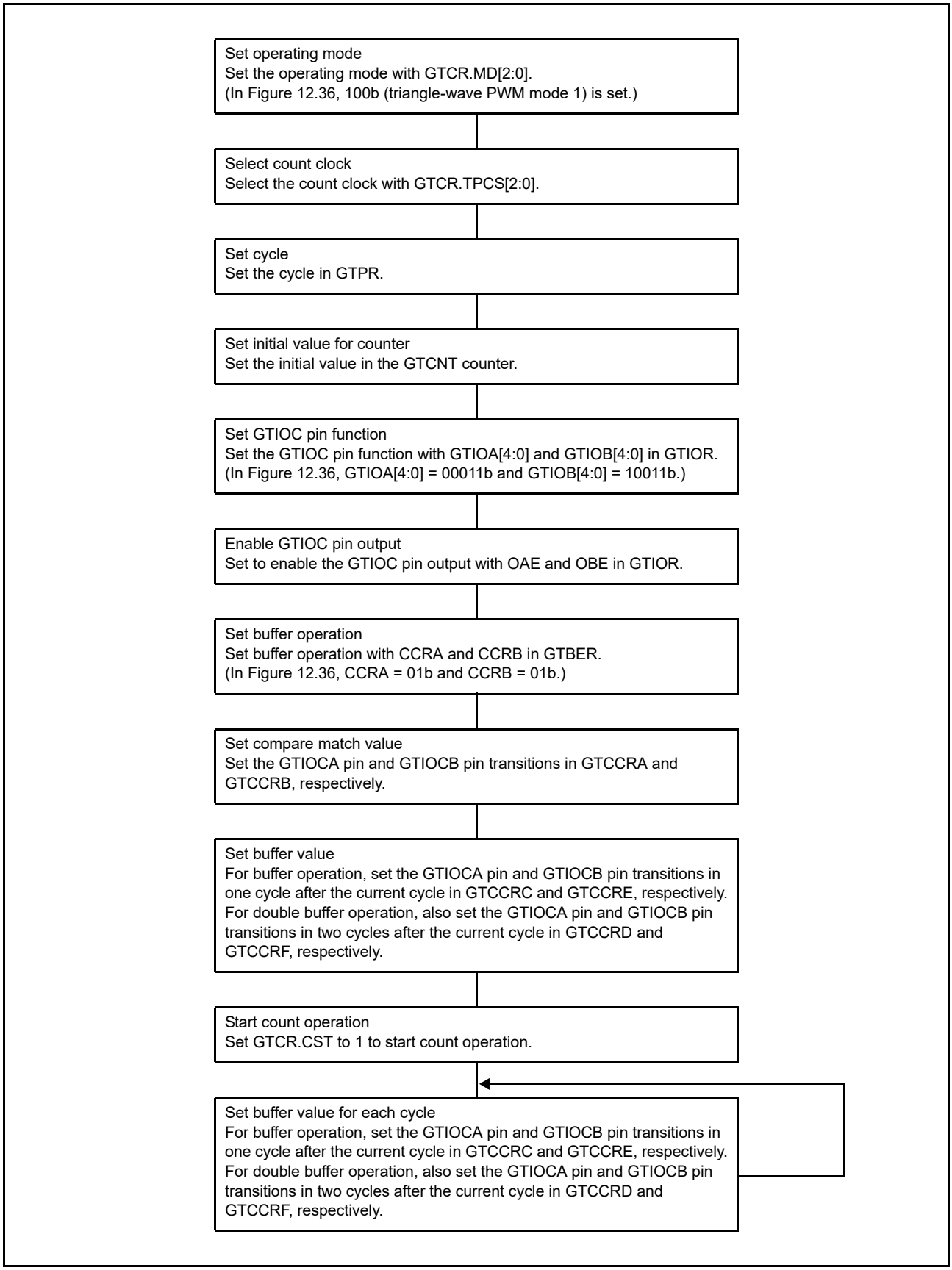


Figure 12.37 Example setting for triangle-wave PWM mode 1

12.3.3.4 Triangle-Wave PWM Mode 2 (32-bit transfer at crest and trough)

Similarly to triangle-wave PWM mode 1, in triangle-wave PWM mode 2 the cycle is set in GTPR. The GTCNT counter performs triangle-wave (full-wave) operation, and a PWM waveform is output to the GTIOCA or GTIOCB pin when a GTCCRA or GTCCRB compare match occurs. The buffer transfer is performed at both crests and troughs. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the cycle end based on the GTIOR setting.

By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

Figure 12.38 shows an example of triangle-wave PWM mode 2 operation, and Figure 12.39 shows an example setting for triangle-wave PWM mode 2.

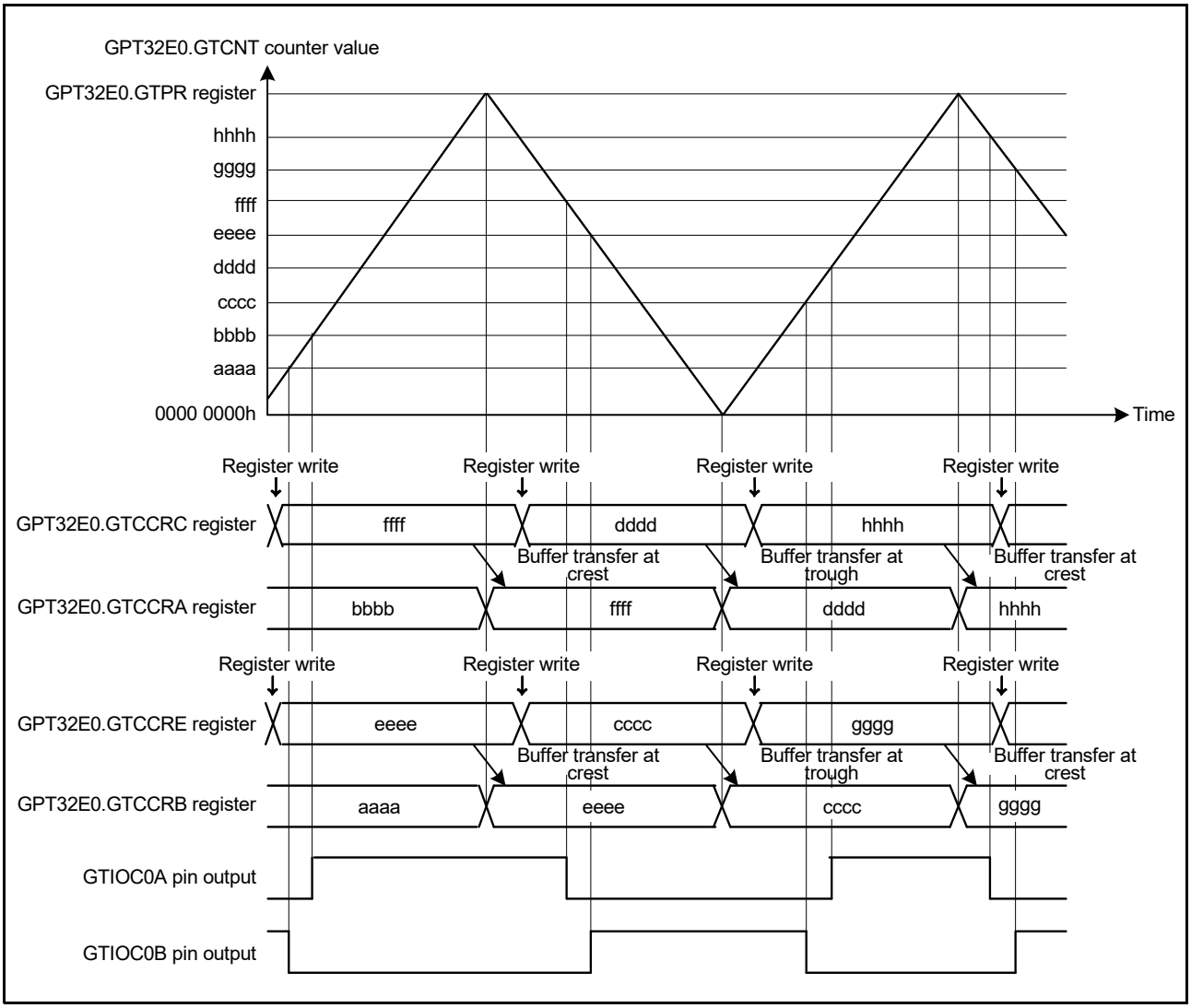


Figure 12.38 Example of triangle-wave PWM mode 2 operation with buffer operation, low output from the GTIOC0A pin and high output from the GTIOC0B pin at count start, output toggled at GTCCRA/GTCCRB compare match, and output retained at cycle end

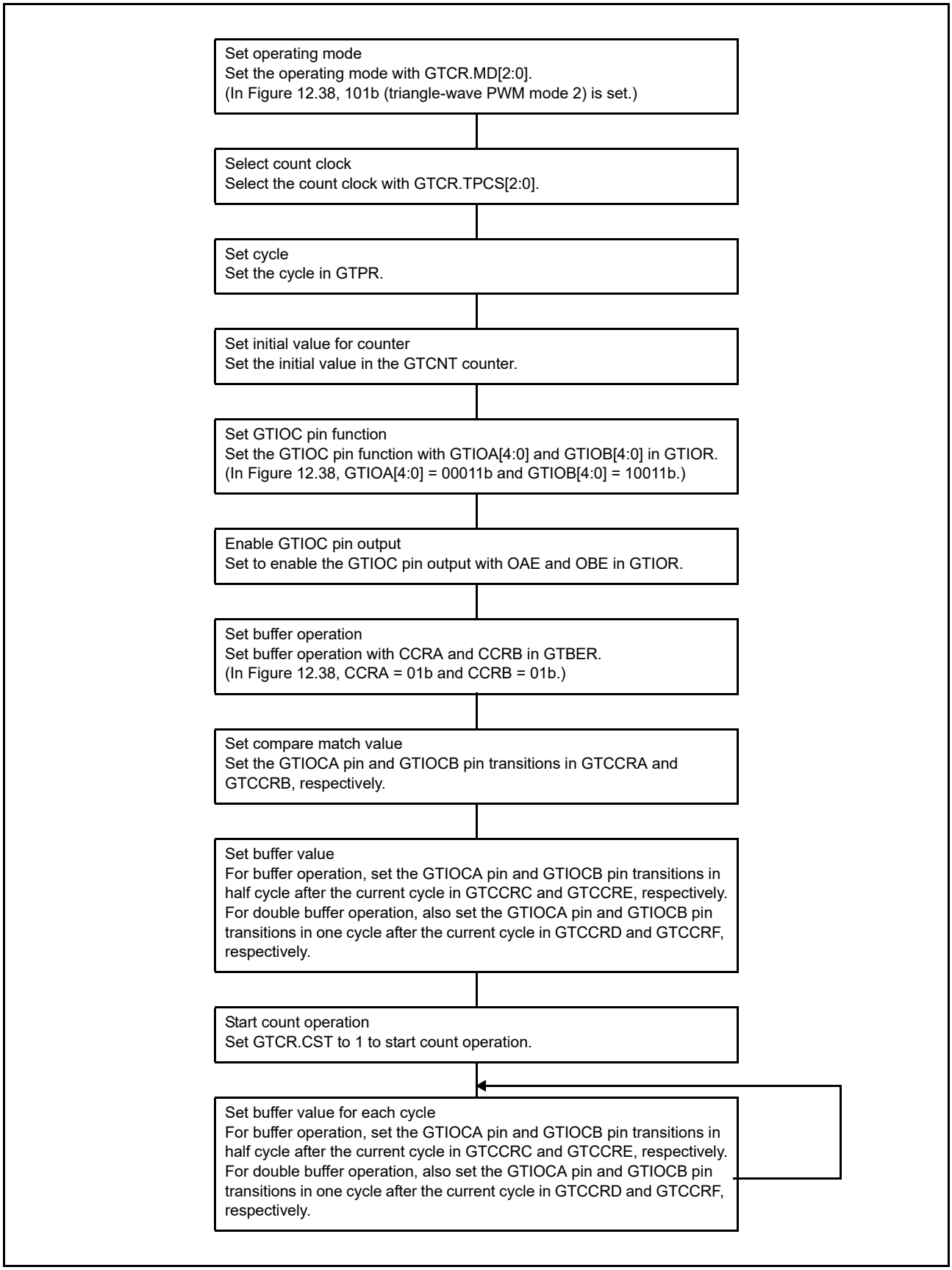


Figure 12.39 Example setting for triangle-wave PWM mode 2

12.3.3.5 Triangle-Wave PWM Mode 3 (64-bit transfer at trough)

The triangle-wave PWM mode 3 is a mode in which the cycle is set in GTPR. The GTCNT counter performs triangle-wave (full-wave) operation and a PWM waveform is output to the GTIOCA or GTIOCB pin at a compare match of GTCCRA or GTCCRB with buffer operation fixed. Buffer operation in triangle-wave PWM mode 3 is different from the usual buffer operation. Buffer transfer is performed from the following:

- GTCCRC to GTCCRA at the trough
- GTCCRE to GTCCRB at the trough
- GTCCRD to temporary register A at the trough
- GTCCRF to temporary register B at the trough
- Temporary register A to GTCCRA at the crest
- Temporary register B to GTCCRB at the crest

The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the cycle end based on the GTIOR setting. By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

Figure 12.40 shows an example of triangle-wave PWM mode 3 operation, and Figure 12.41 shows an example setting for triangle-wave PWM mode 3.

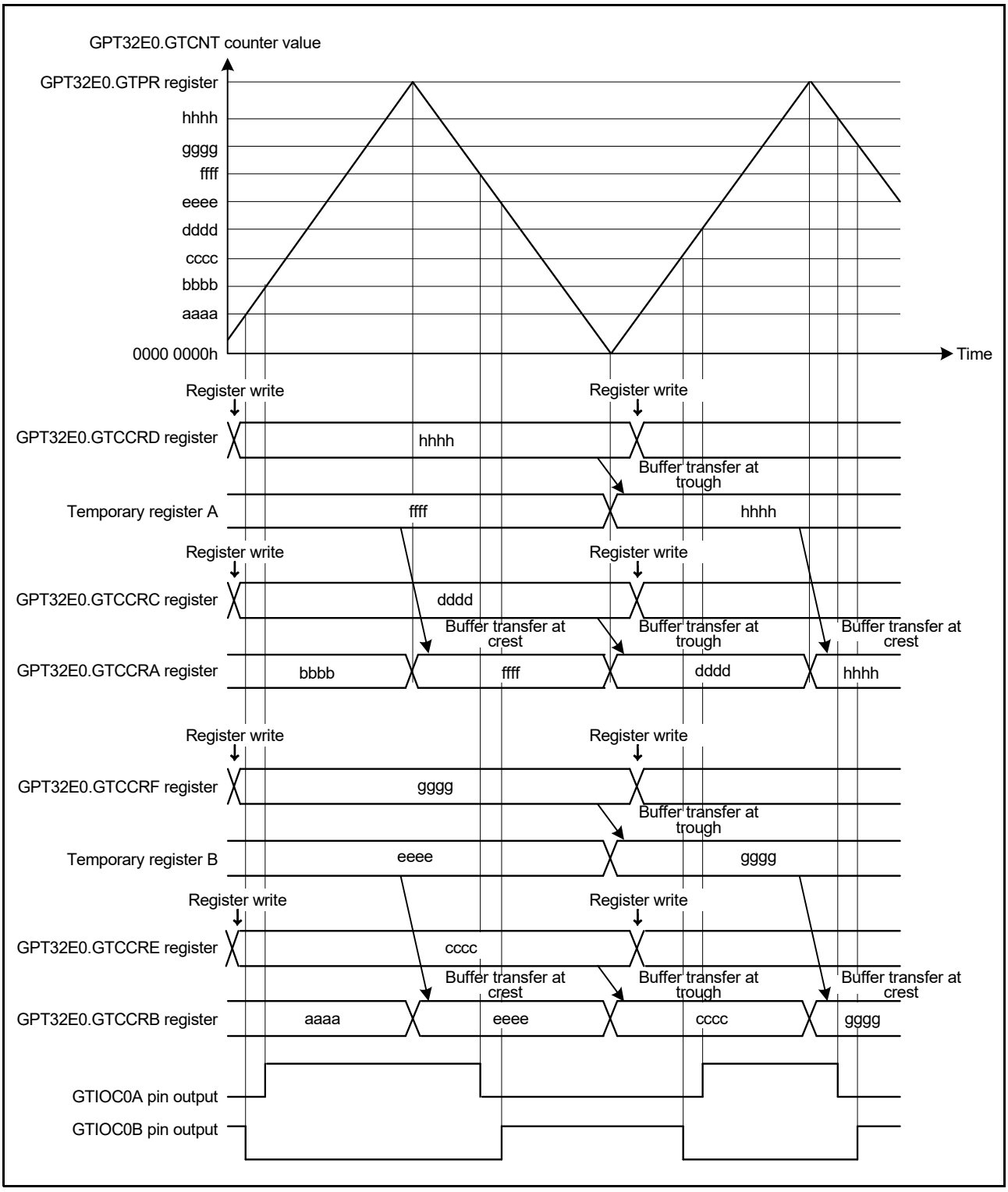


Figure 12.40 Example of triangle-wave PWM mode 3 operation with low output from the GTIOC0A pin and high output from the GTIOC0B pin at count start, output toggled at GTCCRA/GTCCRB compare match, and output retained at cycle end

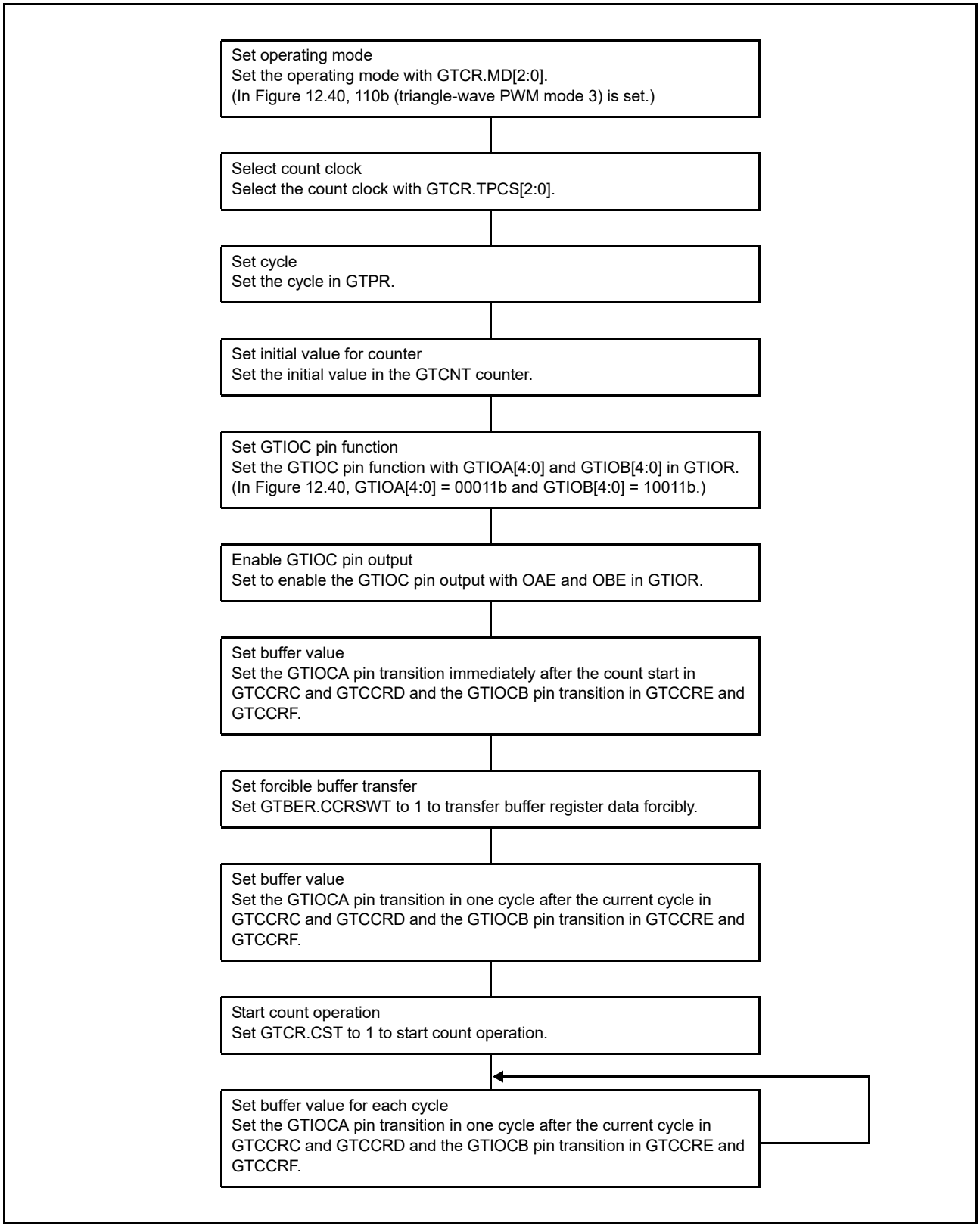


Figure 12.41 Example setting for triangle-wave PWM mode 3

12.3.4 Automatic Dead Time Setting Function

By setting GTDTCR, a compare match value for a negative waveform with dead time obtained by a compare match value for a positive waveform (GTCCRA value) and specified dead time values (GTDVU and GTDVD values) can automatically be set to GTCCRB. The automatic dead time setting function can be used in saw-wave one-shot pulse mode and all the triangle PWM modes.
Dead time can be separately set for the first half and second half of a waveform. Dead time for the transition in the first half of a negative waveform is set in GTDVU and that in the second half is set in GTDVD. The same dead time can also be set for the first and second halves by setting the GTDTCR.TDFER bit to 1.
GTDBU can be used as a buffer register of GTDVU, and GTDBD can be used as a buffer register of GTDVD.
In the saw-wave mode, buffer transfer is performed when the GTCND counter overflows (up-counting), underflows (down-counting), or is cleared. In the triangle-wave mode, transfer proceeds at troughs.

The change point of the negative phase waveform with the automatic dead time setting can be confirmed by reading the GTCCRB register value.
Writing to the GTCCRB register is prohibited when using the automatic dead time setting function.
It is prohibited to set dead time such that the change point of the waveform exceeds the count cycle.
If there is an error condition on the dead time setting, be able to generate waveforms that secured dead time by correcting the transition points of the positive and negative phase waveforms, as shown in Table 12.7.

The transition point of the corrected negative-phase waveform is automatically set in the GTCCRB register. Since the internal signal is used to judge the transition point of the positive phase waveform, the GTCCRA register is not updated with the corrected value.
By correcting the waveform transition point due to a dead time error on the saw-wave one-shot pulse mode, if the order of waveform transition points is disturbed or it exceeds the count cycle period after correction, the complementary relationship between the positive-phase waveform and the negative-phase waveform is not guaranteed.
By setting the GTCCRA equal 0 or grater than or equal to the GTPR of the GTCCRA register in the triangle-wave PWM mode, if the dead time setting exceeds the count cycle period, output transition is suppressed by the output protection function. When GTCCRA ≥ GTPR + GTDVn, GTPR - 1 is set as the upper limit value in the GTCCRB register (Refer to section 12.8.4, Output Protection Function for GTIOC Pin Output).
When GTCCRA ≥ GTPR + GTDVn, GTPR - 1 is set as the upper limit value in the GTCCRB register.

Table 12.7 Correction of waveform transition point at dead time error occurrence

Wave mode	Count direction	Interval	Dead time error condition	Positive-Phase waveform transition point with corrected	Negative-phase waveform transition point with corrected
saw-wave one-shot pulse mode	Up counting	first half	GTCCRA-GTDVU < 0	GTDVU	0
		second half	GTCCRA-GTDVD > GTPR	GTPR-GTDVD	GTPR
	Down counting	first half	GTCCRA-GTDVU > GTPR	GTPR-GTDVU	GTPR
		second half	GTCCRA-GTDVD < 0	GTDVD	0
triangle PWM mode 1/2/3	Up counting	(first half)	GTCCRA-GTDVU ≤ 0	GTDVU+1	1
	Down counting	(second half)	GTCCRA-GTDVD < 0	GTDVD	0

Values for automatic dead time setting can be read from GTCCRB. The automatic dead time value setting to GTCCRB is performed at the next count clock cycle when registers that are used for calculating the automatic dead time value are updated.

The method of writing a new value to a GTDVn register depends on whether buffer operation is enabled or disabled.
When GTDVn buffer operation is enabled: The GTDBn register can be written at any time. The value in the GTDBn register is transferred to the GTDVn register at the cycle end.

When GTDVn buffer operation is disabled: Set the GTCR.CST bit to stop the GPT before changing the value of the

GTDVn register.

Figure 12.42 to Figure 12.45 show examples of automatic dead time setting function operation. Figure 12.46 and Figure 12.47 show the setting examples.

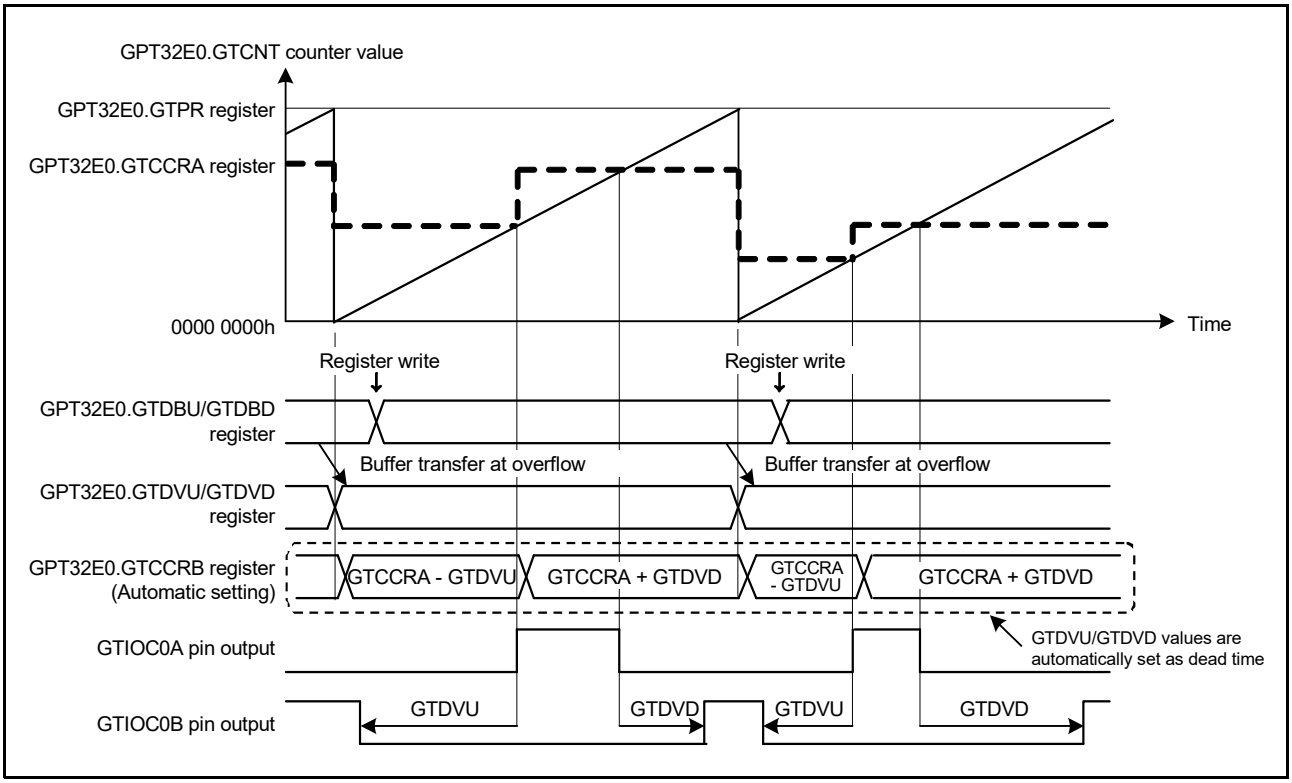


Figure 12.42 Example of automatic dead time setting function operation with saw-wave one-shot pulse mode, up-counting, GTDVU and GTDVD set to buffer operation, and active-high

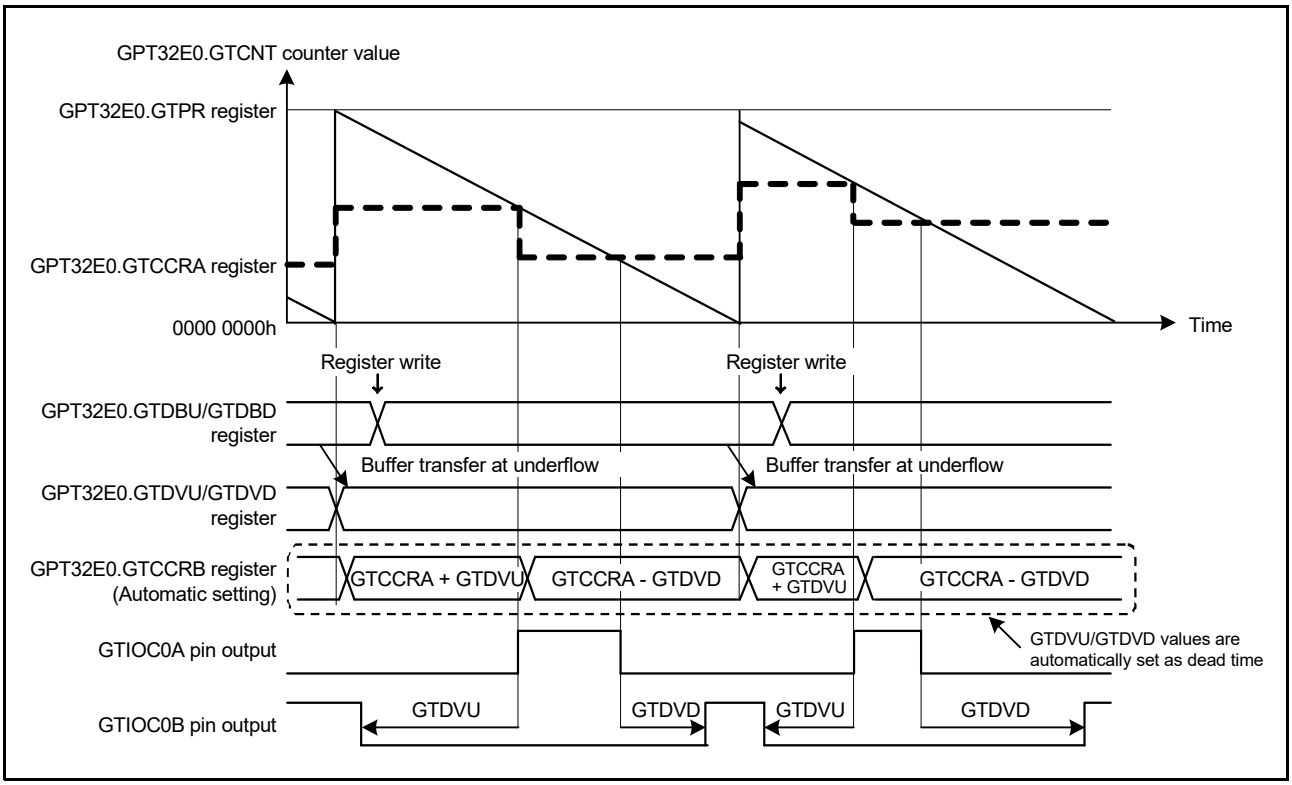


Figure 12.43 Example of automatic dead time setting function operation with saw-wave one-shot pulse mode, down-counting, GTDVU and GTDVD set to buffer operation, and active-high

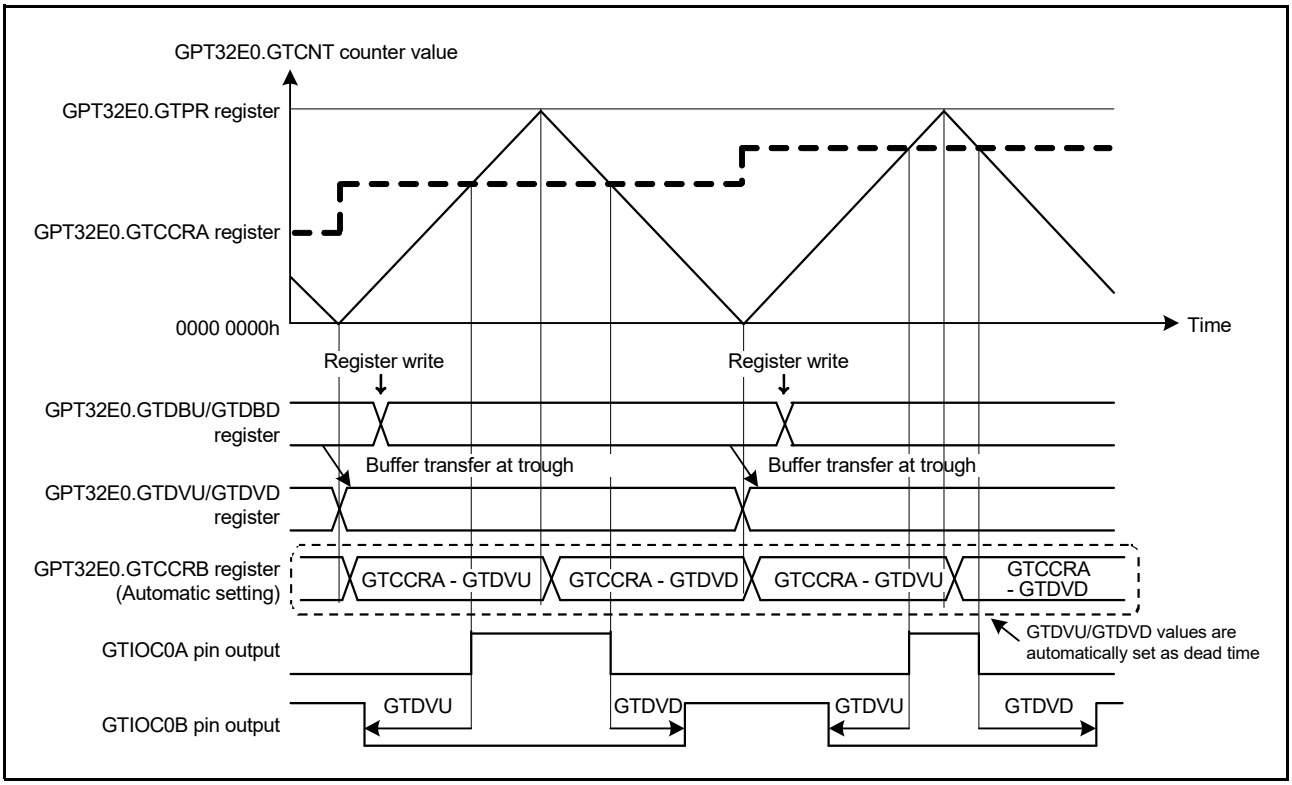


Figure 12.44 Example of automatic compare-match value setting function with dead time with triangle-wave PWM mode 1, GTDVU and GTDVD set to buffer operation, active-high

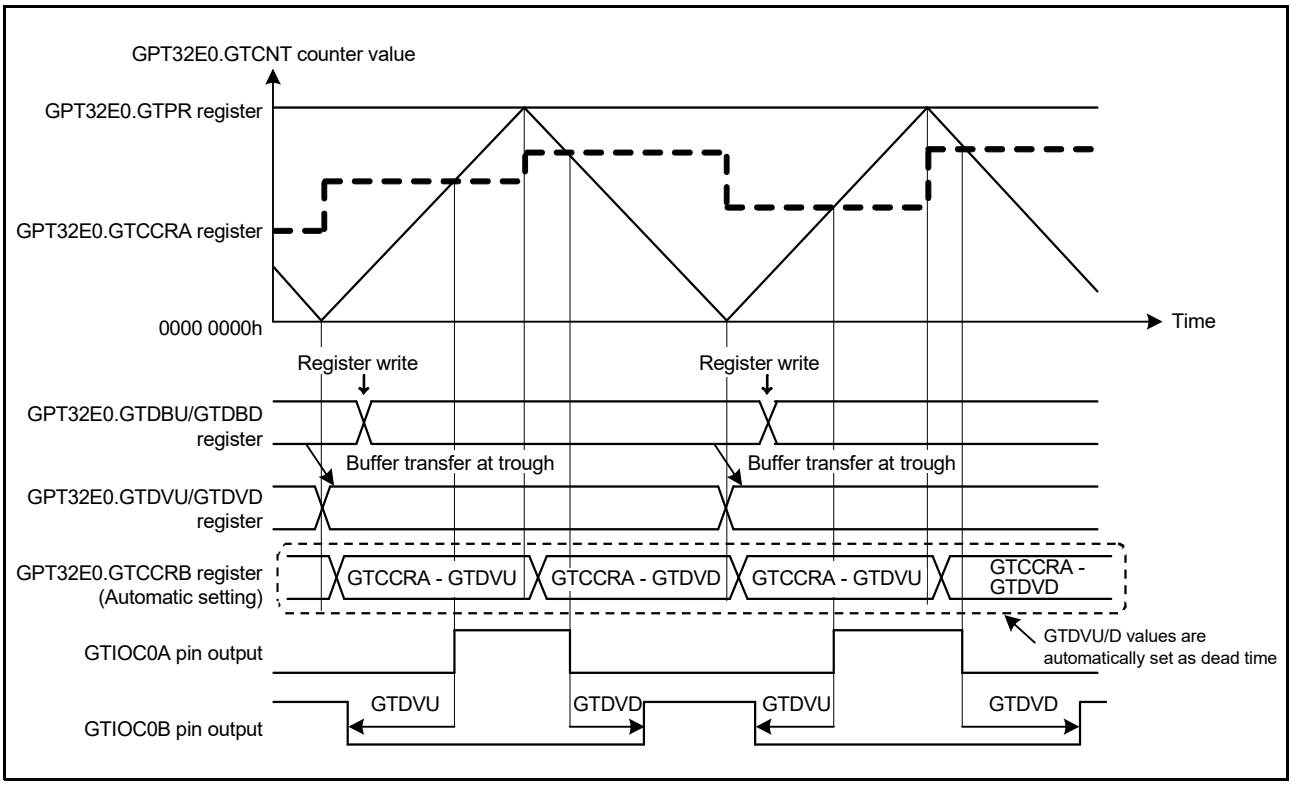


Figure 12.45 Example of automatic compare-match value setting function with dead time, with triangle-wave PWM mode 2 or 3, GTDVU and GTDVD set to buffer operation, and active-high

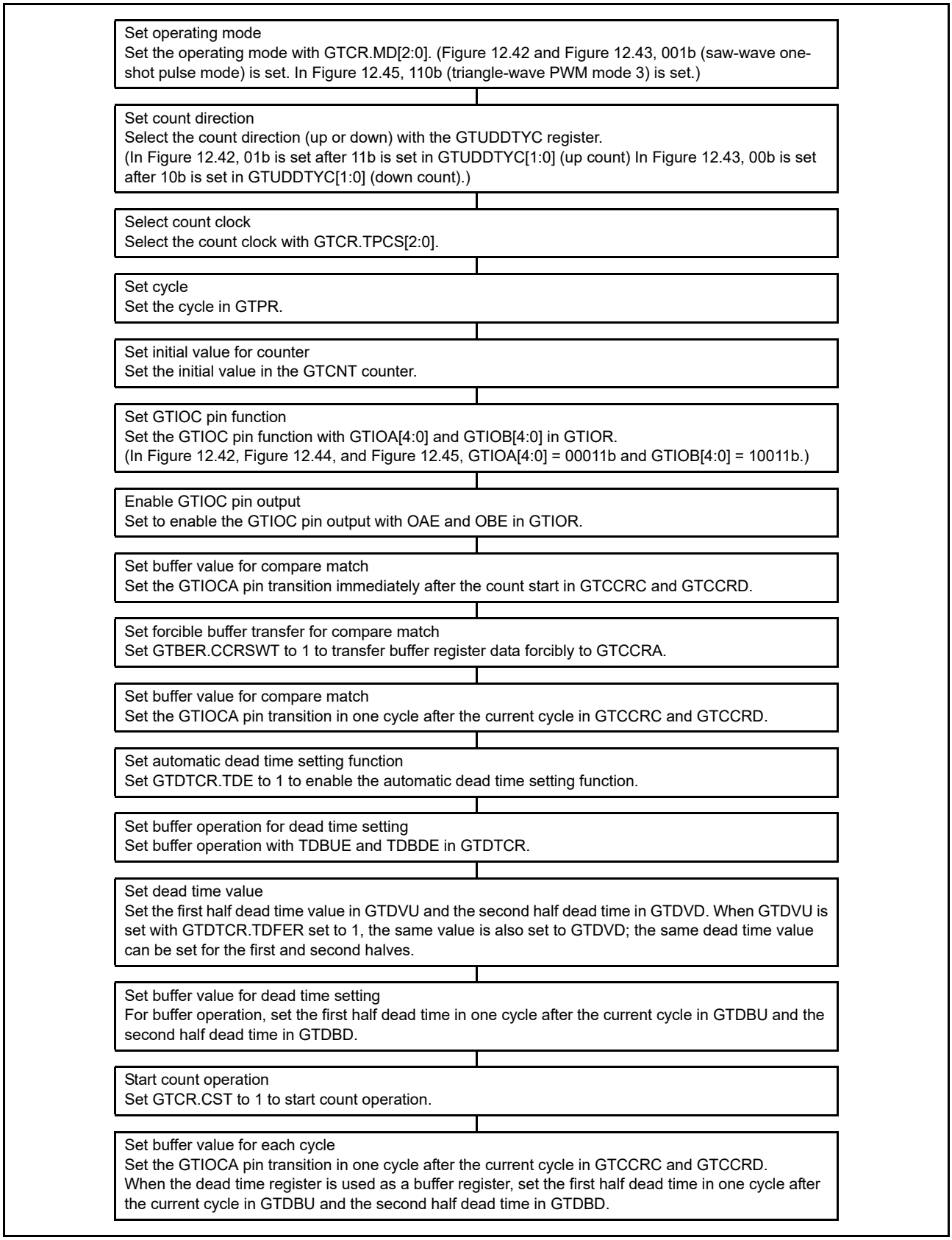


Figure 12.46 Example setting for automatic dead time setting function with saw-wave one-shot pulse mode, and triangle-wave PWM mode 3

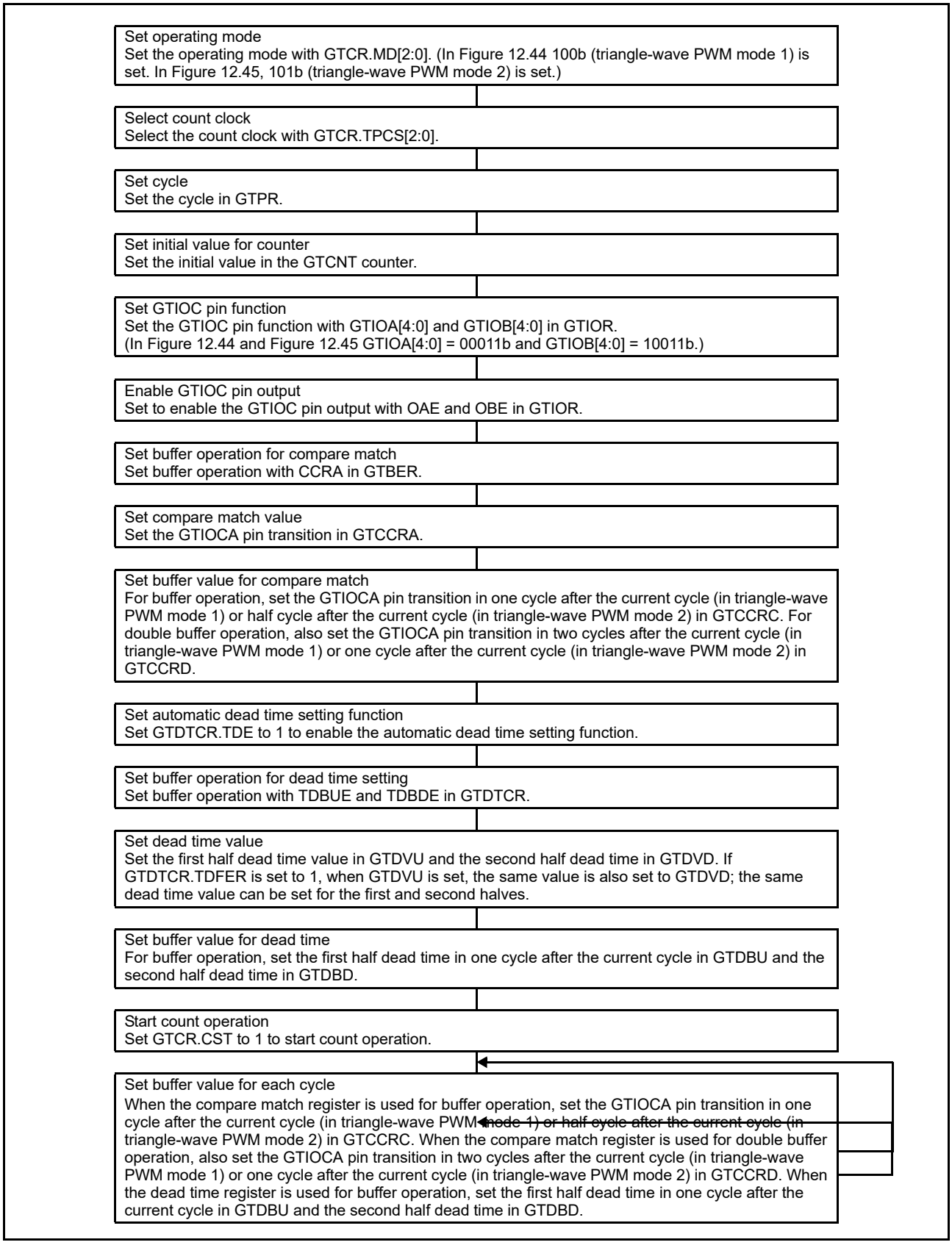


Figure 12.47 Example setting for automatic dead time setting function with triangle-wave PWM mode 1 or 2

12.3.5 Count Direction Changing Function

The count direction of the GTCNT counter can be changed by modifying the UD bit in GTUDDTYC.
In saw-wave mode, if the UD bit in GTUDDTYC is modified during count operation, the count direction is changed at an overflow (when modified during up-counting) or an underflow (when modified during down-counting). If the GTUDDTYC.UD bit is modified while the count operation stops and the GTUDDTYC.UDF bit is 0, the GTUDDTYC.UD bit modification is not reflected at the start of counting and the count direction is changed at an overflow or an underflow. If the UDF bit is set to 1 while the count operation stops, the GTUDDTYC.UD bit value at that time is reflected at the start of counting.
In triangle-wave mode, the count direction does not change even though the UD bit in GTUDDTYC is modified during the count operation. Similarly, even though the GTUDDTYC.UD bit is modified while the count operation stops and GTUDDTYC.UDF bit is 0, the GTUDDTYC.UD bit value is not reflected to the count operation. If the GTUDDTYC.UDF bit is set to 1 while the count operation stops, the GTUDDTYC.UD bit value at that time is reflected at the start of counting.
If the count direction changes during a saw-wave count operation, the GTPR value after the start of up-counting is reflected to the count cycle during up-counting and the GTPR value before the start of down-counting is reflected during down-counting.
Figure 12.48 shows an example of count direction changing function operation.

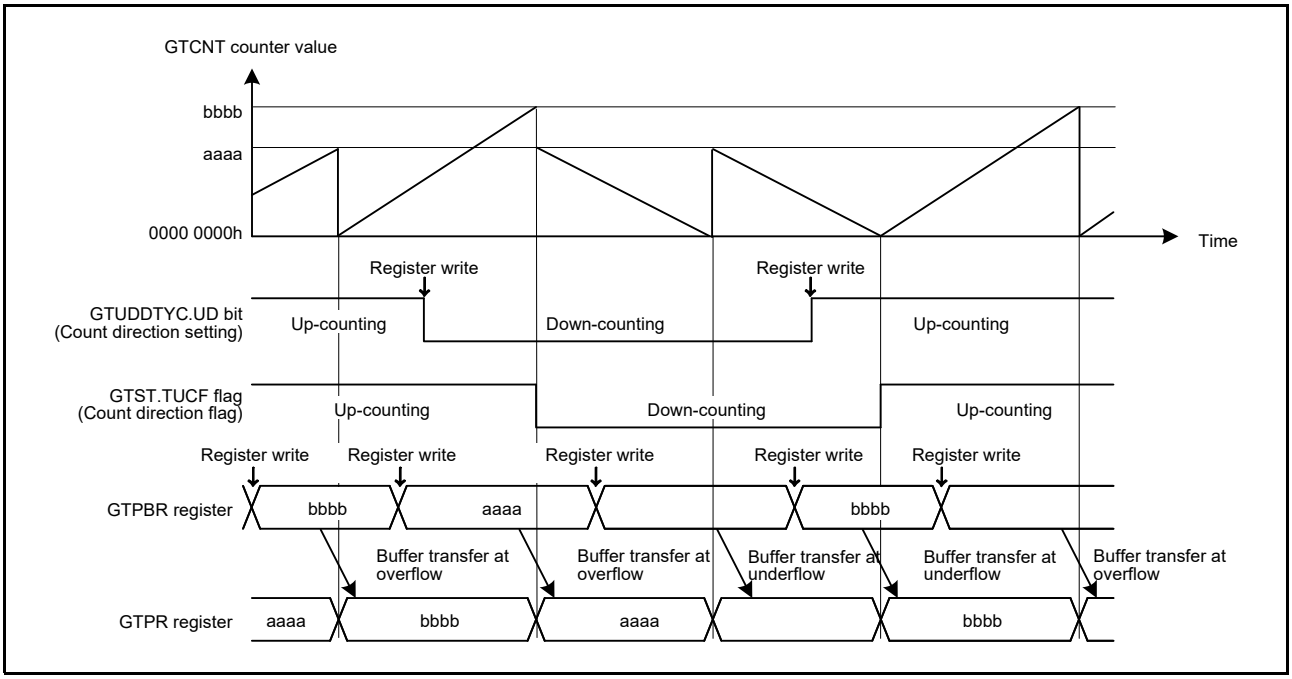


Figure 12.48 Example of count direction changing function operation during buffer operation

12.3.6 Function of Output Duty 0% and 100%

The output duty of the GTIOCA pin and the GTIOCB pin are set to 0% or 100% by changing the GTUDDTYC.OADTY bit or GTUDDTYC.OBDTY bit.

In saw-wave mode, if the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified during the count operation, the output duty setting is reflected at an overflow (when modified during up-counting) or an underflow (when modified during down-counting). If the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified while the count operation stops and the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is 0, the output duty modification is not reflected at the start of counting. The output duty changes at an overflow or an underflow. If the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is set to 1 while the count operation stops, the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit value at that time is reflected at the start of counting.

In triangle-wave mode, if the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified during the count operation, the output duty setting is reflected an underflow.

If the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified while the count operation stops and the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is 0, the output duty modification is not reflected at the start of counting. The output duty changes at an underflow. If the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified while the count operation stops and the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is 1, the output duty modification is reflected at the start of counting.

In performing 0%/100% duty operation, GPT internally continues to:

- Perform compare match operation
- Set compare match flag
- Output interrupt
- Perform buffer operation.

When the control is changed from 0% or 100% duty setting to compare match, the output value of GTIOCA pin at cycle end is decided by GTIOR.GTIOA[3:2] and GTUDDTYC.OADTYR. The output value of GTIOCB pin at cycle end is decided by GTIOR.GTIOB[3:2] and GTUDDTYC.OBDTYR.

When GTIOR.GTIOA[3:2] and GTIOR.GTIOB[3:2] are set to 01b, the output pins output low at cycle end. When GTIOR.GTIOA[3:2] and GTIOR.GTIOB[3:2] are set to 10b, the output pins output high at cycle end.

GTUDDTYC.OADTYR selects the value that is the object of output retained/toggled at cycle end, when GTIOR.GTIOm[3:2] are set to 00b (output retained at cycle end) or when GTIOR.GTIOm[3:2] are set to 11b (output toggled at cycle end). Table 12.8 shows the values of GTIOCA/GTIOCB pin output at cycle end.

Table 12.8 Output values after releasing 0% or 100% duty setting (m = A, B)

GTIOR.GTIOm[3:2]	Compare match value at cycle end masked by 0% or 100% duty setting	GTUDDTYC.OmDTYR in duty 0% setting		GTUDDTYC.OmDTYR in duty 100% setting	
		0	**1**	**0**	**1**
00 (Output retained at cycle end)	0	0	0	1	0
	1	0	1	1	1
01 (low output at cycle end)	—	0	0	0	0
10 (high output at cycle end)	—	1	1	1	1
11 (Output toggled at cycle end)	0	1	1	0	1
	1	1	0	0	0

Figure 12.49 shows the example of output duty 0% and 100% functions.

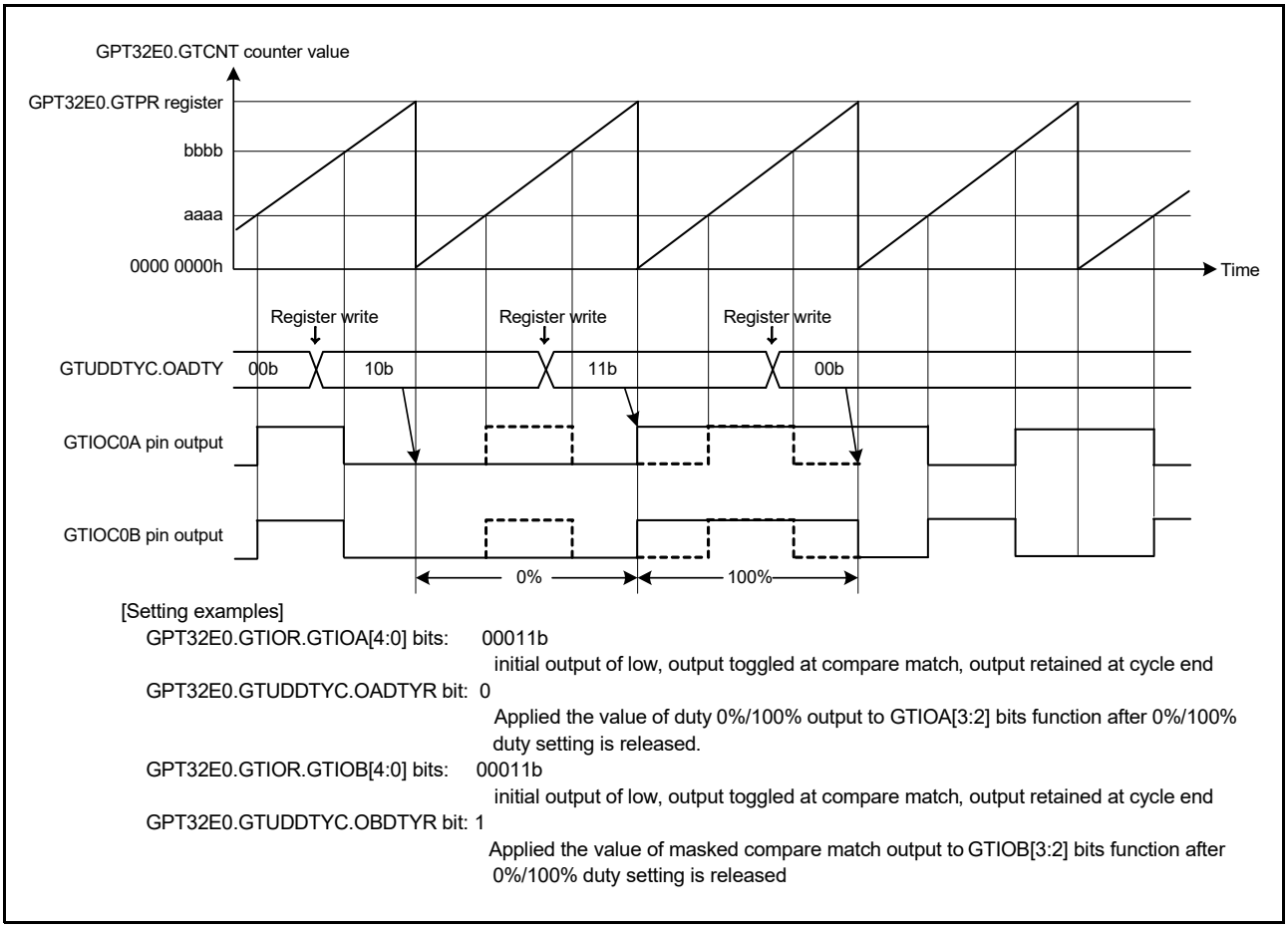


Figure 12.49 Example of output duty 0% and 100% functions

12.3.7 Hardware Count Start/Count Stop and Clear Operation

The GTCNT counter can be started, stopped, or cleared by the following hardware sources:

- External trigger input
- Event input
- GTIOCA/GTIOCB pin input.

12.3.7.1 Hardware Start Operation

The GTCNT counter can be started by selecting a hardware source using GTSSR.
Figure 12.50 shows an example of a count start operation by a hardware source. Figure 12.51 shows the setting example.

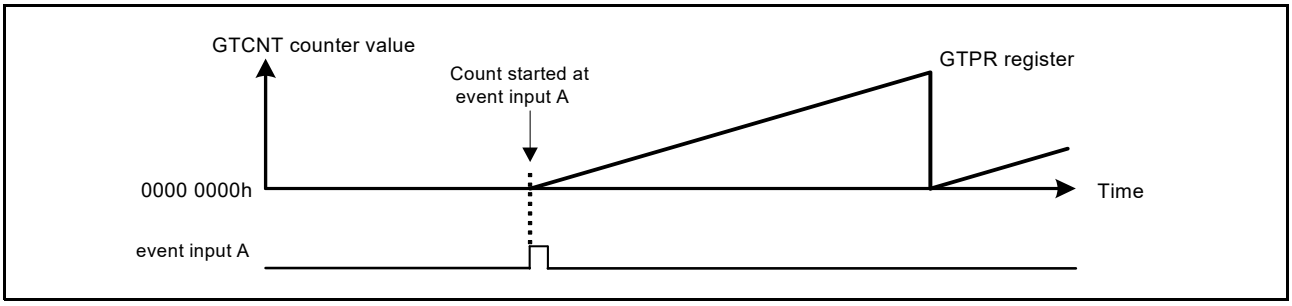


Figure 12.50 Example of count start operation by hardware source, started at the input of the signal from the event A

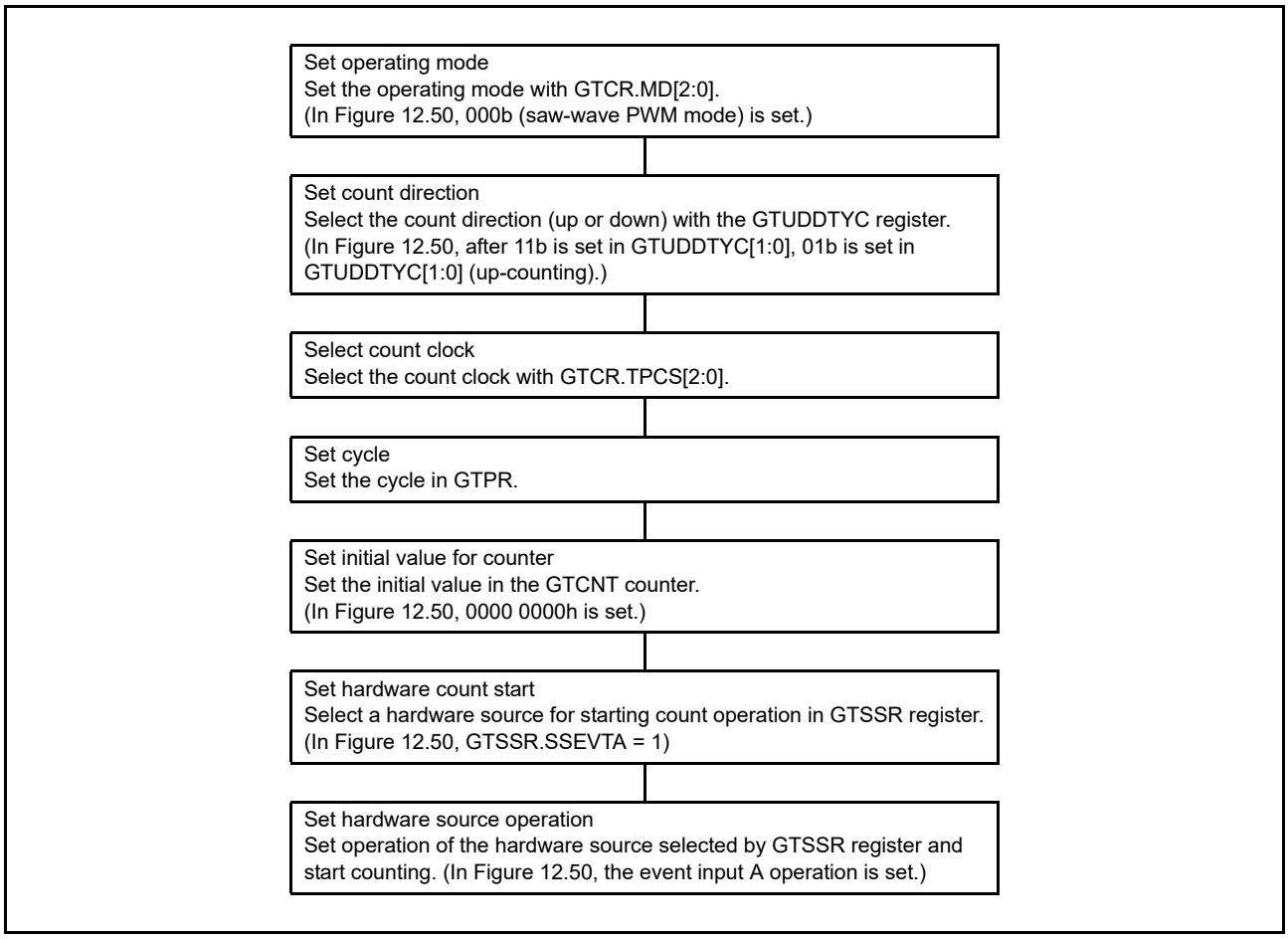


Figure 12.51 Example setting for count start operation by hardware source

12.3.7.2 Hardware Stop Operation

The GTCNT counter can be stopped by selecting a hardware source using GTPSR. Figure 12.52 shows an example of a count stop operation by a hardware source. Figure 12.53 shows the setting example. In this example, the count operation stops at the edge of the event input A and restarts at the edges of the event input B.

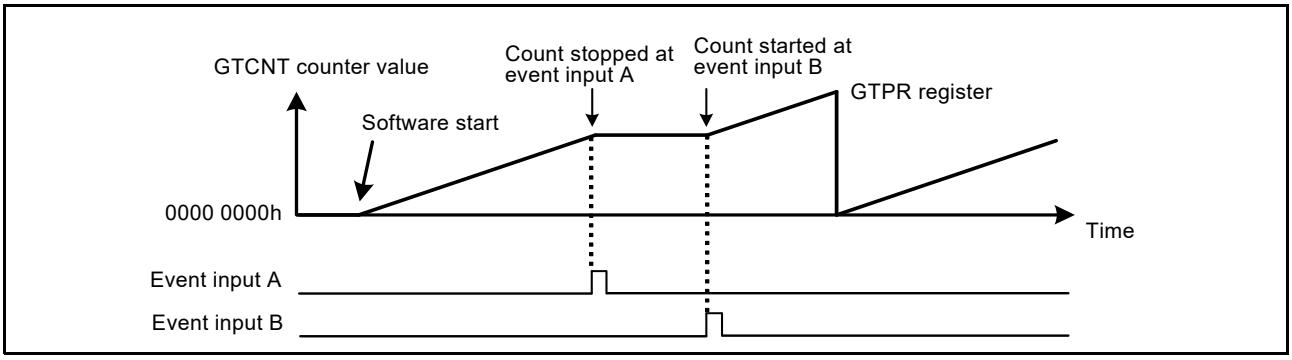


Figure 12.52 Example of count stop operation by hardware source, started by software, stopped at event input A, and restarted at event input B

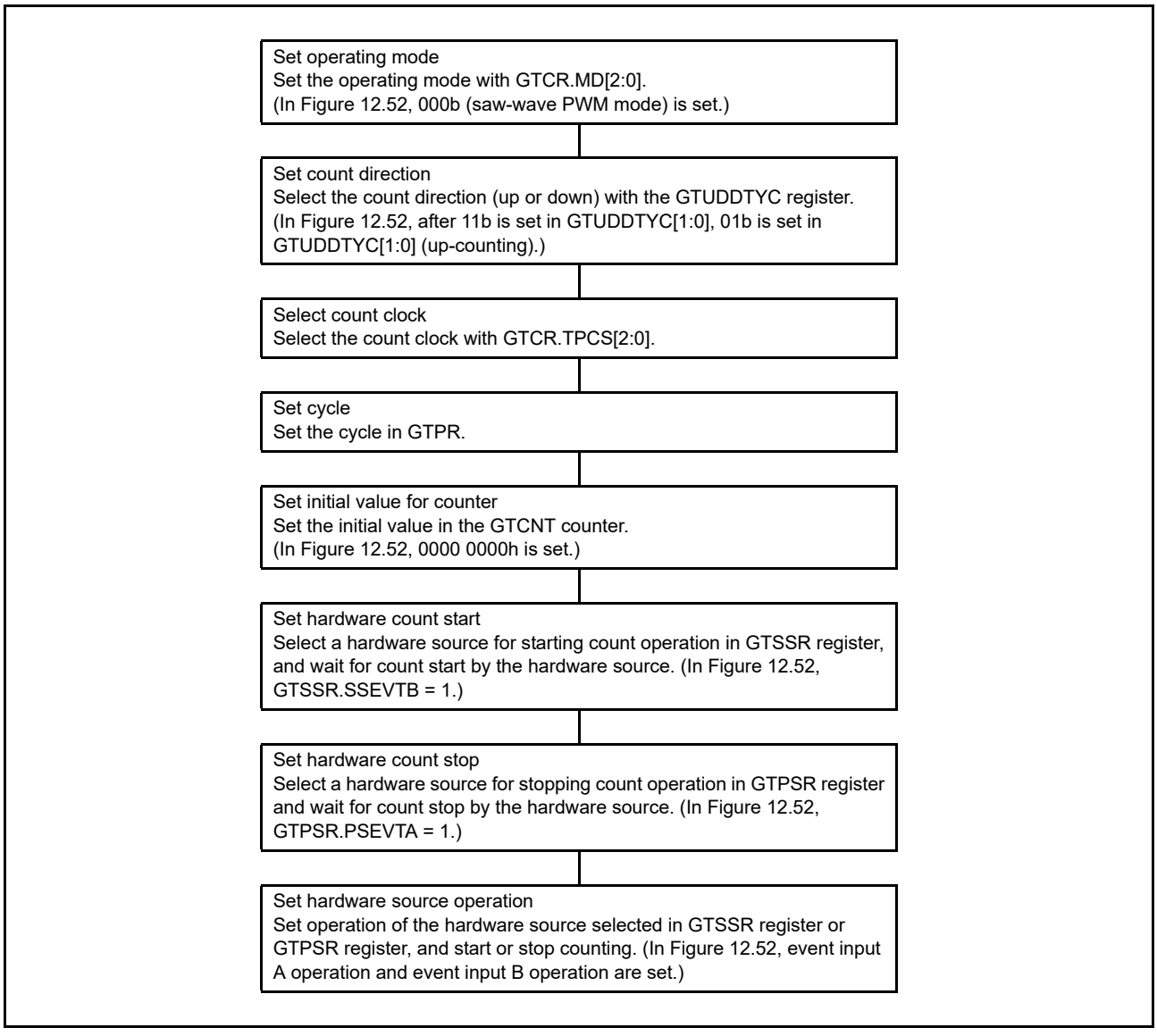


Figure 12.53 Example setting for count stop operation by hardware source

Figure 12.54 shows an example of a count start/stop operation by a hardware source. Figure 12.55 shows the setting example. In this example, the counter operates during the high-level periods of the external trigger input GTETRGA.

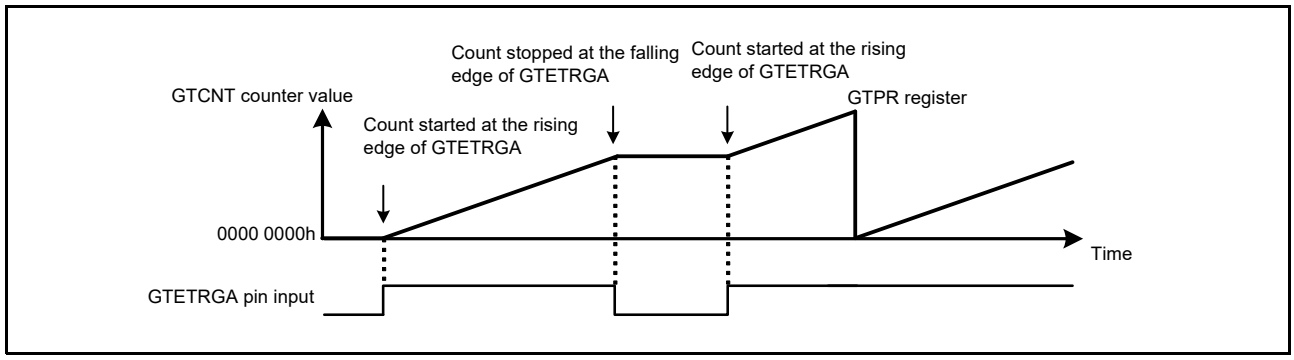


Figure 12.54 Example of count start/stop operation by hardware source, started on the rising edge of the GTETRGA pin input and stopped on the falling edge of the GTETRGA pin input

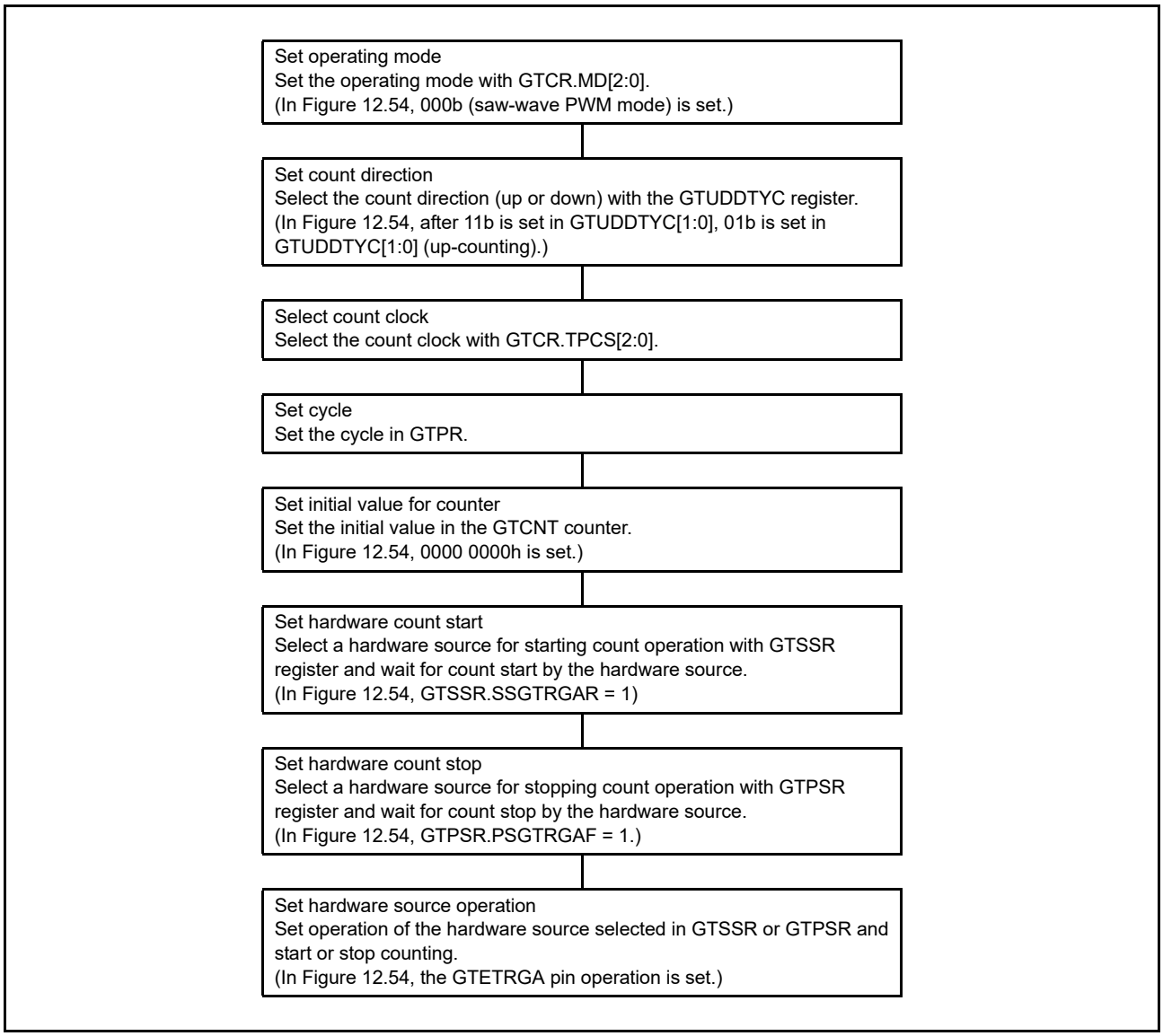


Figure 12.55 Example setting for count start/stop operation by hardware source

12.3.7.3 Hardware Clear Operation

The GTCNT counter can be cleared by selecting a hardware source using GTCSR. The OVFn/UNFn (n=0 to 7) interrupt (overflow/underflow interrupt) is not generated when the GTCNT counter is cleared by a hardware source or by software.

Figure 12.56 and Figure 12.57 show examples of the GTCNT counter clearing operation by a hardware source. Figure 12.58 shows the setting example. In this example, the GTCNT counter starts at the edge of the event input A, and the counter stops/clears at the edge of the event input B.

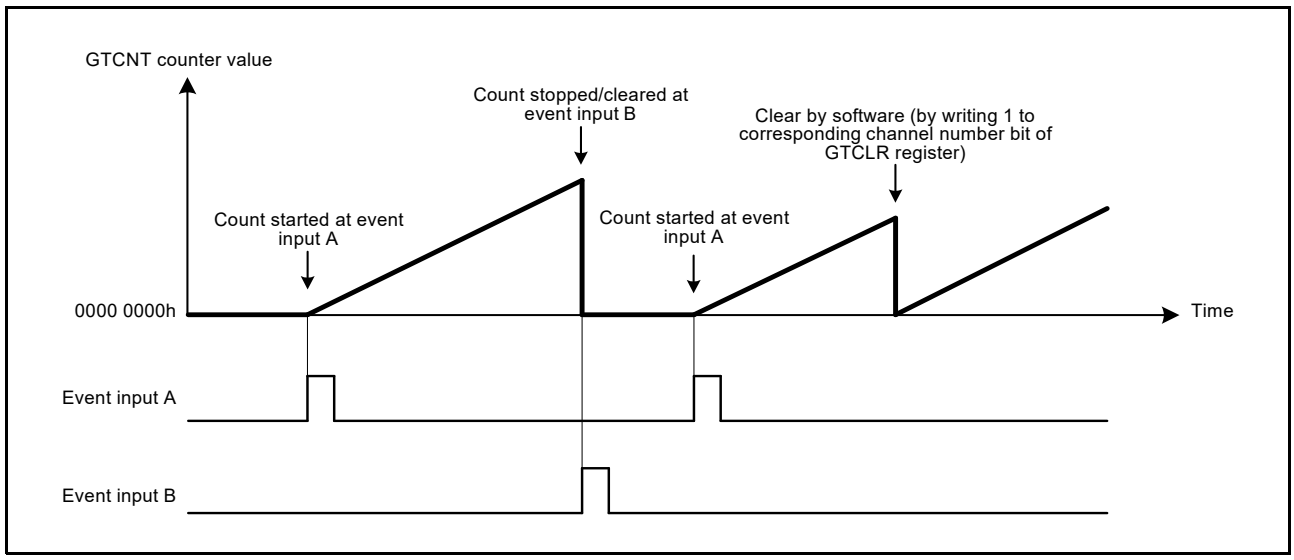


Figure 12.56 Examples of count clearing operation by hardware source with saw wave up-counting, started at event input A, and stopped/cleared at event input B

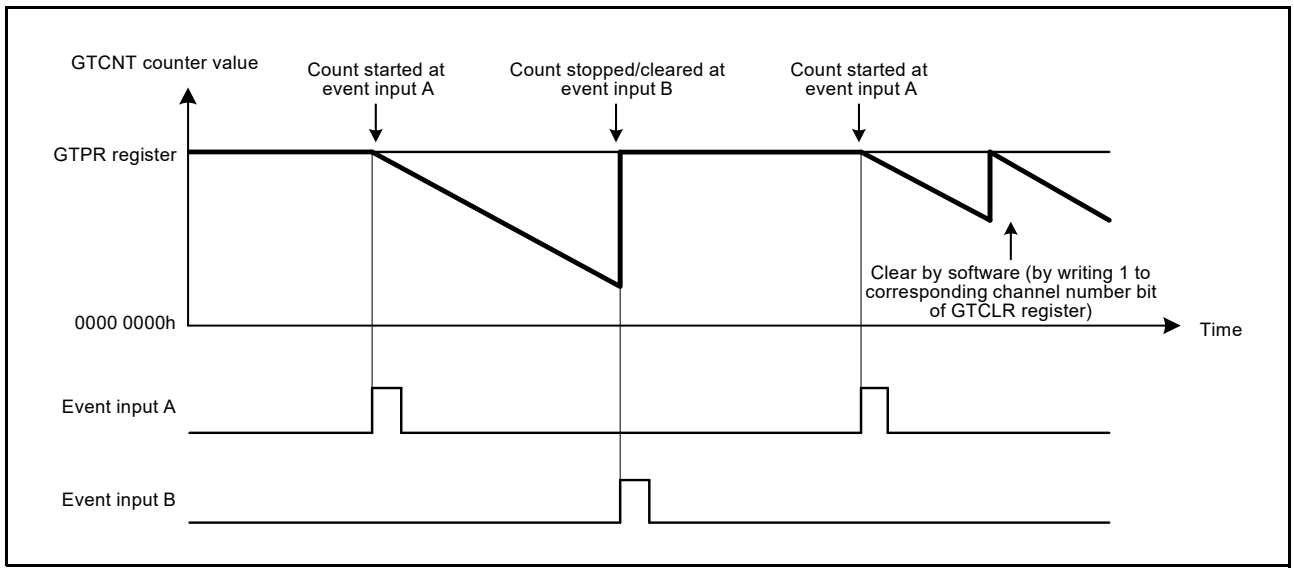


Figure 12.57 Examples of count clearing operation by hardware source with saw wave down-counting, started at event input A, and stopped/cleared at event input B

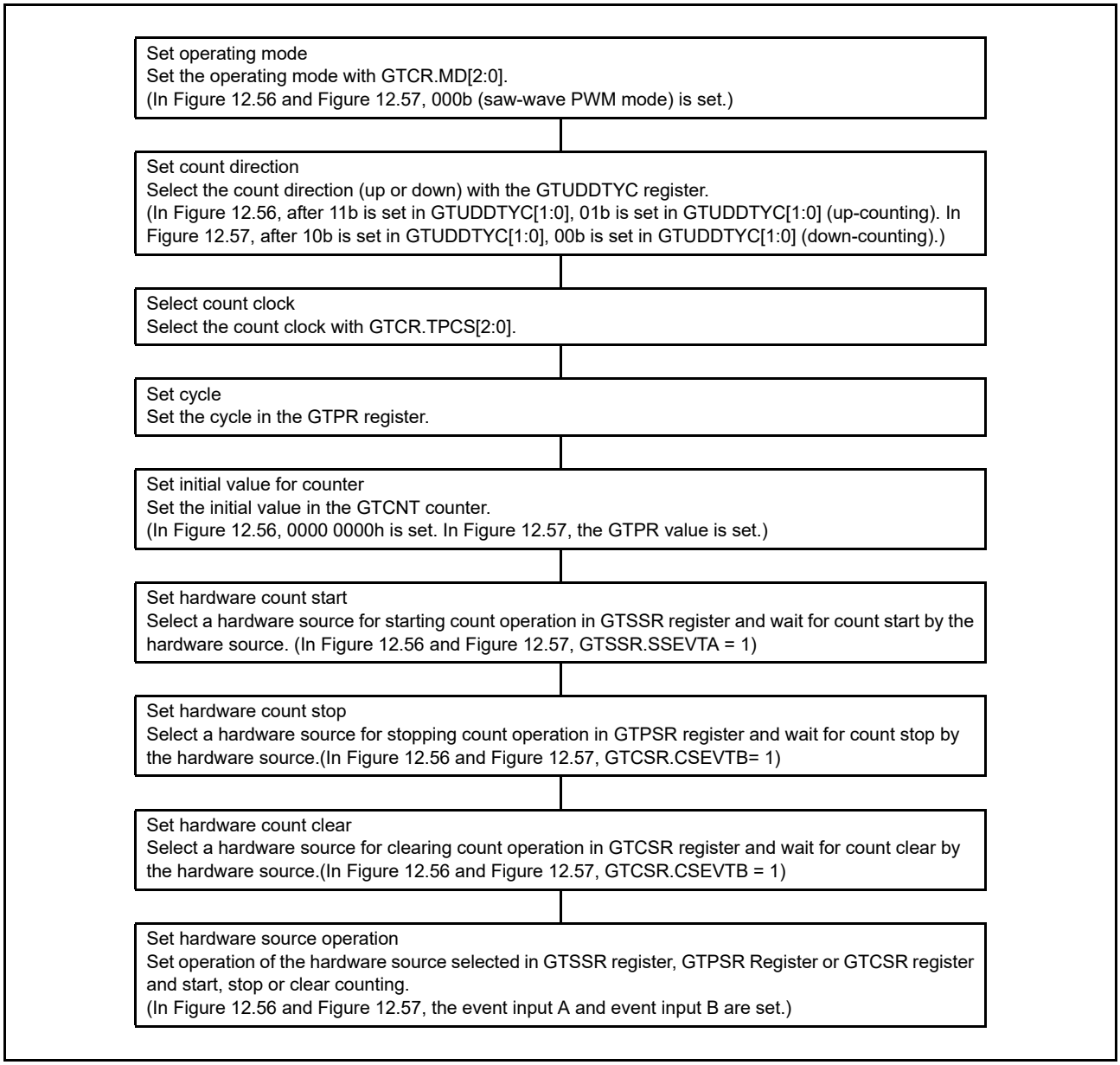


Figure 12.58 Example setting for count clearing operation by hardware source

The OVFn/UNFn (n=0 to 7) interrupt (overflow/underflow interrupt) is not generated when the counter is cleared by a hardware source or by software.

Figure 12.59 shows the relationship between the counter clearing by a hardware source and the OVFn (n=0 to 7) interrupt.

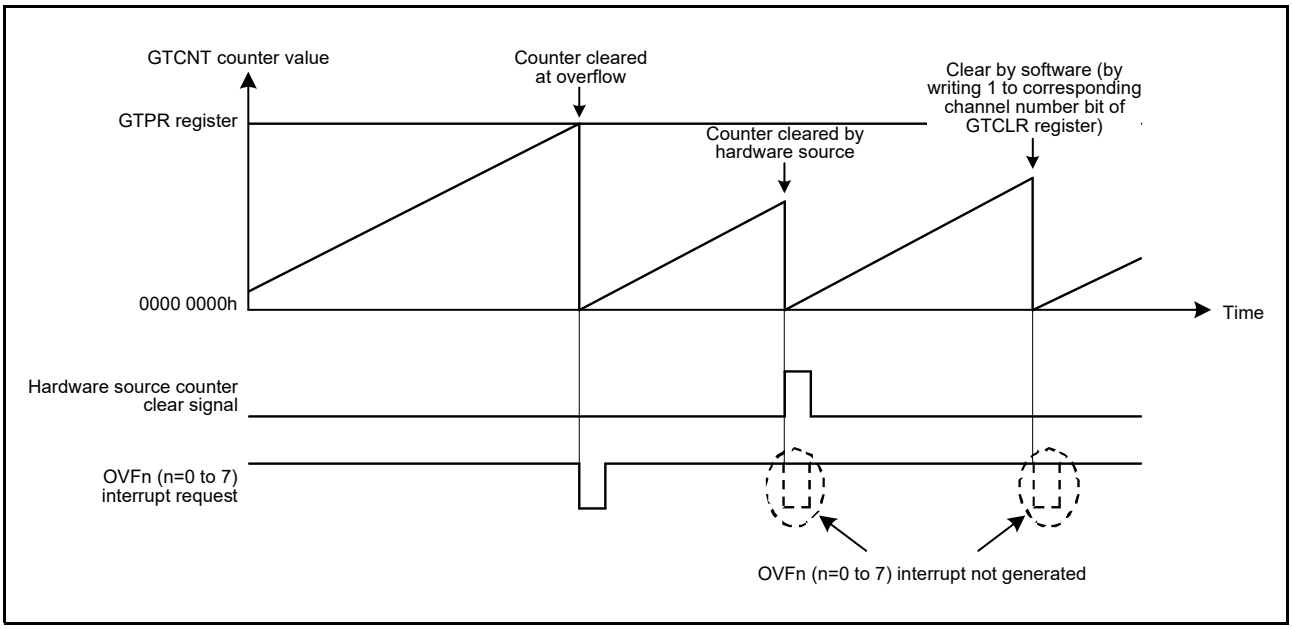


Figure 12.59 Relationship between counter clearing by hardware source and OVFn (n = 0 to 7) interrupt

12.3.8 Synchronized Operation

Synchronized operation on channels such as a synchronized start, stop and clear operation can be performed.

12.3.8.1 Synchronized Operation by Software

The GTCNT counters can be started, stopped, and cleared on multiple channels by setting the associated GTSTR, GTSTP or GTCLR bits simultaneously to 1.
Count start with a phase difference is possible by setting the initial value in the GTCNT counter and setting the associated GTSTR bits simultaneously to 1.
Figure 12.60 shows an example of a simultaneous start, stop and clear by software. Figure 12.61 shows an example of phase start operation by software.

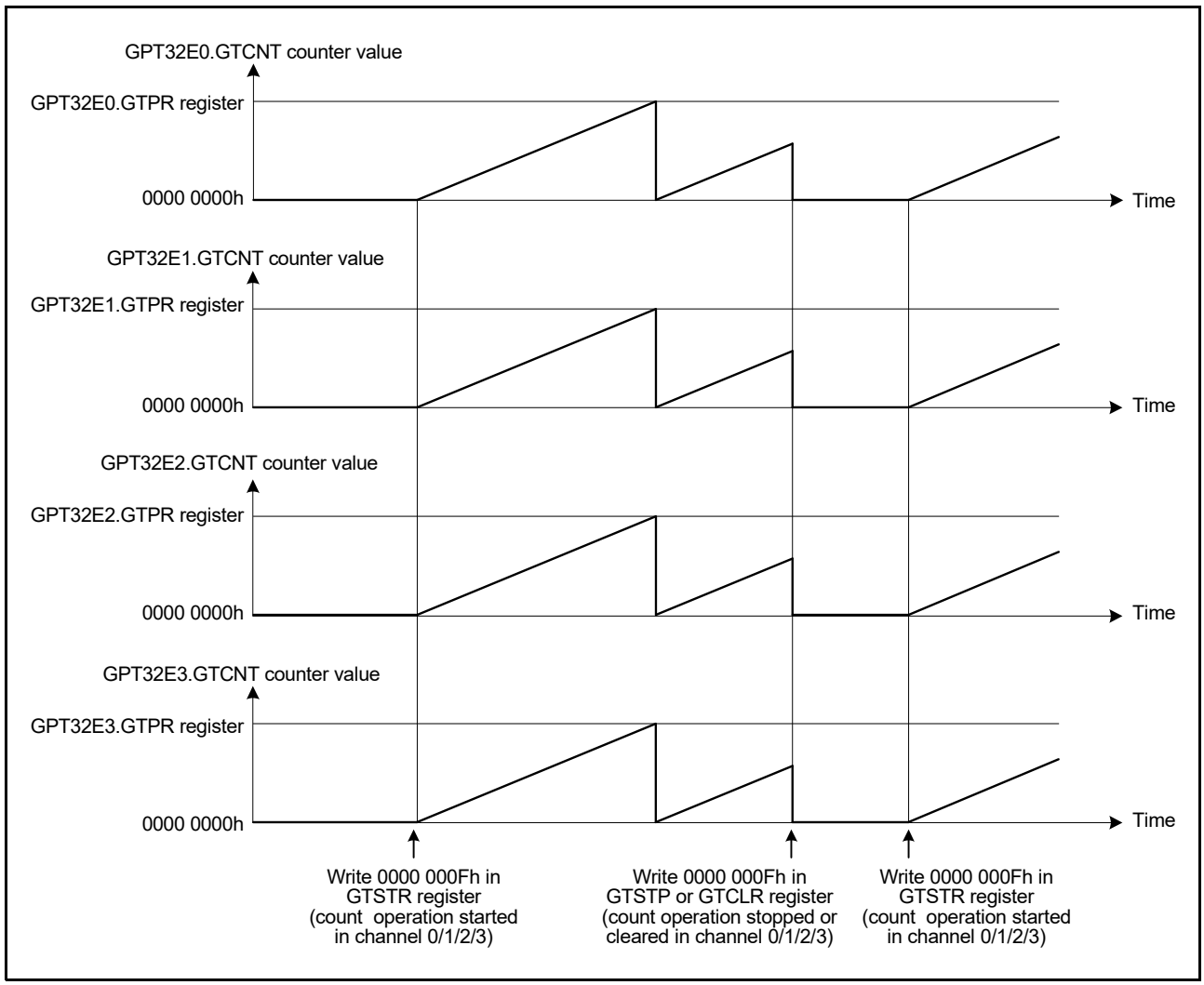


Figure 12.60 Example of a simultaneous start, stop, and clear by software, with the same count cycle (GTPR register value)

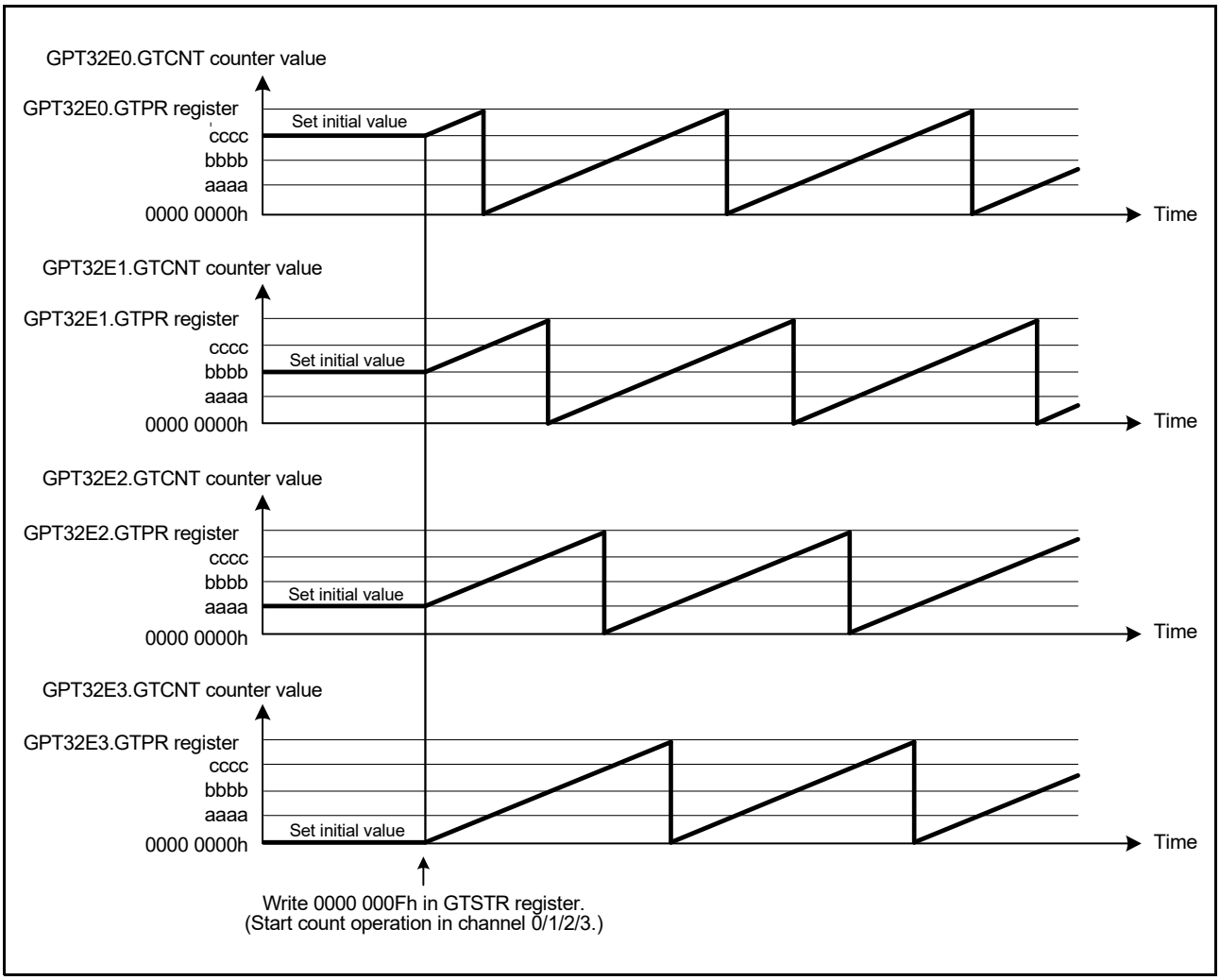


Figure 12.61 Example of software phase start with the same count cycle (GTPR register value)

12.3.8.2 Synchronized Operation by Hardware

The GTCNT counters can be started simultaneously by the following hardware sources:

- External trigger input
- Event input

Figure 12.62 shows an example of a simultaneous start, stop and clear operation by a hardware source. Figure 12.63 shows the setting example.

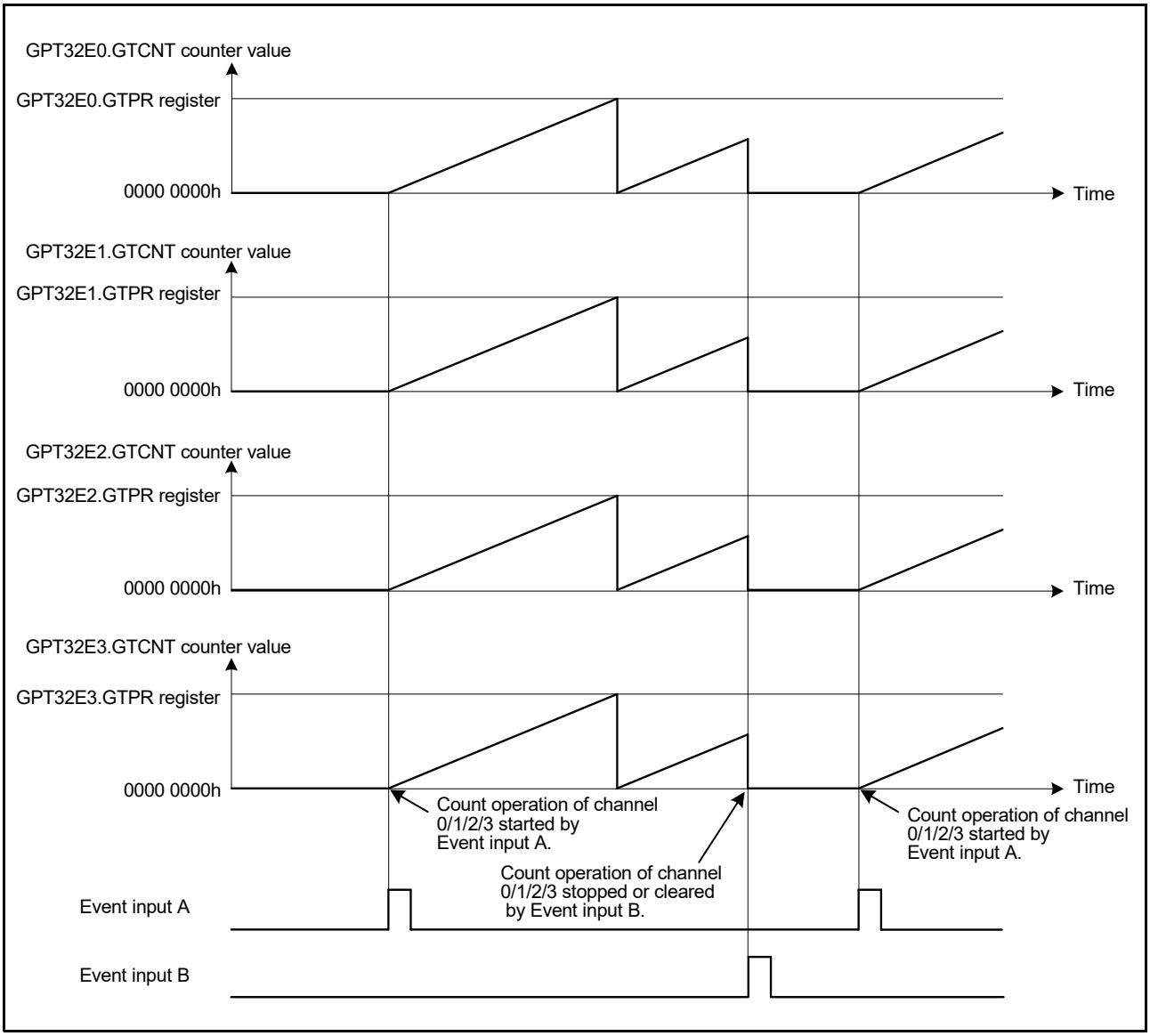


Figure 12.62 Example of a simultaneous start, stop, and clear by the hardware sources, with the same count cycle (GTPR register value)

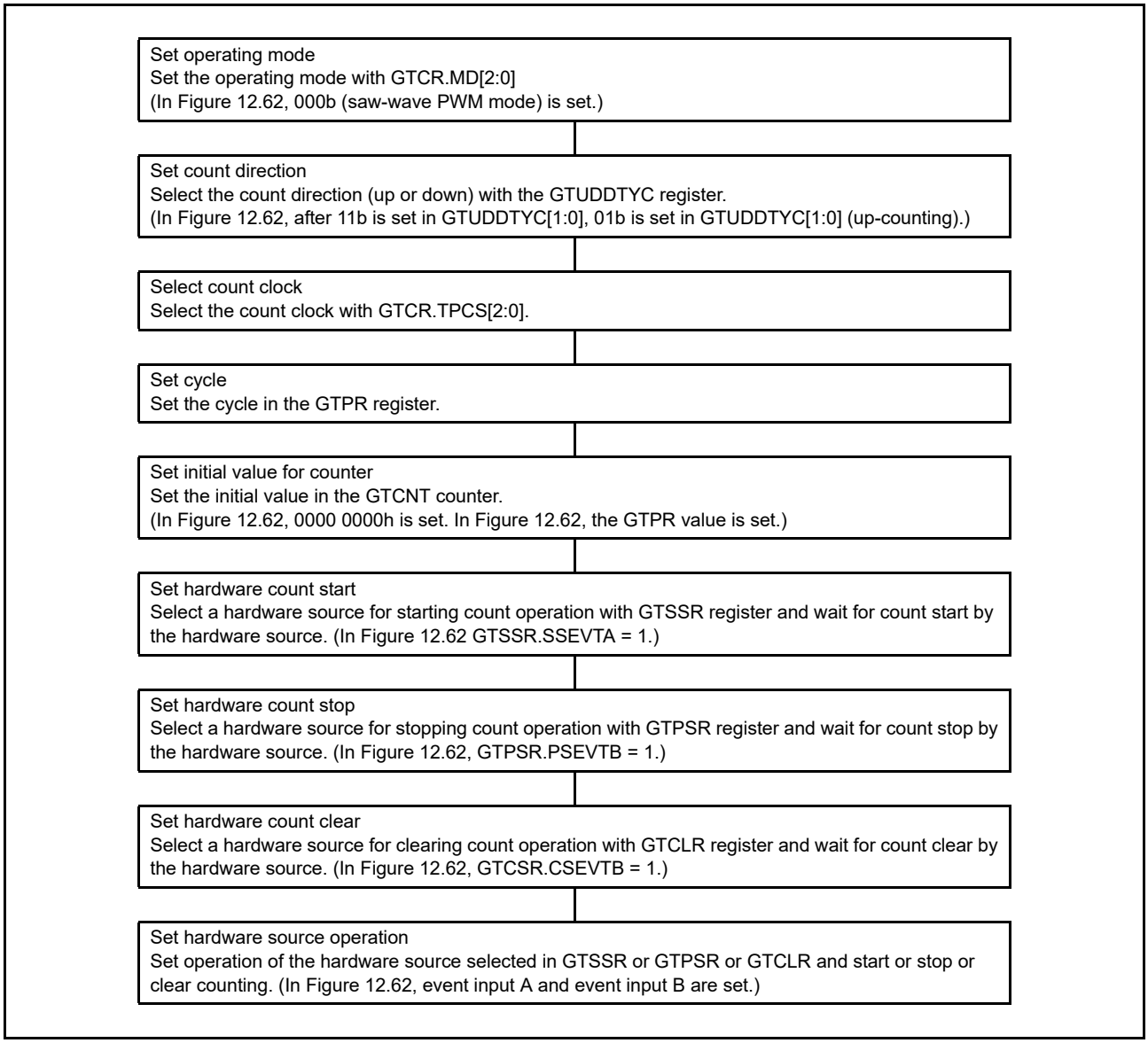


Figure 12.63 Example setting for simultaneous start by hardware source

12.3.9 PWM Output Operation Examples

12.3.9.1 Synchronized PWM Output

The GPT output 16 phases of linked PWM waveforms for a maximum of 8 channels by synchronizing operation on channels.

Figure 12.64 shows an example in which four channels perform synchronized operation in saw-wave PWM mode and eight phases of PWM waveforms are output. The GTIOCA is set so that it outputs low as the initial value, high at a GTCCRA compare match, and low at the cycle end. The GTIOCB is set so that it outputs low as the initial value, high at a GTCCRB compare match, and low at the cycle end.

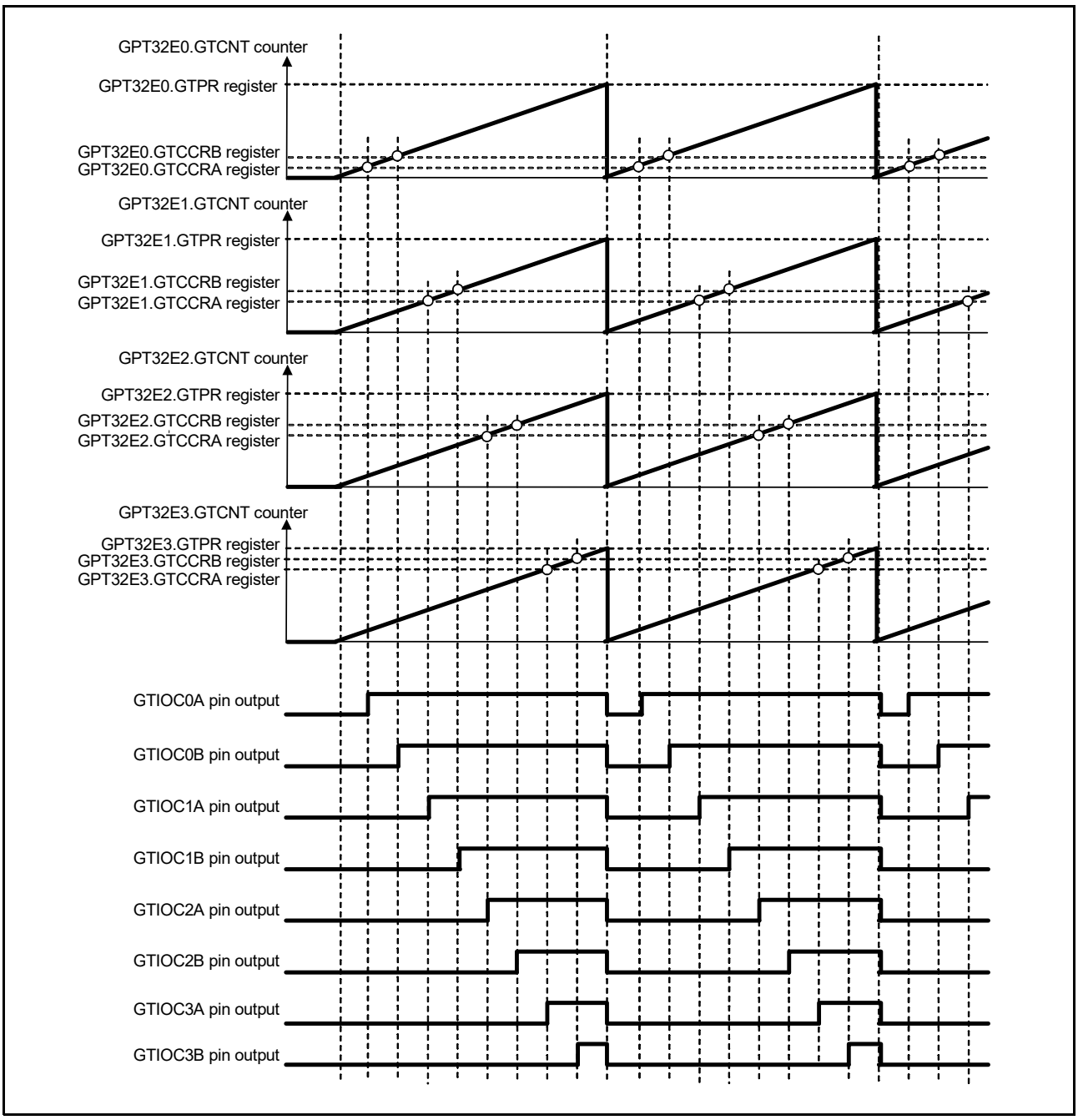


Figure 12.64 Example of synchronized PWM output

12.3.9.2 Three-Phase Saw-Wave Complementary PWM Output

Figure 12.65 shows an example in which three channels perform synchronized operation in saw-wave PWM mode and 3-phase complementary PWM waveforms are output. The GTIOCA pin is set so that it outputs low as the initial value, high at a GTCCRA compare match, and low at the cycle end. The GTIOCB pin is set so that it outputs high as the initial value, low at a GTCCRB compare match, and high at the cycle end.

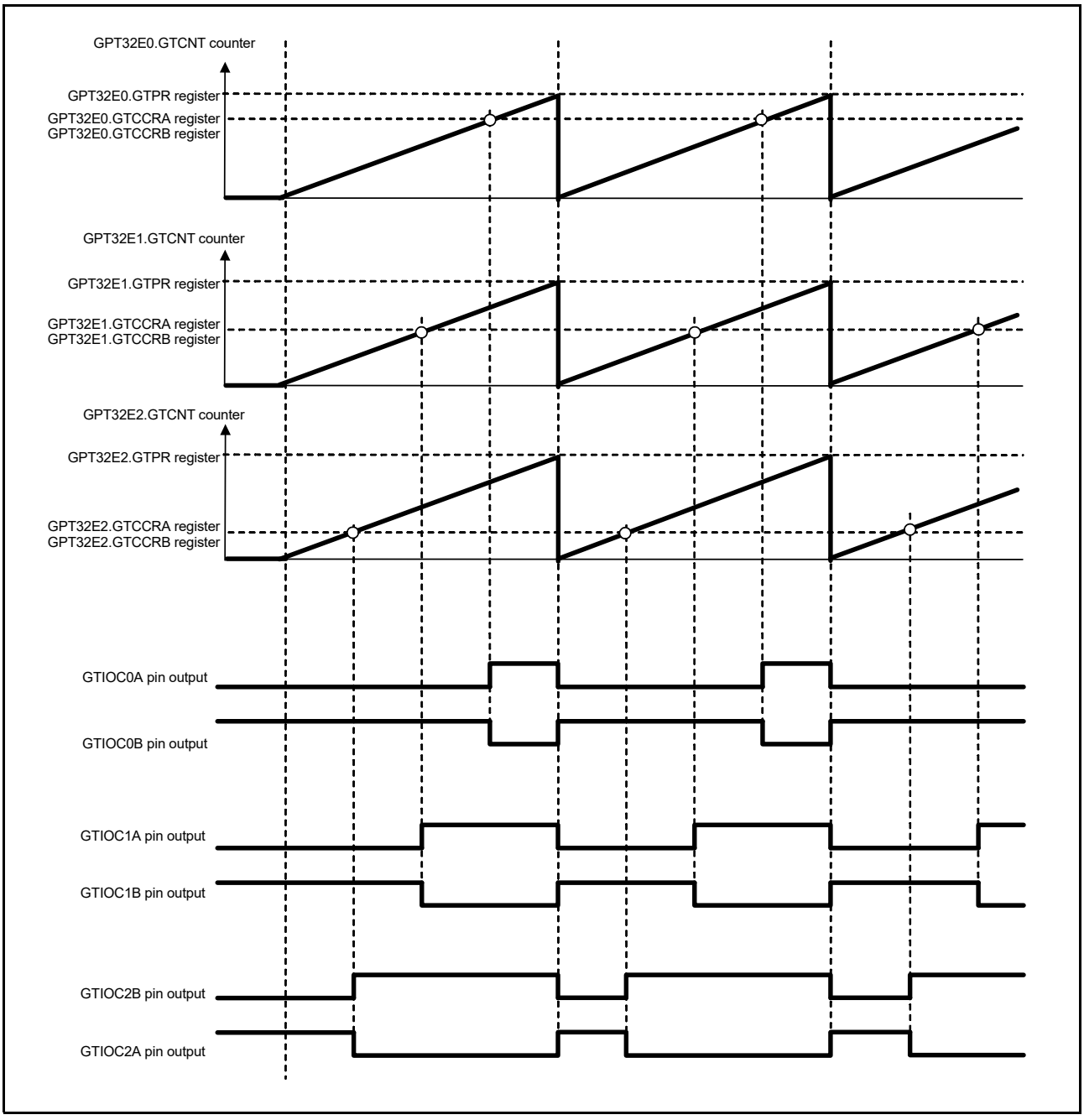


Figure 12.65 Example of 3-phase saw-wave complementary PWM output

12.3.9.3 3-Phase Saw-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 12.66 shows an example in which three channels perform synchronized operation in saw-wave one-shot pulse mode with automatic dead time setting and 3-phase complementary PWM waveforms are output. The GTIOCA pin is set so that it outputs low as the initial value, toggle the output at a GTCCRA compare match, and retain the output at the cycle end. The GTIOCB pin is set so that it outputs high as the initial value, toggle the output at a GTCCRB compare match, and retain the output at the cycle end.

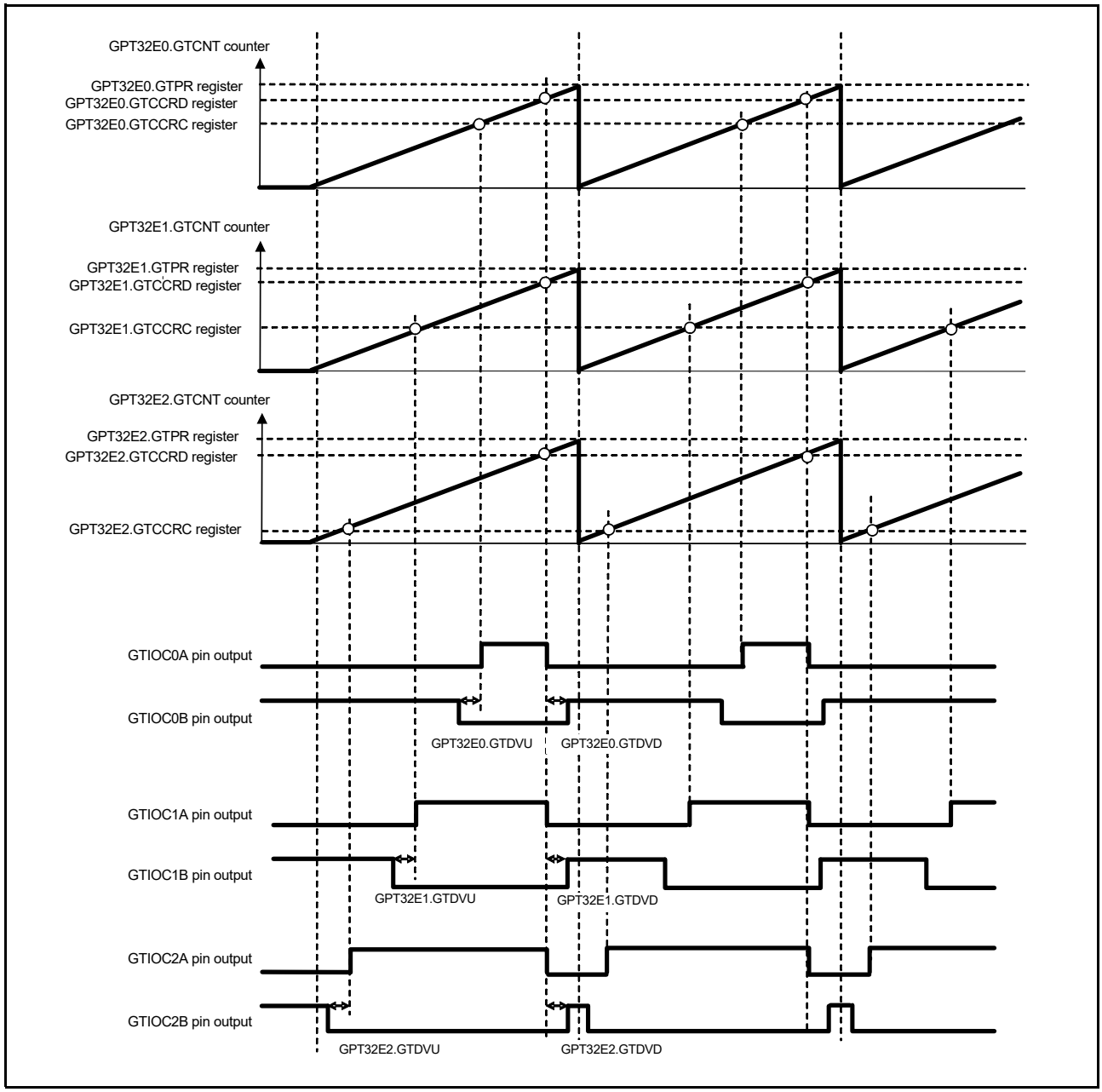


Figure 12.66 Example of 3-phase saw-wave complementary PWM output with automatic dead time setting

12.3.9.4 3-Phase Triangle-Wave Complementary PWM Output

Figure 12.67 shows an example in which three channels perform synchronized operation in triangle-wave PWM mode 1 and 3-phase complementary PWM waveforms are output. The GTIOCA pin is set so that it outputs low as the initial value, toggles the output at a GTCCRA compare match, and retains the output at the cycle end. The GTIOCB pin is set so that it outputs high as the initial value, toggles the output at a GTCCRB compare match, and retains the output at the cycle end.

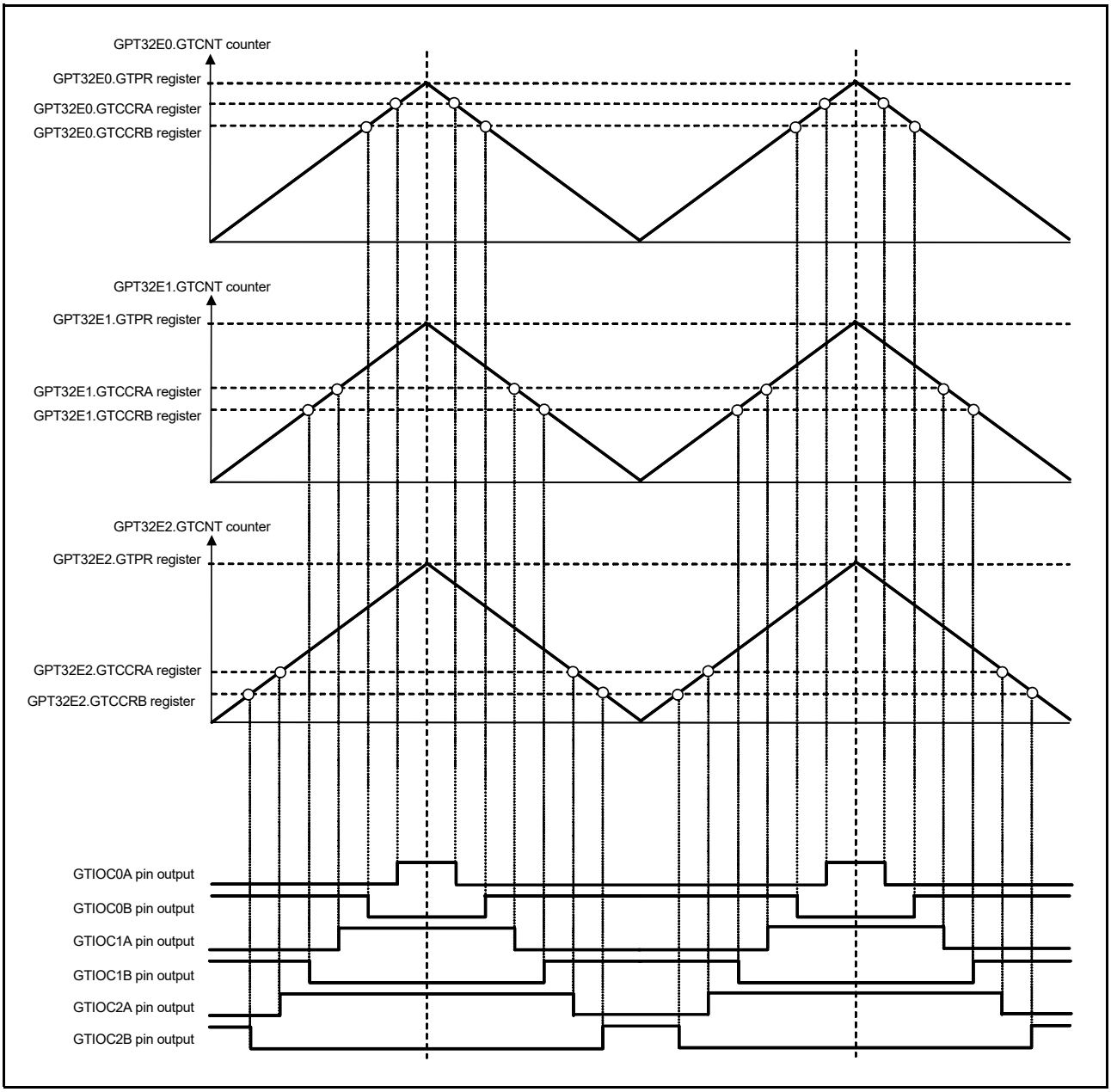


Figure 12.67 Example of 3-phase triangle-wave complementary PWM output

12.3.9.5 3-Phase Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 12.68 shows an example in which three channels perform synchronized operation in triangle-wave PWM mode 1 with automatic dead time setting and 3-phase complementary PWM waveforms are output. The GTIOCA pin is set so that it outputs low as the initial value, toggles the output at a GTCCRA compare match, and retains the output at the cycle end. The GTIOCB pin is set so that it outputs high as the initial value, toggles the output at a GTCCRB compare match, and retains the output at the cycle end.

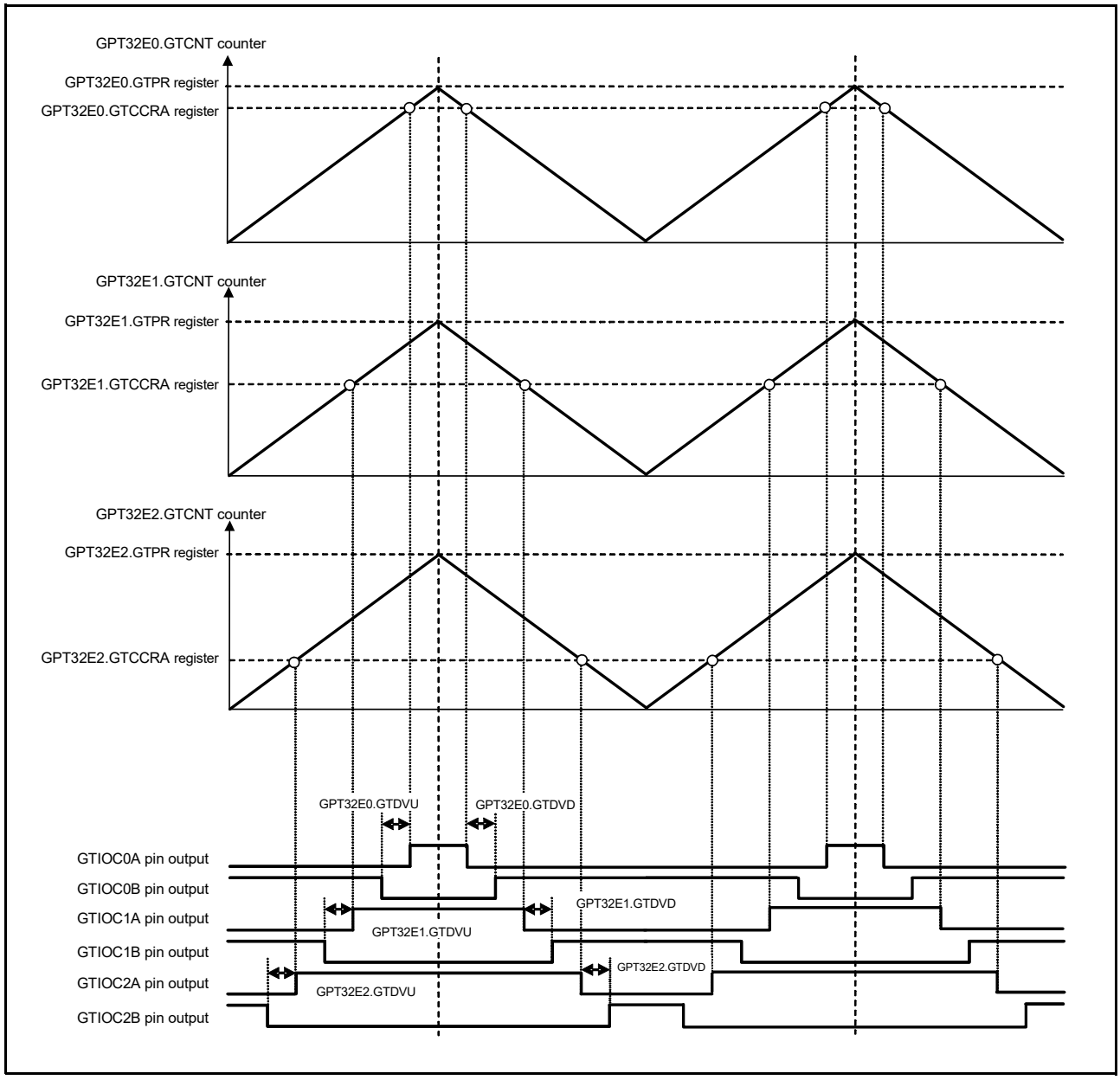


Figure 12.68 Example of 3-phase triangle-wave complementary PWM output with automatic dead time setting

12.3.9.6 3-Phase Asymmetric Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 12.69 shows an example in which three channels perform synchronized operation in triangle-wave PWM mode 3 with automatic dead time setting and 3-phase complementary PWM waveforms are output. The GTIOCA is set so that it outputs low as the initial value, toggles the output at a GTCCRA compare match, and retains the output at the cycle end. The GTIOCB is set so that it outputs high as the initial value, toggles the output at a GTCCRB compare match, and retains the output at the cycle end.

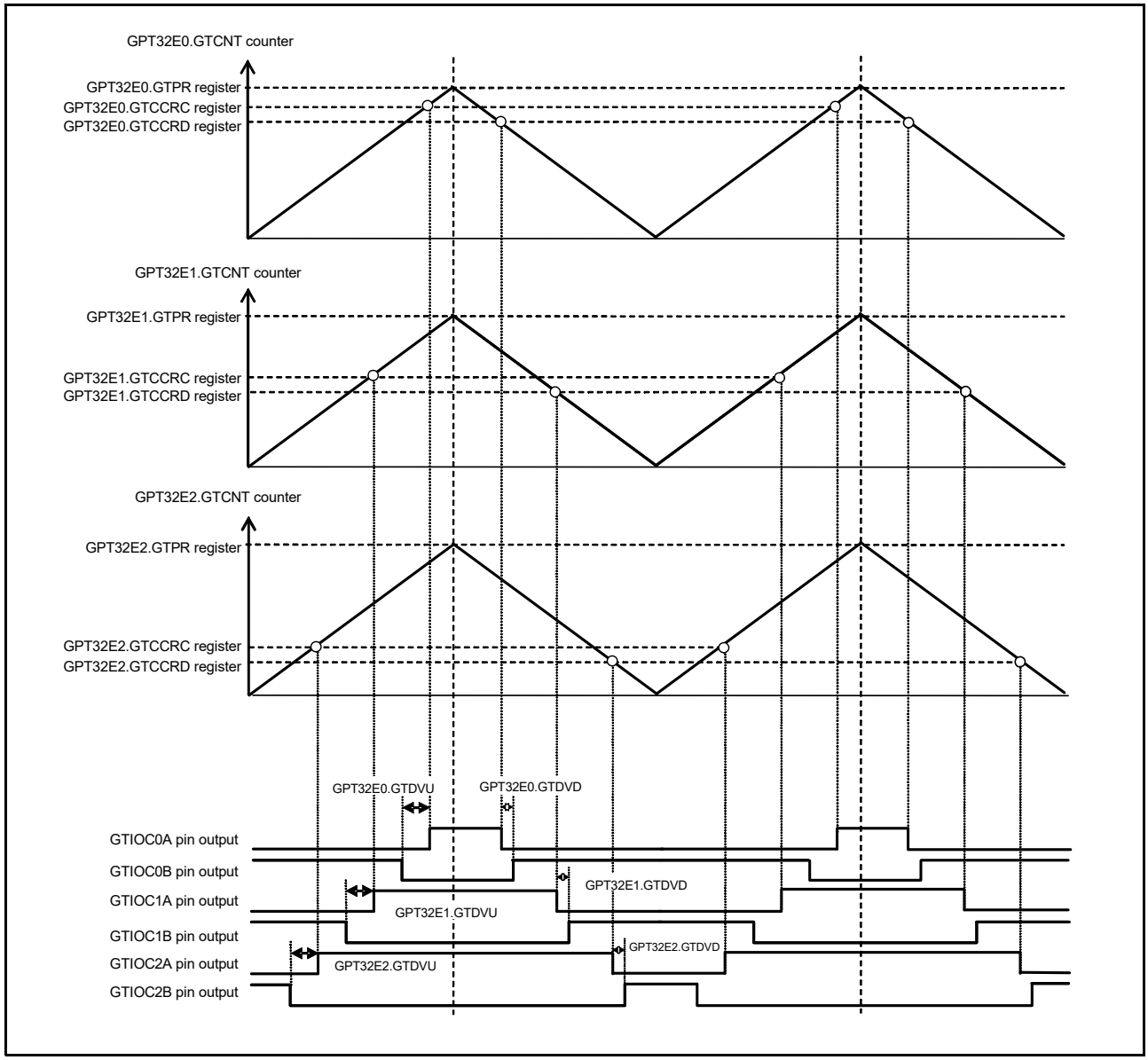


Figure 12.69 Example of 3-phase asymmetric triangle-wave complementary PWM output with automatic dead time setting

12.3.10 Phase Counting Function

The phase difference between the GTIOCA pin input and GTIOCB pin input is detected and the associated GTCNT counts up or counts down. The detectable phase difference is available in any combination with the relationship between the edge and the level of GTIOCA pin and GTIOCB pin input being set in the GTUPSR and GTDNSR registers. For details on count operation, see section 12.3.1.1, Counter Operation.

Figure 12.70 to Figure 12.79 show phase counting modes 1 to 5. Table 12.9 to Table 12.18 show conditions of up-counting or down-counting and lists settings for the GTUPSR and GTDNSR registers.

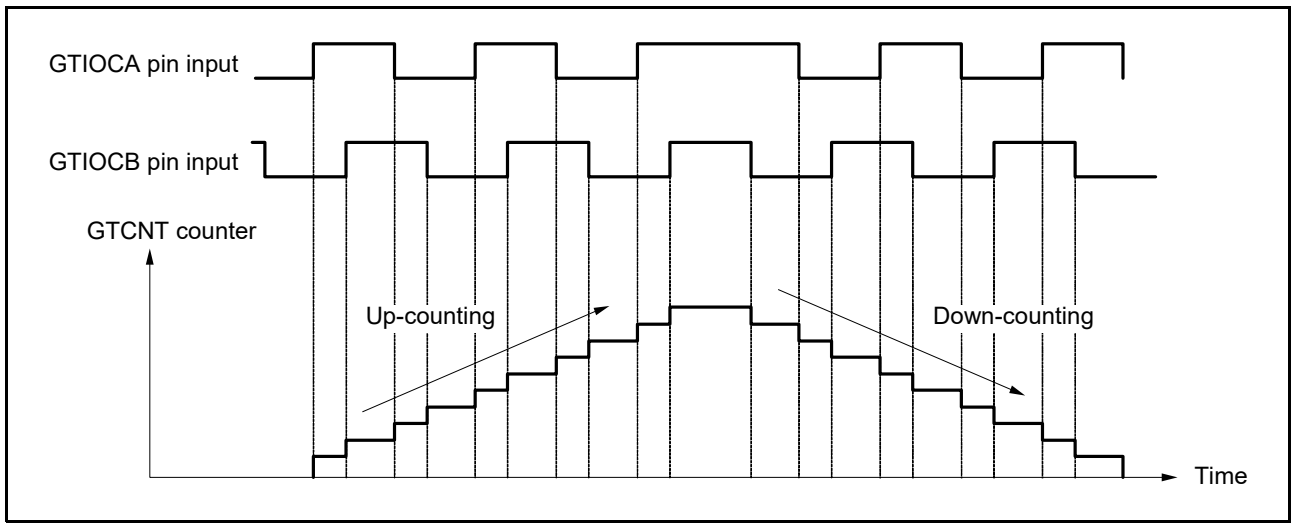


Figure 12.70 Example of phase counting mode 1

Table 12.9 Conditions of up-counting and down-counting in phase counting mode 1

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
high	↑	Up-counting	GTUPSR = 00006900h GTDNSR = 00009600h
low	↓		
↑	low		
↓	high		
high	↓	Down-counting	
low	↑		
↑	high		
↓	low		

Note: ↑ : Rising edge

↓ : Falling edge

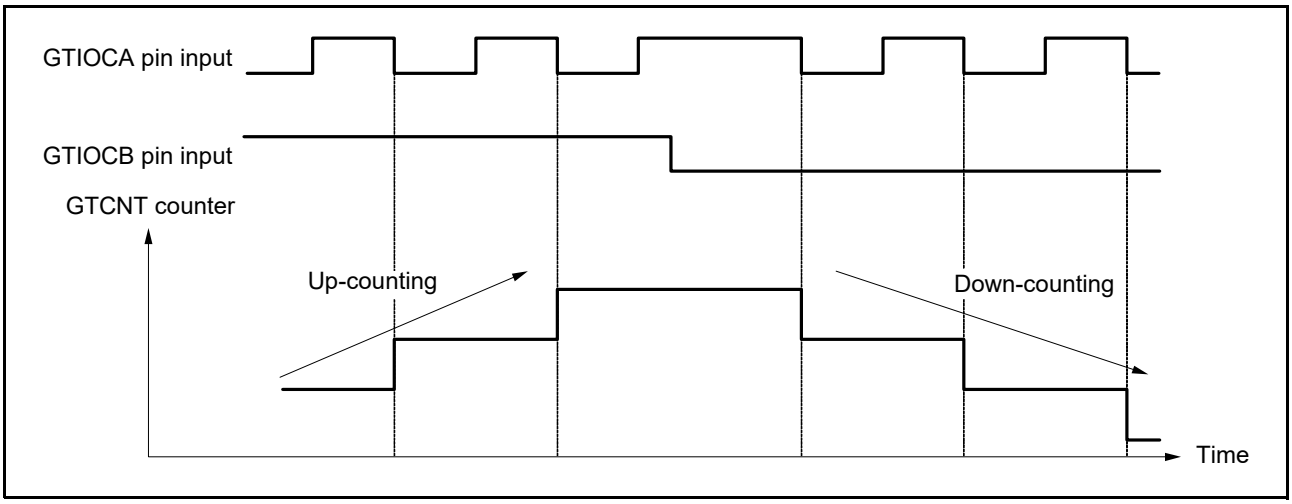


Figure 12.71 Example of phase counting mode 2 (A)

Table 12.10 Conditions of up-counting and down-counting in phase counting mode 2 (A)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
high	↑	Don't care	GTUPSR = 00000800h GTDNSR = 00000400h
low	↓		
↑	low		
↓	high	Up-counting	
high	↓	Don't care	
low	↑		
↑	high		
↓	low	Down-counting	

Note: ↑ : Rising edge
Note: ↓ : Falling edge

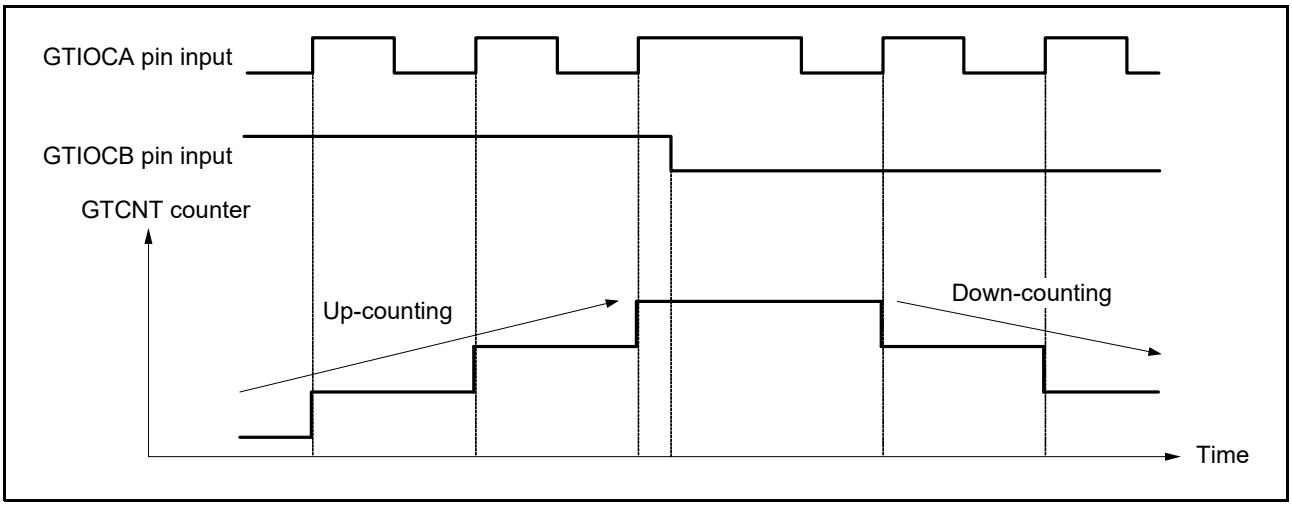


Figure 12.72 Example of phase counting mode 2 (B)

Table 12.11 Conditions of up-counting and down-counting in phase counting mode 2 (B)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
high	↑	Don't care	GTUPSR = 00000200h GTDNSR = 00000100h
low	↓		
↑	low	Down-counting	
↓	high	Don't care	
high	↓		
low	↑		
↑	high	Up-counting	
↓	low	Don't care	

Note: ↑ : Rising edge
Note: ↓ : Falling edge

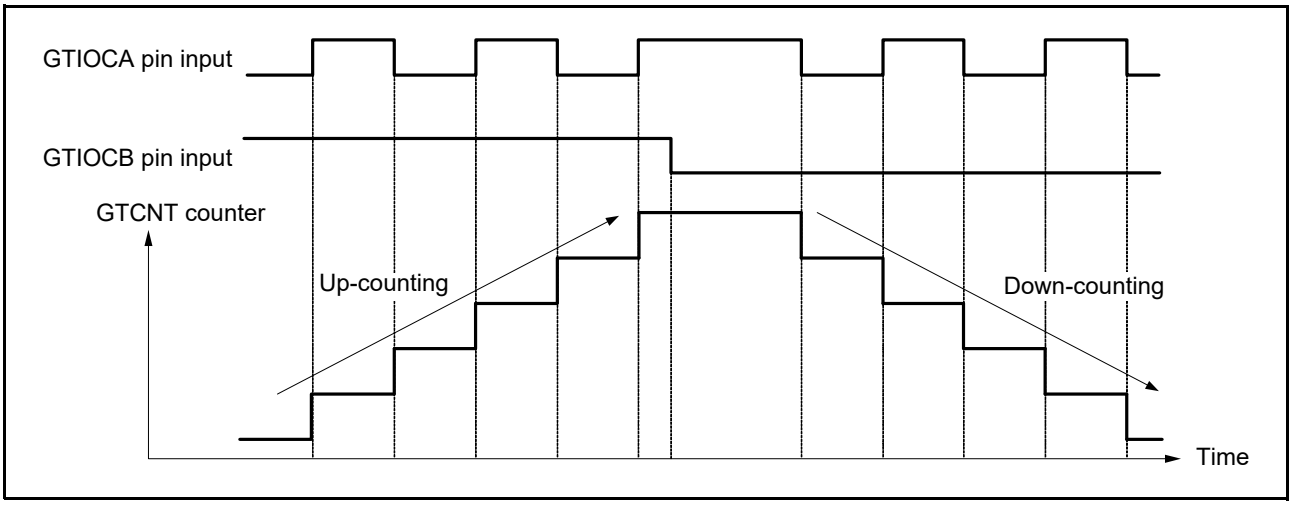


Figure 12.73 Example of phase counting mode 2 (C)

Table 12.12 Conditions of up-counting and down-counting in phase counting mode 2 (C)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
high	↑	Don't care	GTUPSR = 00000A00h GTDNSR = 00000500h
low	↓		
↑	low	Down-counting	
↓	high	Up-counting	
high	↓	Don't care	
low	↑		
↑	high	Up-counting	
↓	low	Down-counting	

Note: ↑ : Rising edge
Note: ↓ : Falling edge

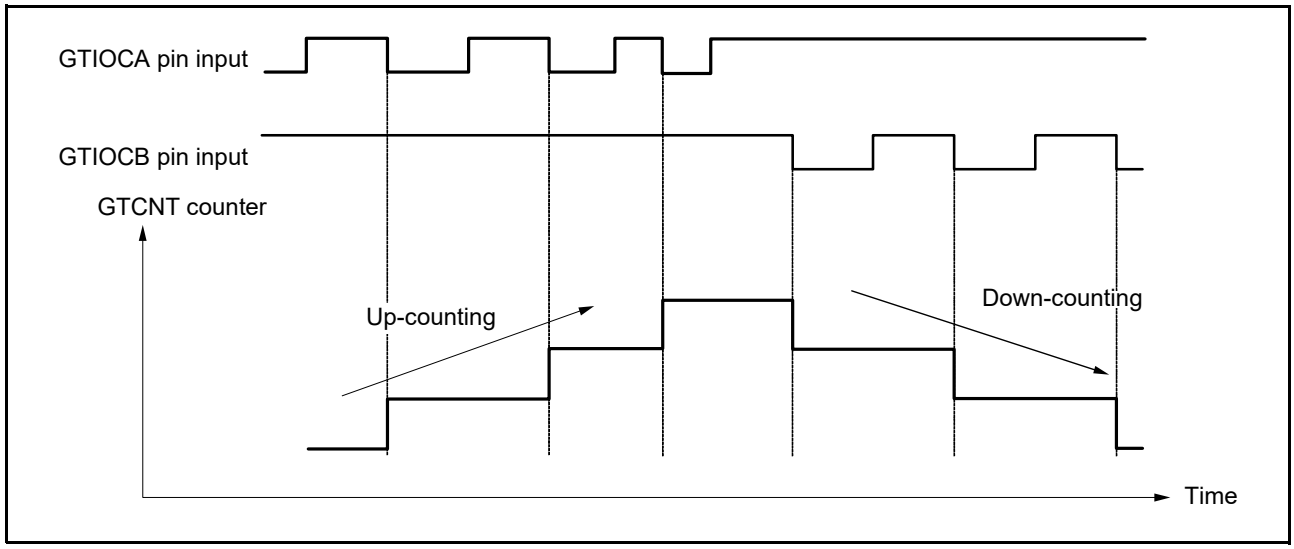


Figure 12.74 Example of phase counting mode 3 (A)

Table 12.13 Conditions of up-counting and down-counting in phase counting mode 3 (A)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
high	↑	Don't care	GTUPSR = 00000800h GTDNSR = 00008000h
low	↓		
↑	low		
↓	high	Up-counting	
high	↓	Down-counting	
low	↑	Don't care	
↑	high		
↓	low		

Note: ↑ : Rising edge
Note: ↓ : Falling edge

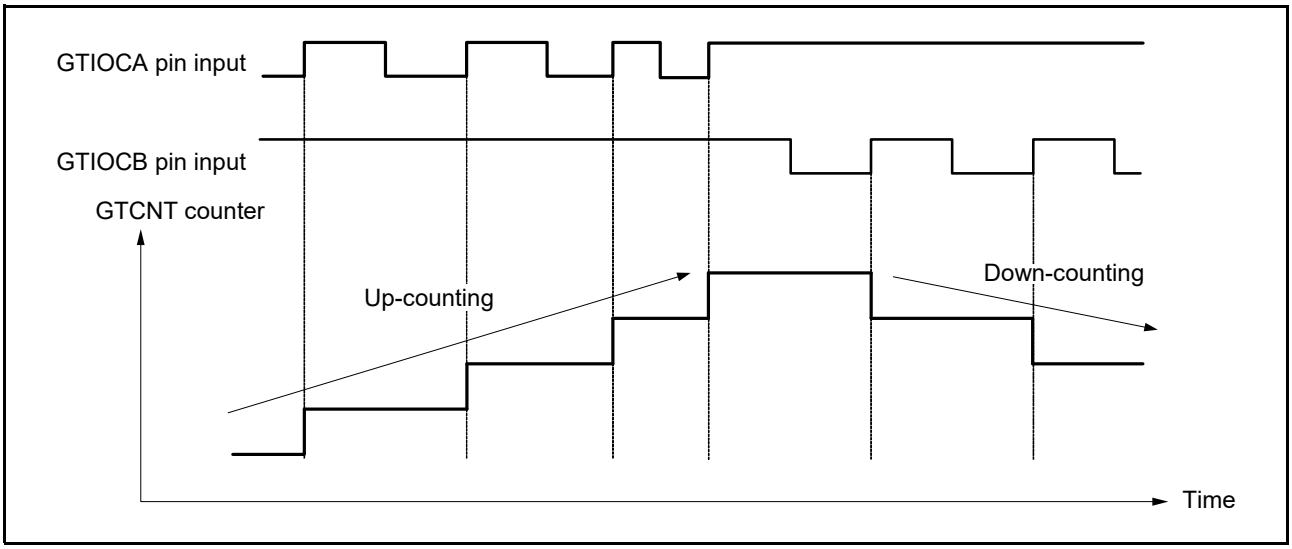


Figure 12.75 Example of phase counting mode 3 (B)

Table 12.14 Conditions of up-counting and down-counting in phase counting mode 3 (B)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
high	↑	Down-counting	GTUPSR = 00000200h GTDNSR = 00002000h
low	↓	Don't care	
↑	low		
↓	high		
high	↓		
low	↑		
↑	high	Up-counting	
↓	low	Don't care	

Note: ↕ : Rising edge
Note: ↕ : Falling edge

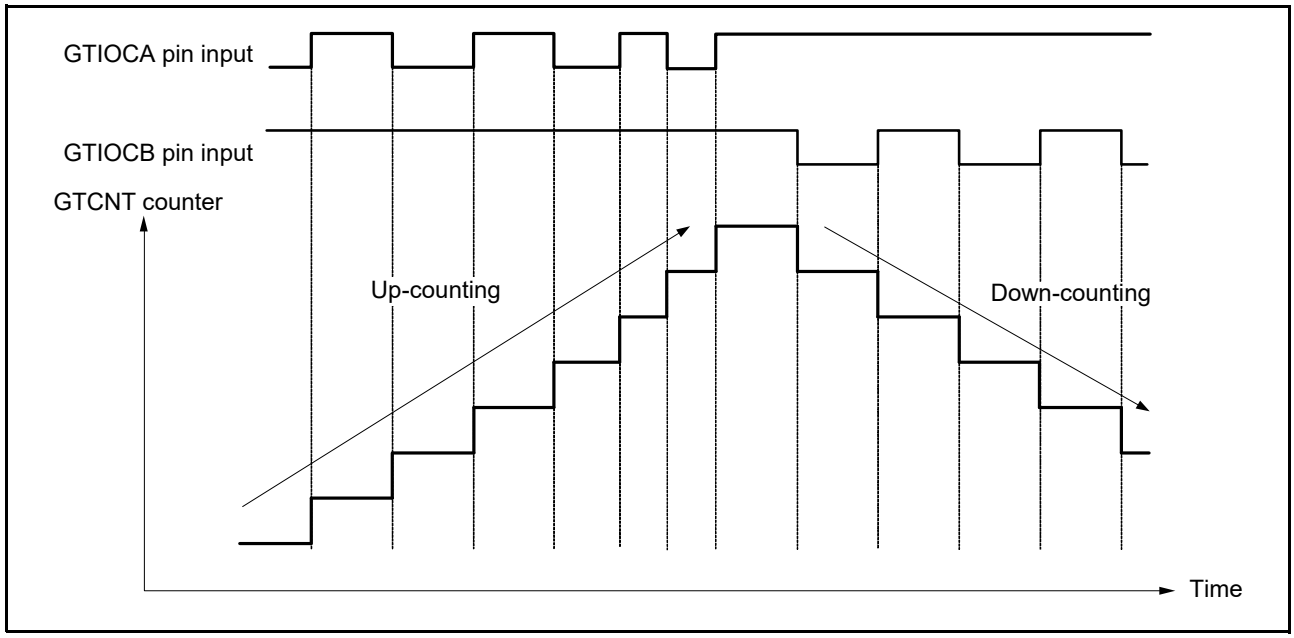


Figure 12.76 Example of phase counting mode 3 (C)

Table 12.15 Conditions of up-counting and down-counting in phase counting mode 3 (C)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
high	↑	Down-counting	GTUPSR = 00000A00h GTDNSR = 0000A000h
low	↓	Don't care	
↑	low		
↓	high	Up-counting	
high	↓	Down-counting	
low	↑	Don't care	
↑	high	Up-counting	
↓	low	Don't care	

Note: ↑ : Rising edge
Note: ↓ : Falling edge

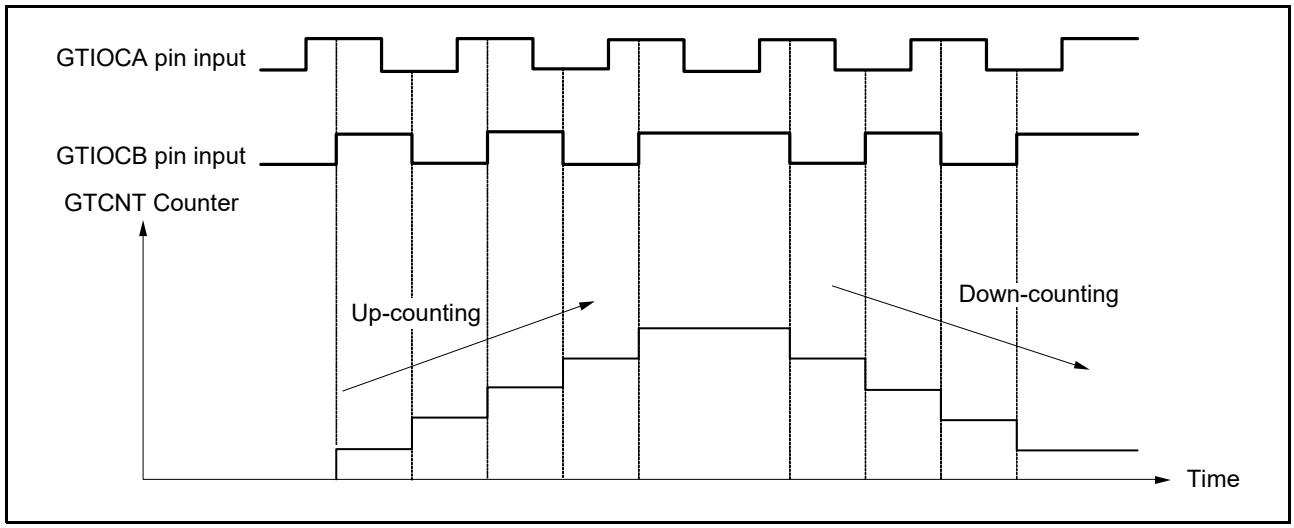


Figure 12.77 Example of phase counting mode 4

Table 12.16 Conditions of up-counting and down-counting in phase counting mode 4

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
high	↑	Up-counting	GTUPSR = 00006000h GTDNSR = 00009000h
low	↓		
↑	low	Don't care	
↓	high		
high	↓	Down-counting	
low	↑		
↑	high	Don't care	
↓	low		

Note: ↑ : Rising edge
Note: ↓ : Falling edge

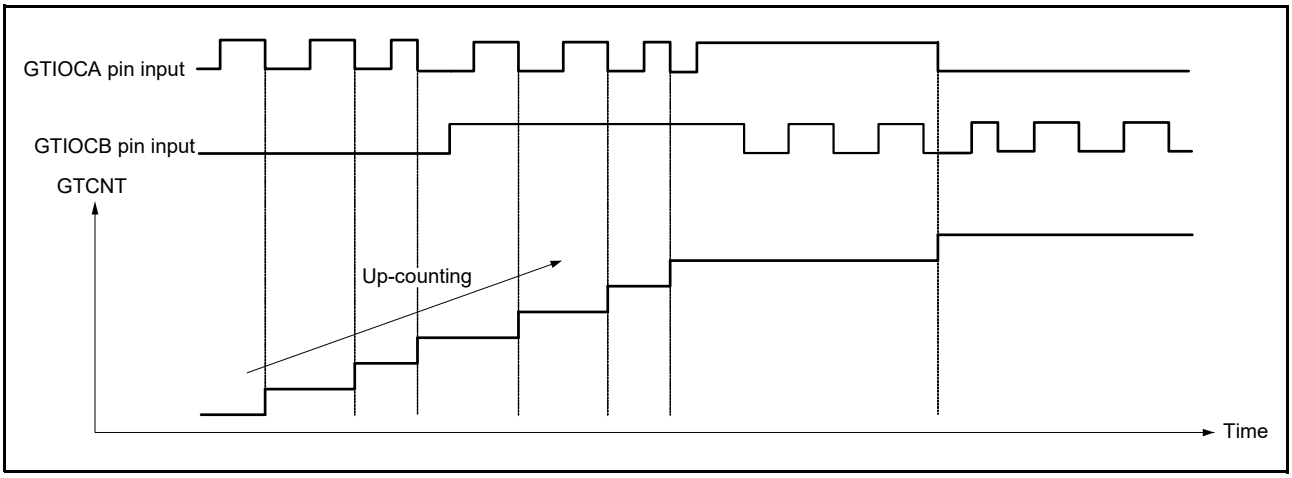


Figure 12.78 Example of phase counting mode 5 (A)

Table 12.17 Conditions of up-counting and down-counting in phase counting mode 5 (A)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
high	↑	Don't care	GTUPSR = 00000C00h GTDNSR = 00000000h
low	↓		
↑	low		
↓	high	Up-counting	
high	↓	Don't care	
low	↑		
↑	high		
↓	low	Up-counting	

Note: ↑ : Rising edge
Note: ↕ : Falling edge

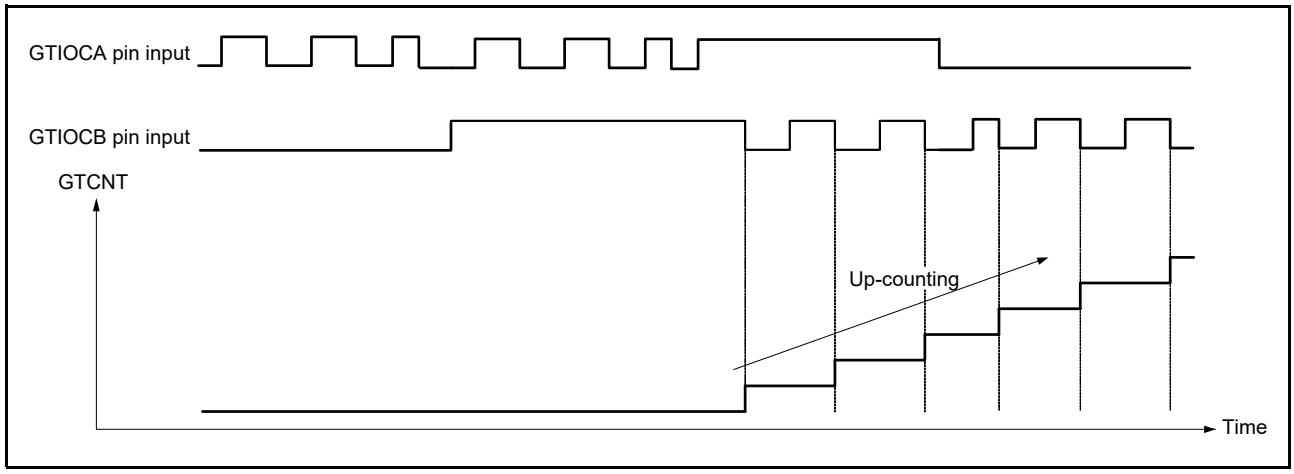


Figure 12.79 Example of phase counting mode 5 (B)

Table 12.18 Conditions of up-counting and down-counting in phase counting mode 5 (B)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
high	↑	Don't care	GTUPSR = 0000C000h GTDNSR = 00000000h
low	↓	Up-counting	
↑	low	Don't care	
↓	high		
high	↓	Up-counting	
low	↑	Don't care	
↑	high		
↓	low		

Note: ↑ : Rising edge
Note: ↓ : Falling edge

12.4 Interrupt Sources

12.4.1 Interrupt Sources and Priorities

The GPT provides the following interrupt sources:

- GTCCR input capture/compare match
- GTADTR compare match
- GTCNT counter overflow (GTPR compare match)/underflow.

Each interrupt source has its own status flag. When an interrupt source signal is generated, the associated status flag in GTST is set to 1. The associated status flag in GTST can be cleared by writing 0. If flag set and flag clear happen at the same time, flag clear takes priority over flag set. These flags are automatically updated by internal state. Relative channel priorities can be changed by the Interrupt Controller. However the priority within a channel is fixed. For details, see section 7, Interrupt Controller. Table 12.19 lists the GPT interrupt sources.

Table 12.19 Interrupt sources

Channel	Name	Interrupt source	Interrupt flag	DMAC activation
0	CCMPA0	GPT32E0.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB0	GPT32E0.GTCCRB input capture/compare match	TCFB	Possible
	CMPC0	GPT32E0.GTCCRC compare match	TCFC	Possible
	CMPD0	GPT32E0.GTCCRD compare match	TCFD	Possible
	CMPE0	GPT32E0.GTCCRE compare match	TCFE	Possible
	CMPF0	GPT32E0.GTCCRF compare match	TCFF	Possible
	ADTRGA0	GPT32E0.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB0	GPT32E0.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF0	GPT32E0.GTCNT overflow (GPT32E0.GTPR compare match)	TCFPO	Possible
	UNF0	GPT32E0.GTCNT underflow	TCFPU	Possible
1	CCMPA1	GPT32E1.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB1	GPT32E1.GTCCRB input capture/compare match	TCFB	Possible
	CMPC1	GPT32E1.GTCCRC compare match	TCFC	Possible
	CMPD1	GPT32E1.GTCCRD compare match	TCFD	Possible
	CMPE1	GPT32E1.GTCCRE compare match	TCFE	Possible
	CMPF1	GPT32E1.GTCCRF compare match	TCFF	Possible
	ADTRGA1	GPT32E1.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB1	GPT32E1.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF1	GPT32E1.GTCNT overflow (GPT32E1.GTPR compare match)	TCFPO	Possible
	UNF1	GPT32E1.GTCNT underflow	TCFPU	Possible

Table 12.19 Interrupt sources

Channel	Name	Interrupt source	Interrupt flag	DMAC activation
2	CCMPA2	GPT32E2.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB2	GPT32E2.GTCCRB input capture/compare match	TCFB	Possible
	CMPC2	GPT32E2.GTCCRC compare match	TCFC	Possible
	CMPD2	GPT32E2.GTCCRD compare match	TCFD	Possible
	CMPE2	GPT32E2.GTCCRE compare match	TCFE	Possible
	CMPF2	GPT32E2.GTCCRF compare match	TCFF	Possible
	ADTRGA2	GPT32E2.GTCCRE compare match	ADTRAUF ADTRADF	Possible
	ADTRGB2	GPT32E2.GTCCRF compare match	ADTRBUF ADTRBDF	Possible
	OVF2	GPT32E2.GTCNT overflow (GPT32E2.GTPR compare match)	TCFPO	Possible
	UNF2	GPT32E2.GTCNT underflow	TCFPU	Possible
3	CCMPA3	GPT32E3.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB3	GPT32E3.GTCCRB input capture/compare match	TCFB	Possible
	CMPC3	GPT32E3.GTCCRC compare match	TCFC	Possible
	CMPD3	GPT32E3.GTCCRD compare match	TCFD	Possible
	CMPE3	GPT32E3.GTCCRE compare match	TCFE	Possible
	CMPF3	GPT32E3.GTCCRF compare match	TCFF	Possible
	ADTRGA3	GPT32E3.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB3	GPT32E3.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF3	GPT32E3.GTCNT overflow (GPT32E3.GTPR compare match)	TCFPO	Possible
	UNF3	GPT32E3.GTCNT underflow	TCFPU	Possible
4	CCMPA4	GPT32E4.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB4	GPT32E4.GTCCRB input capture/compare match	TCFB	Possible
	CMPC4	GPT32E4.GTCCRC compare match	TCFC	Possible
	CMPD4	GPT32E4.GTCCRD compare match	TCFD	Possible
	CMPE4	GPT32E4.GTCCRE compare match	TCFE	Possible
	CMPF4	GPT32E4.GTCCRF compare match	TCFF	Possible
	ADTRGA4	GPT32E4.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB4	GPT32E4.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF4	GPT32E4.GTCNT overflow (GPT32E4.GTPR compare match)	TCFPO	Possible
	UNF4	GPT32E4.GTCNT underflow	TCFPU	Possible

Table 12.19 Interrupt sources

Channel	Name	Interrupt source	Interrupt flag	DMAC activation
5	CCMPA5	GPT32E5.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB5	GPT32E5.GTCCRB input capture/compare match	TCFB	Possible
	CMPC5	GPT32E5.GTCCRC compare match	TCFC	Possible
	CMPD5	GPT32E5.GTCCRD compare match	TCFD	Possible
	CMPE5	GPT32E5.GTCCRE compare match	TCFE	Possible
	CMPF5	GPT32E5.GTCCRF compare match	TCFF	Possible
	ADTRGA5	GPT32E5.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB5	GPT32E5.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF5	GPT32E5.GTCNT overflow (GPT32E5.GTPR compare match)	TCFPO	Possible
	UNF5	GPT32E5.GTCNT underflow	TCFPU	Possible
6	CCMPA6	GPT32E6.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB6	GPT32E6.GTCCRB input capture/compare match	TCFB	Possible
	CMPC6	GPT32E6.GTCCRC compare match	TCFC	Possible
	CMPD6	GPT32E6.GTCCRD compare match	TCFD	Possible
	CMPE6	GPT32E6.GTCCRE compare match	TCFE	Possible
	CMPF6	GPT32E6.GTCCRF compare match	TCFF	Possible
	ADTRGA6	GPT32E6.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB6	GPT32E6.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF6	GPT32E6.GTCNT overflow (GPT32E6.GTPR compare match)	TCFPO	Possible
	UNF6	GPT32E6.GTCNT underflow	TCFPU	Possible
7	CCMPA7	GPT32E7.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB7	GPT32E7.GTCCRB input capture/compare match	TCFB	Possible
	CMPC7	GPT32E7.GTCCRC compare match	TCFC	Possible
	CMPD7	GPT32E7.GTCCRD compare match	TCFD	Possible
	CMPE7	GPT32E7.GTCCRE compare match	TCFE	Possible
	CMPF7	GPT32E7.GTCCRF compare match	TCFF	Possible
	ADTRGA7	GPT32E7.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB7	GPT32E7.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF7	GPT32E7.GTCNT overflow (GPT32E7.GTPR compare match)	TCFPO	Possible
	UNF7	GPT32E7.GTCNT underflow	TCFPU	Possible

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

(1) ADTRGAn interrupt (n = 0 to 7)

When the GTCNT counter value matches with the GTADTRA register, an interrupt request is generated under the following condition:

- In up-counting, the interrupt enable bit (ADTRAUEN) in the GTINTAD register is 1
- In down-counting, the interrupt enable bit (ADTRADEN) in the GTINTAD register is 1.

In event count operation performing, this interrupt request is not generated.

(2) ADTRGBn interrupt (n = 0 to 7)

When the GTCNT counter value matches with the GTADTRB register, an interrupt request is generated under the following condition:

- In up-counting, the interrupt enable bit (ADTRBUEN) in the GTINTAD register is 1
- In down-counting, the interrupt enable bit (ADTRBDEN) in the GTINTAD register is 1.

In event count operation performing, this interrupt request is not generated.

(3) CCMPAn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRA register functions as a compare match register, the GTCNT counter value matches with the GTCCRA register.
- When the GTCCRA register functions as an input capture register, the GTCNT counter value is transferred to the GTCCRA register by an input capture signal.

(4) CCMPBn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRB register functions as a compare match register, the GTCNT counter value matches with the GTCCRB register.
- When the GTCCRB register functions as an input capture register, the GTCNT counter value is transferred to the GTCCRB register by an input capture signal.

(5) CMPCn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRC register functions as a compare match register, the GTCNT counter value matches with the GTCCRC register.

A compare match is not performed and an interrupt is not requested under the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 01b, 10b, 11b (buffer operation with the GTCCRC register).

(6) CMPDn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRD register functions as a compare match register, the GTCNT counter value matches with the GTCCRD register.

A compare match is not performed and therefore interrupt is not requested under the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 10b, 11b (buffer operation with the GTCCRD register).

(7) CMPEn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition.

- When the GTCCRE register functions as a compare match register, the GTCNT counter value matches with the GTCCRE register.

A compare match is not performed and an interrupt is not requested under the following conditions.

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 01b, 10b, 11b (buffer operation with the GTCCRE register)

(8) CMPFn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRF register functions as a compare match register, the GTCNT counter value matches with the GTCCRF register.

A compare match is not performed and therefore interrupt is not requested under the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 10b, 11b (buffer operation with the GTCCRF register).

(9) OVFn interrupt (n = 0 to 7)

An interrupt request is generated under the following conditions:

- In saw-wave mode, interrupt requests are enabled at overflows (when the GTCNT counter value changes from GTPR to 0 during up-counting)
- In triangle-wave mode, interrupt requests are enabled at crests (GTCNT changes from GTPR to GTPR-1)
- In counting by hardware sources, overflow (GTCNT changes from GTPR to 0 in up count) has occurred.

(10) UNFn interrupt (n = 0 to 7)

An interrupt request is generated under the following conditions:

- In saw-wave mode, interrupt requests are enabled at underflows (when the GTCNT counter value changes from 0 to GTPR during down-counting).
- In triangle-wave mode, interrupt requests are enabled at troughs (GTCNT changes from 0 to 1).
- In counting by hardware sources, underflow (GTCNT changes from 0 to GTPR in down count) has occurred.

Table 12.20 Interrupt signals, interrupt permission bits, and interrupt status flags

Interrupt signal	Interrupt permission bit	Interrupt status flag
UNFn	GTINTAD[7:6] (GTINTPR[1:0])	GTST[7] (TCFPU)
OVFn		GTST[6] (TCFPO)
ADTRGBn	GTINTAD[19] (ADTRBDEN) GTINTAD[18] (ADTRBUEN)	GTST[19] (ADTRBDF) GTST[18] (ADTRBUF)
ADTRGAn	GTINTAD[17] (ADTRADEN) GTINTAD[16] (ADTRAUEN)	GTST[17] (ADTRADF) GTST[16] (ADTRAUF)
CMPFn	GTINTAD[5] (GTINTF)	GTST[5] (TCFF)
CMPEn	GTINTAD[4] (GTINTE)	GTST[4] (TCFE)
CMPDn	GTINTAD[3] (GTINTD)	GTST[3] (TCFD)
CMPCn	GTINTAD[2] (GTINTC)	GTST[2] (TCFC)
CCMPBn	GTINTAD[1] (GTINTB)	GTST[1] (TCFB)
CCMPAn	GTINTAD[0] (GTINTA)	GTST[0] (TCFA)

12.4.2 DMAC Activation

The DMAC can be activated by the interrupt in each channel. For details, see section 9, Direct Memory Access Controller.

12.4.3 Interrupt and A/D Conversion Request Skipping Function

By setting the GTITC register, the GTCNT counter overflow (GTPR compare match) interrupt (OVFn) and underflow interrupt (UNFn) can be skipped. Other interrupts and A/D converter start request signals can be skipped in coordination with the OVFn/UNFn skipping function.

The interrupt request skipping function only depends on the setting of GTITC register and is independent of the setting of interrupt permission bits in the GTINTAD register.

When both troughs and crests are counted and skipped in triangle-wave mode, if the number of times of skipping is odd, OVFn/UNFn interrupt requests cannot be generated at troughs only or at crests only depending on the skipping counter start timing. To count both troughs and crests and generate the OVFn/UNFn interrupts at troughs only or crests only in triangle-wave mode, you must set an even number of skips.

Similarly, in saw-wave mode, when both overflows and underflows are counted and skipped with the count direction changed, OVFn interrupt requests cannot be generated on either overflows or underflows only. To count both overflows and underflows with the count direction changed and generate the OVFn/UNFn interrupts on either overflows or underflows only in saw wave mode, first check the skipping state carefully.

Before changing the skipping count, you must release the skipping count setting (GTITC.IVTC[1:0] bits = 00b).

Figure 12.80 to Figure 12.85 show examples of skipping function operation.

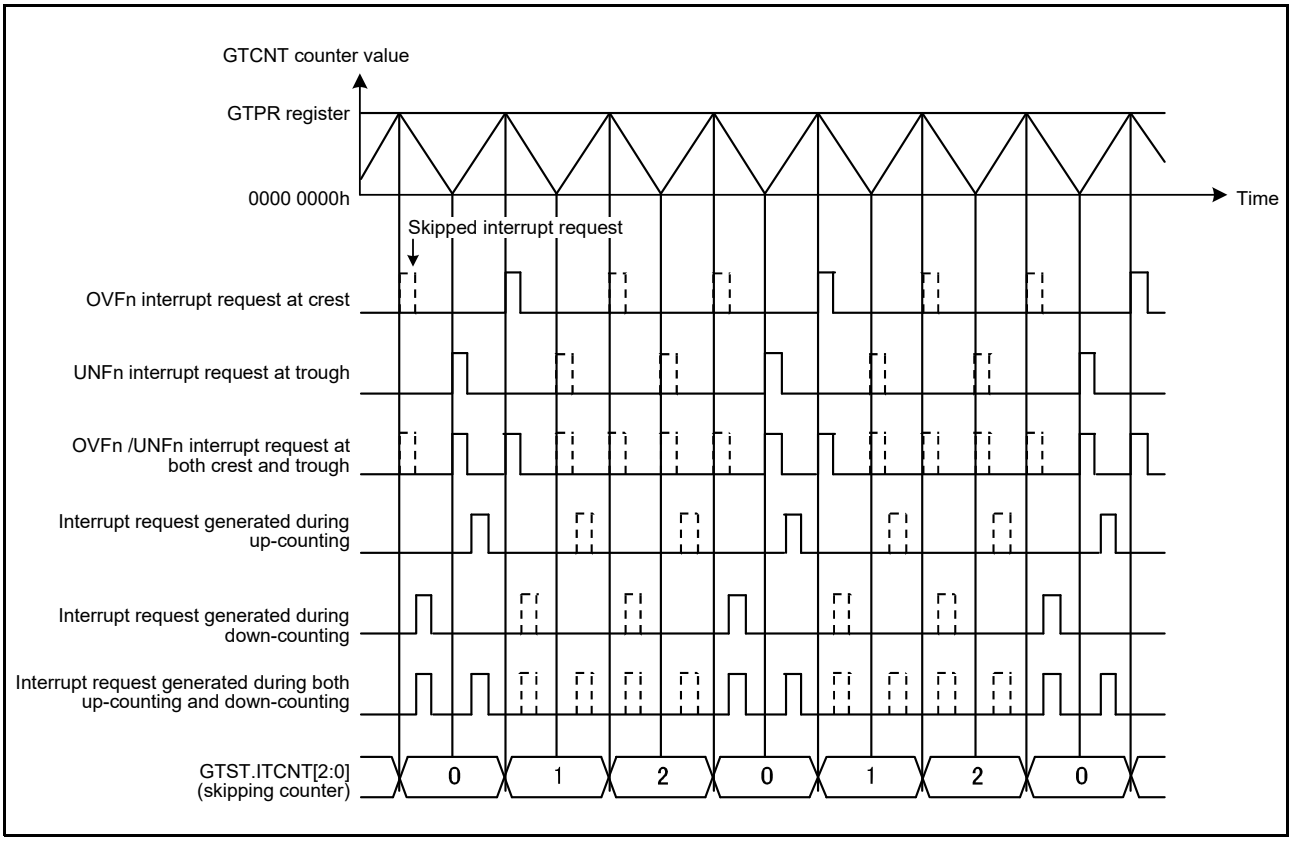


Figure 12.80 Example of interrupt skipping function operation with triangle waves, counting and skipping crests, and skipping count = 2

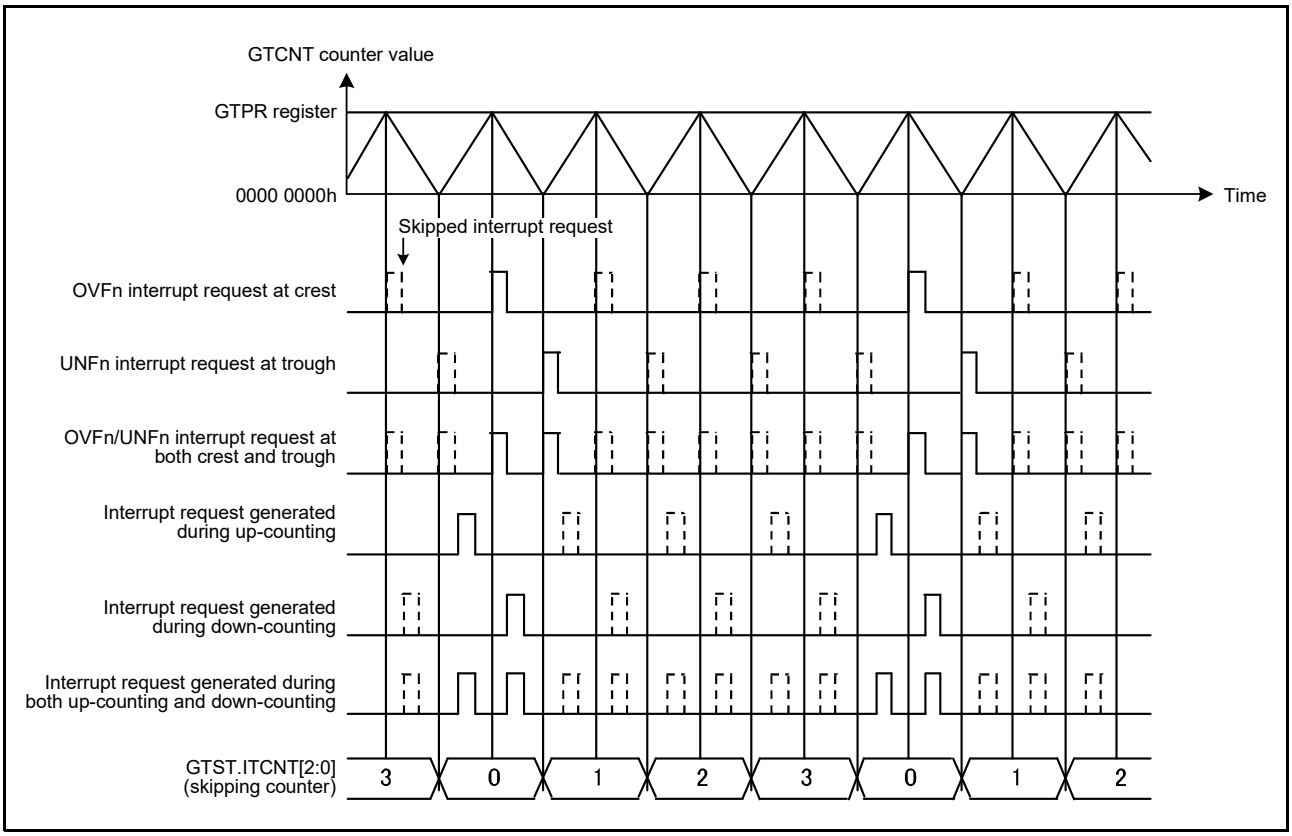


Figure 12.81 Example of interrupt skipping function operation with triangle waves, counting and skipping troughs, and skipping count = 3

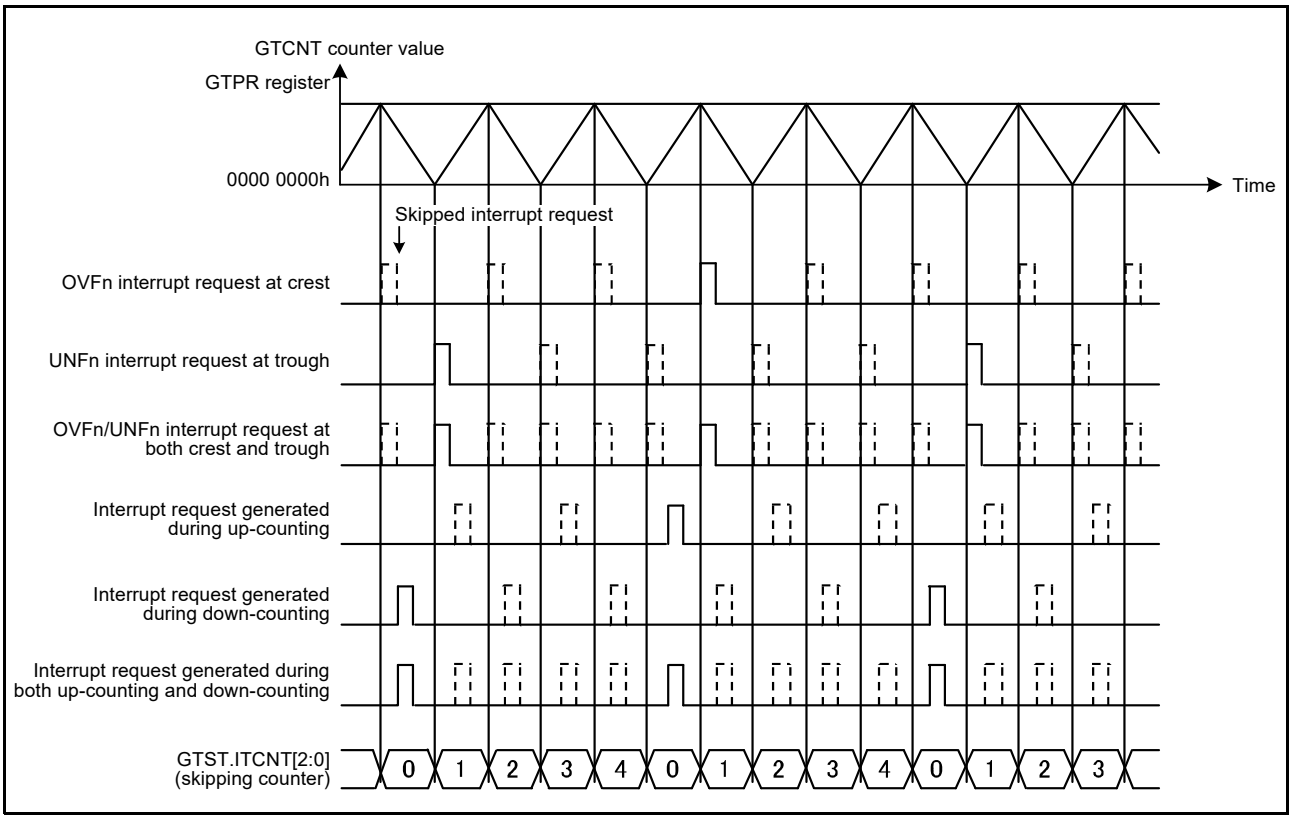


Figure 12.82 Example of interrupt skipping function operation with triangle waves, counting and skipping both troughs and crests, and skipping count = 4

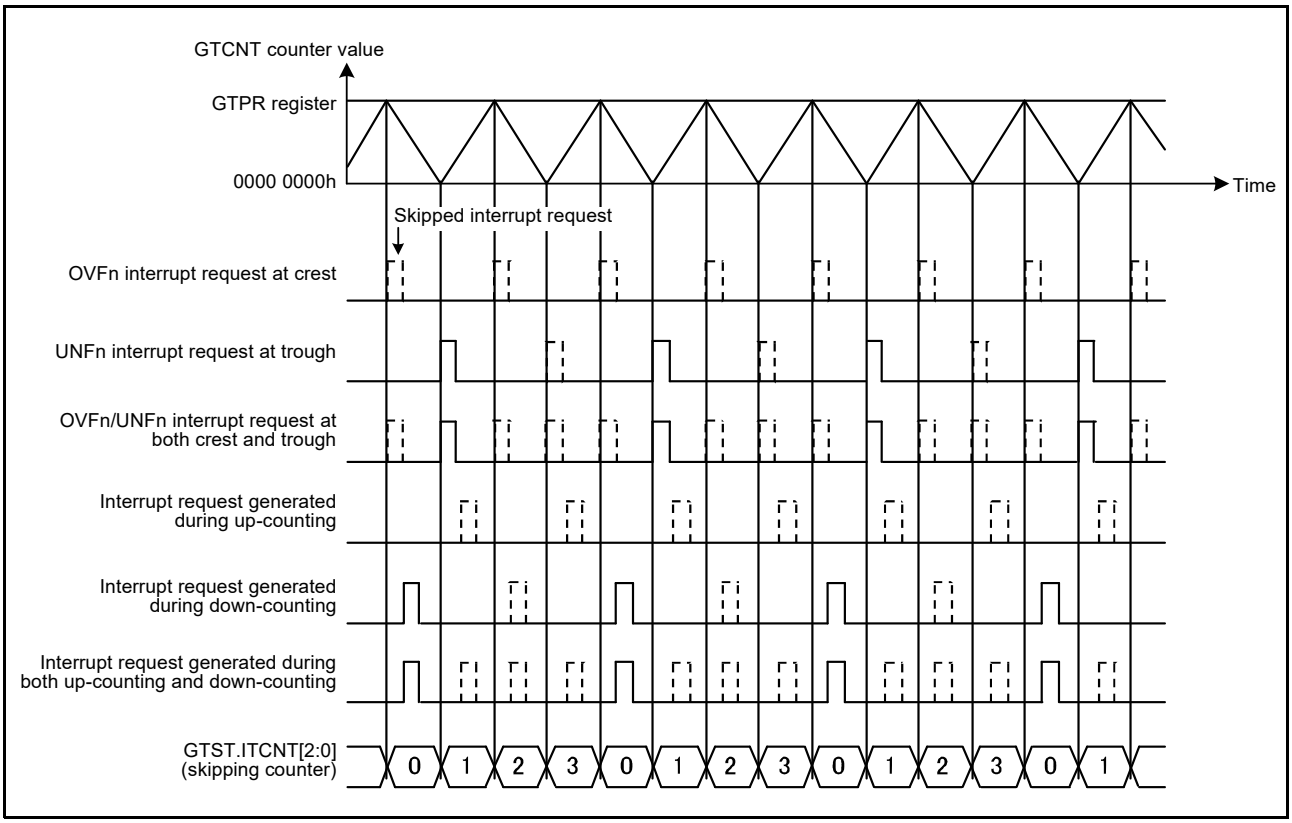


Figure 12.83 Example of interrupt skipping function operation with triangle waves, counting and skipping both troughs and crests, skipping count = 3, and skipping started at up-counting

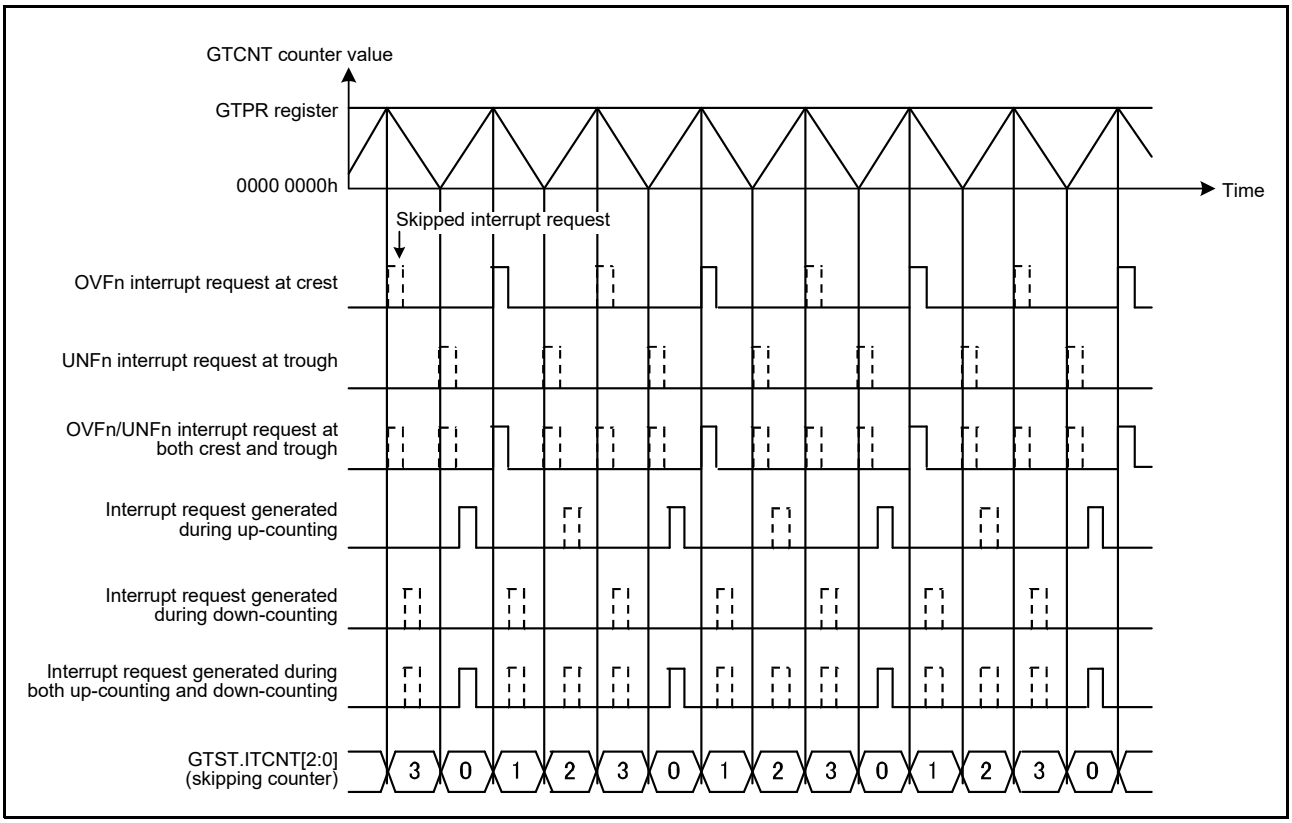


Figure 12.84 Example of interrupt skipping function operation with triangle waves, counting and skipping both troughs and crests, skipping count = 3, and skipping started at down-counting

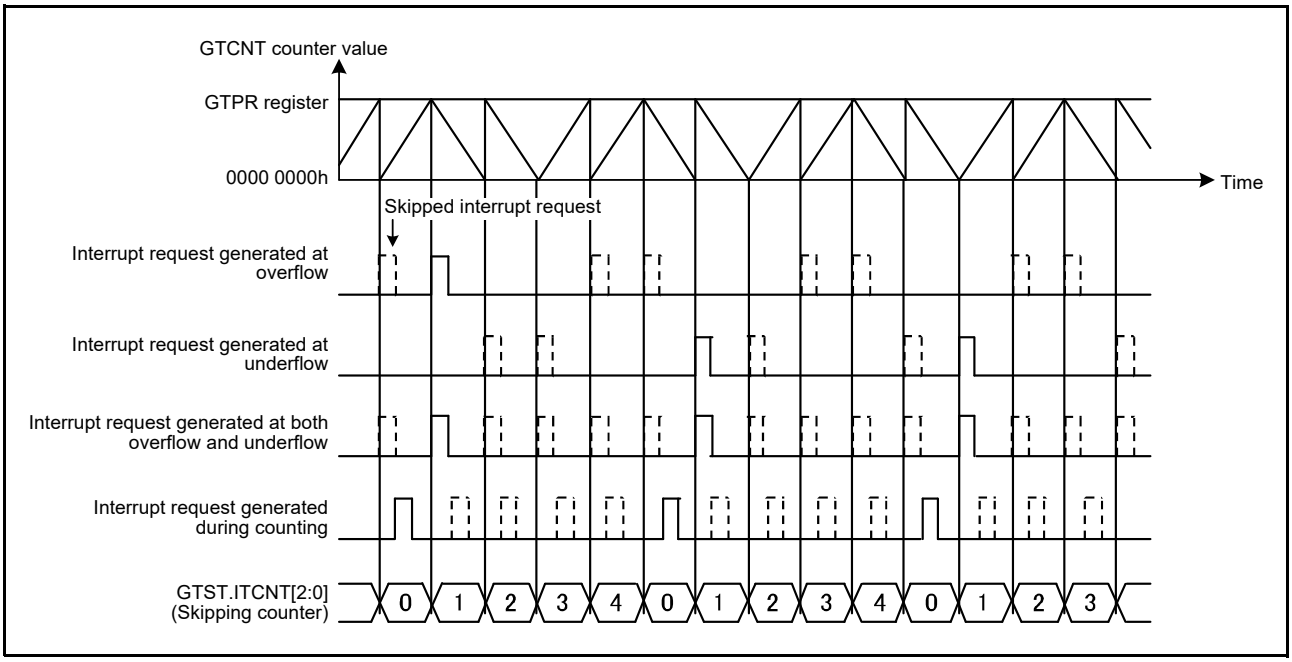


Figure 12.85 Example of interrupt skipping function operation with saw waves, operation with count direction changed, counting and skipping both overflows and underflows, and skipping count = 4

12.5 A/D Converter Start Request

An A/D converter start request can be issued at a compare match between the GTCNT counter and GTADTRA or GTADTRB, and up-counting only, down-counting only, or both up-counting and down-counting can be specified.
In event count operation performing, A/D converter start requests interrupt cannot be generated.
GTADTRA and GTADTRB each has two buffer registers. Buffer operation with GTADTRA combined with GTADTBRA and GTADTDBRA, and buffer operation with GTADTRB combined with GTADTBRB and GTADTDBRB can be performed.
Figure 12.86 shows an example of A/D converter start request operation, and Figure 12.87 shows an example setting for A/D converter start request operation.

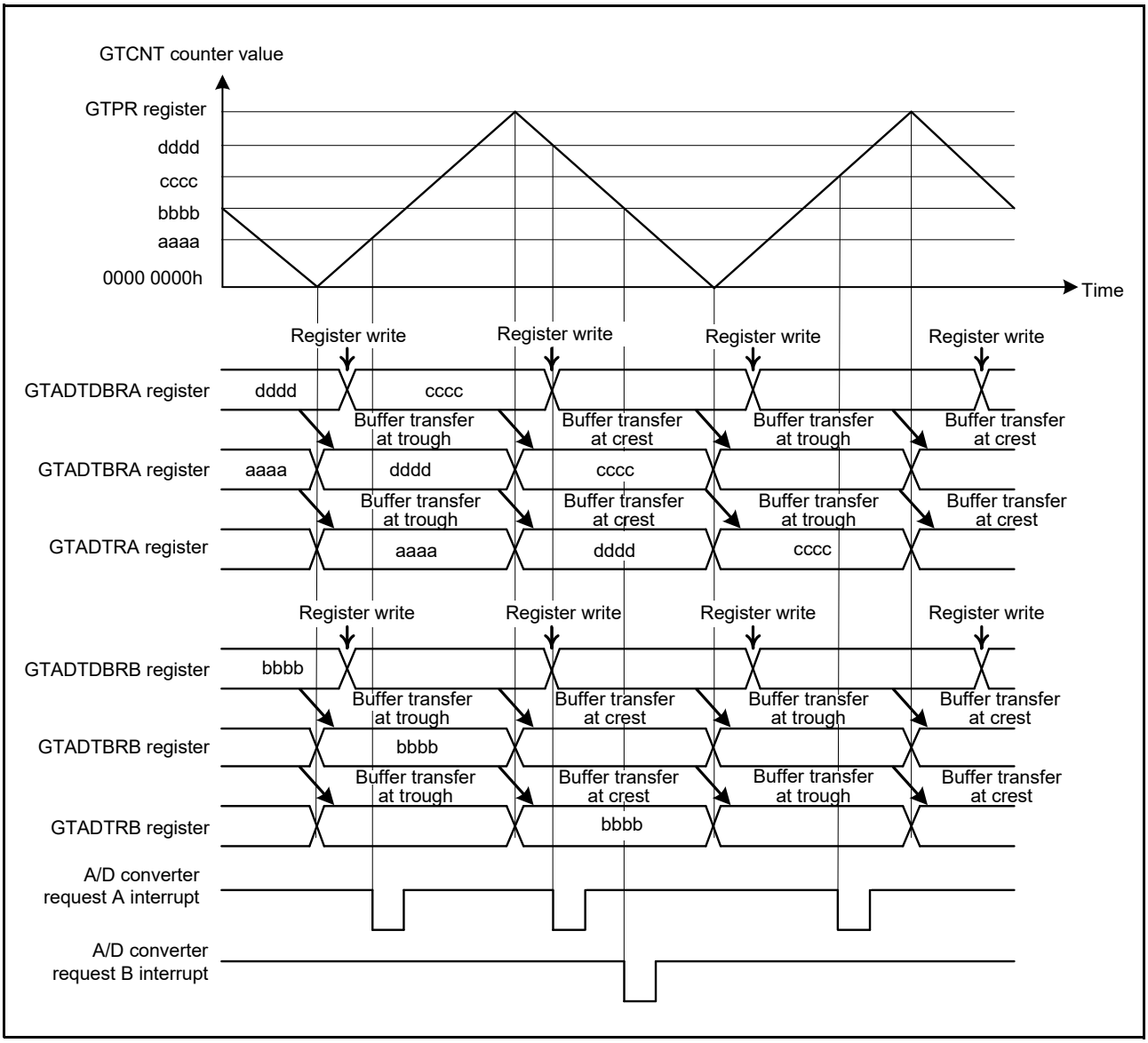


Figure 12.86 Example of A/D converter start request timing operation with triangle waves, double buffer operation, buffer transfer at both troughs and crests, A/D converter start request interrupt by GTADTRA at both up-counting and down-counting, and A/D converter start request interrupt by GTADTRB at down-counting

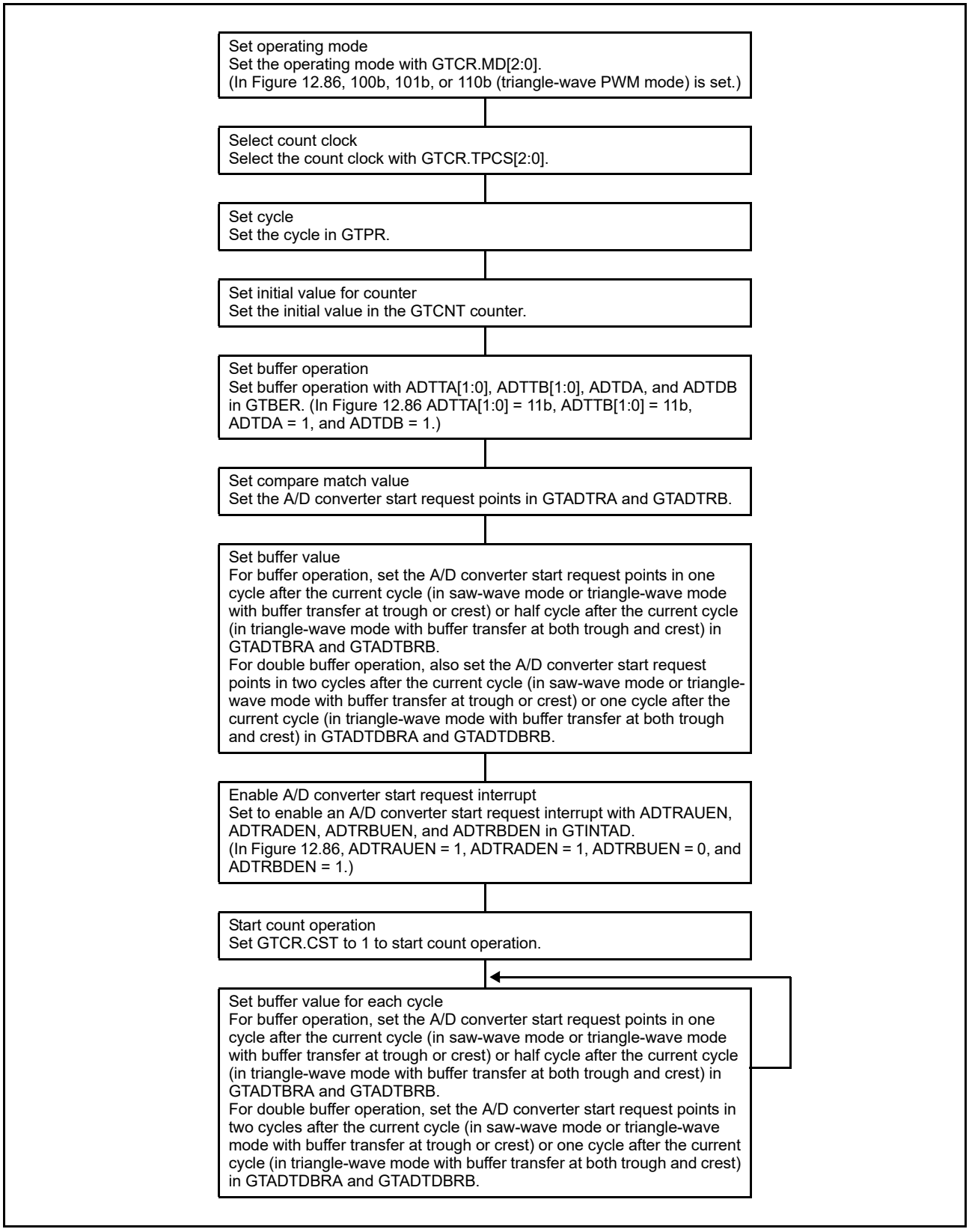


Figure 12.87 Example setting for A/D converter start request timing operation

12.6 Cooperative operation by the event

12.6.1 Generation of the event signal

The operation of GPT channels can be interlinked by using the interrupt signal generated in one channel as an event signal for another.

The GPT has the following event signals:

- Generating of compare match A interrupt (CCMPAn (n= 0 to 7))
- Generating of compare match B interrupt (CCMPBn (n= 0 to 7))
- Generating of compare match C interrupt (CMPCn (n= 0 to 7))
- Generating of compare match D interrupt (CMPDn (n= 0 to 7))
- Generating of compare match E interrupt (CMPEn (n= 0 to 7))
- Generating of compare match F interrupt (CMPFn (n= 0 to 7))
- Generating of overflow interrupt (OVFn (n= 0 to 7))
- Generating of underflow interrupt (UNFn (n= 0 to 7))

Table 12.21 shows the correspondence between the interrupt request signal and the Event Signal specified by the GTESRn registers.

Table 12.21 Correspondence between Interrupt request signal and Event Signal

Event number	Event generation channel	Event Signal
01h	GPT32E0	CCMPA0
02h		CCMPB0
03h		CMPC0
04h		CMPD0
05h		CMPE0
06h		CMPF0
07h		OVF0
08h		UNF0
09h	GPT32E1	CCMPA1
0Ah		CCMPB1
0Bh		CMPC1
0Ch		CMPD1
0Dh		CMPE1
0Eh		CMPF1
0Fh		OVF1
10h		UNF1
11h	GPT32E2	CCMPA2
12h		CCMPB2
13h		CMPC2
14h		CMPD2
15h		CMPE2
16h		CMPF2
17h		OVF2
18h		UNF2

Table 12.21 Correspondence between Interrupt request signal and Event Signal

Event number	Event generation channel	Event Signal
19h	GPT32E3	CCMPA3
1Ah		CCMPB3
1Bh		CMPC3
1Ch		CMPD3
1Dh		CMPE3
1Eh		CMPF3
1Fh		OVF3
20h		UNF3
21h	GPT32E4	CCMPA4
22h		CCMPB4
23h		CMPC4
24h		CMPD4
25h		CMPE4
26h		CMPF4
27h		OVF4
28h		UNF4
29h	GPT32E5	CCMPA5
2Ah		CCMPB5
2Bh		CMPC5
2Ch		CMPD5
2Dh		CMPE5
2Eh		CMPF5
2Fh		OVF5
30h		UNF5
31h	GPT32E6	CCMPA6
32h		CCMPB6
33h		CMPC6
34h		CMPD6
35h		CMPE6
36h		CMPF6
37h		OVF6
38h		UNF6
39h	GPT32E7	CCMPA7
3Ah		CCMPB7
3Bh		CMPC7
3Ch		CMPD7
3Dh		CMPE7
3Eh		CMPF7
3Fh		OVF7
40h		UNF7

12.6.2 Operation on event signal

The GPT can perform the following operations in response to a maximum of eight events:

- Start counting, stop counting, clear counting
- Up-counting, down counting
- Input capture.

See section 12.3, Operation for detail on hardware resources.

12.6.3 Example of procedure for setting cooperative action by event

The coordinated operation method is as follows.

1. Configure the operation of event input.
2. Set the interrupt request signals which are the event input in the GTESRn registers.
3. Sets the GTECR.EVCON bit to 1 for enabling all event signals.
4. Activate GPT. As a result, the cooperation action of the event becomes active.
5. To disable the event signal, set the corresponding GTESRn.EVS[6:0] bits to 0000000b. In addition, by making the GTECR.EVCON bit to 0 to disable all of the event signal.

12.7 Noise Filter Function

Each pin for use in input capture to the GPT is equipped with a noise filter. The noise filter samples input signals at the sampling clock and removes the pulses whose length is less than three sampling cycles.
The noise filter functionality includes enabling and disabling the noise filter for each pin and setting of the sampling clock for each channel.
Figure 12.88 shows the timing of noise filtering.

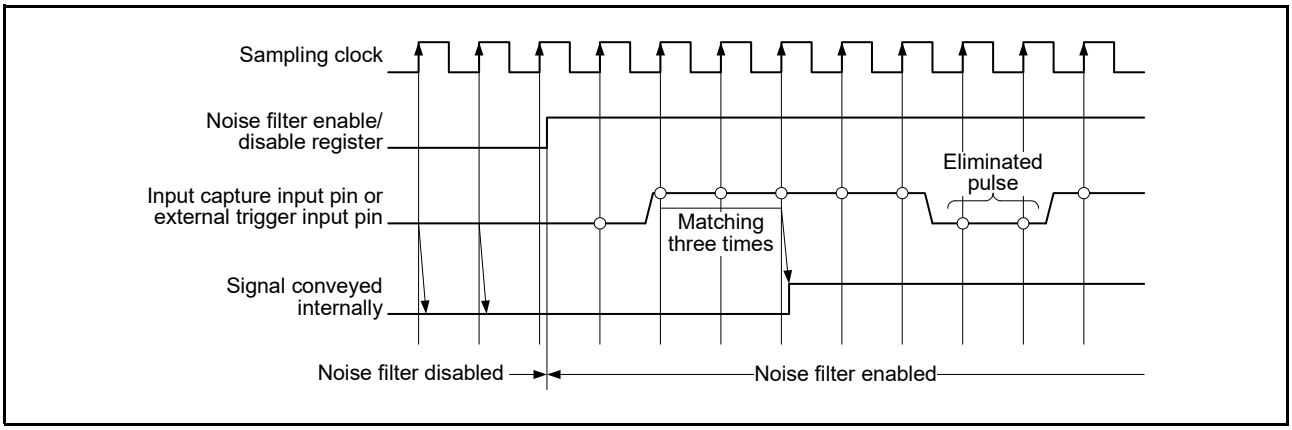


Figure 12.88 Timing of noise filtering

If noise filtering is enabled, the input capture operation or external trigger operation performs on the edges of the noise filtered signal after a delay of a sampling interval × 3 + P1φ. This is caused by the noise filtering for the input capture input or external trigger operation.

12.8 Protection Function

12.8.1 Write-Protection for Registers

To prevent registers from being accidentally modified, registers can be write-protected in channel units by setting GTWP.WP. Write-protection can be set for the following registers:
GTSSR, GTPSR, GTCSR, GTUPSR, GTDNSR, GTICASR, GTICBSR, GTCR, GTUDDTYC, GTIOR, GTINTAD,GTST, GTBER, GTITC, GTCNT, GTCCRA, GTCCRB, GTCCRC, GTCCRD, GTCCRE, GTCCRF, GTPR, GTPBR, GTPDBR, GTADTRA, GTADTBRA, GTADTDBRA, GTADTRB, GTADTBRB, GTADTDBRB, GTDTCR, GTDVU, GTDVD, GTDBU, GTDBD, GTSOS, GTSOTR.

12.8.2 Disabling of Buffer Operation

If the timing of buffer register write is delayed in relative to the timing for the buffer transfer, buffer operation can be suspended with the GTBER.BD setting. Buffer transfer can be temporarily disabled even when a buffer transfer condition is generated during a buffer register write. This can be done by setting the associated GTBER.BD bit to 1 (buffer operation disabled) before a buffer register write and clearing the bit to 0 (buffer operation enabled) after completion of writing to all buffer registers. Figure 12.89 shows an example of operation for disabling buffer operation.

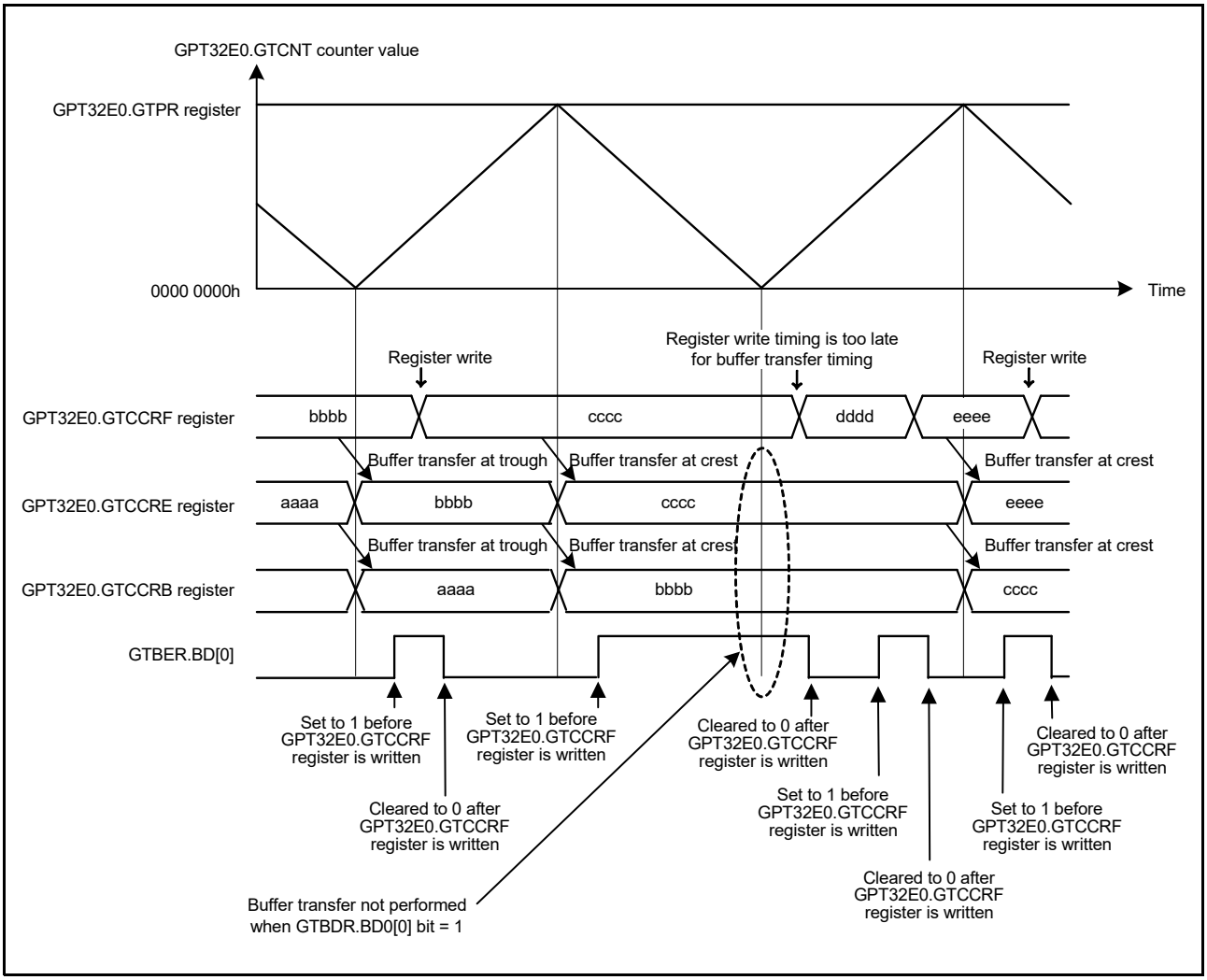


Figure 12.89 Example of operation for disabling buffer operation with triangle waves, double buffer operation, and buffer transfer at both troughs and crests

12.8.3 GTIOC Pin Output Negate Control

For protection from system failure, the output disable control that changes the GTIOC pin output value forcibly is provided for GTIOC pin output by the request of output disable from POEG.

When dead time error occurs or the GTIOCA pin output value is the same as the GTIOCB pin output value, output protection is required. GPT detects this condition and generates output disable requests to POEG based on the settings in the output disable request permission bits, such as GTINTAD.GRPDTE, GTINTAD.GRPABH, GTINTAD.GRPABL. After the POEG receives output disable requests from each channel and calculates external input using an OR operation, the POEG generates output disable requests to GPT.

One output disable signal (representing the shared output disable request signal of the GTIOCA pin and the GTIOCB pin) out of four* output disable requests generated by the POEG is selected by setting GTINTAD.GRP[1:0]. The status of the selected disable output request is monitored by reading the GTST.ODF bit. The output level during output disable is based on the GTIOR.OADF[1:0] setting for the GTIOCA pin and the GTIOR.OBDF[1:0] setting for the GTIOCB pin.

The change to the output disable state is performed asynchronously by generating the output disable request from the POEG. The release of the output disable state is performed at end of cycle by terminating the output disable request. The timing of release of the output disable state is a minimum of 3 P1φ cycles after terminating the output disable request. To perform output disable control reliably, allow at least 4 P1φ cycles after generating the output disable request (by clearing the output disable request flag in POEG) until the output disable request is terminated.

When event count is performed or when the output disable state is to be released immediately without waiting for an end of cycle, GTIOR.OADF[1:0] must be set to 00b (for GTIOCA pin) or GTIOR.OBDF[1:0] must be set to 00b (for GTIOCB pin).

Figure 12.90 shows an example of the GTIOC pin output disable control operation.

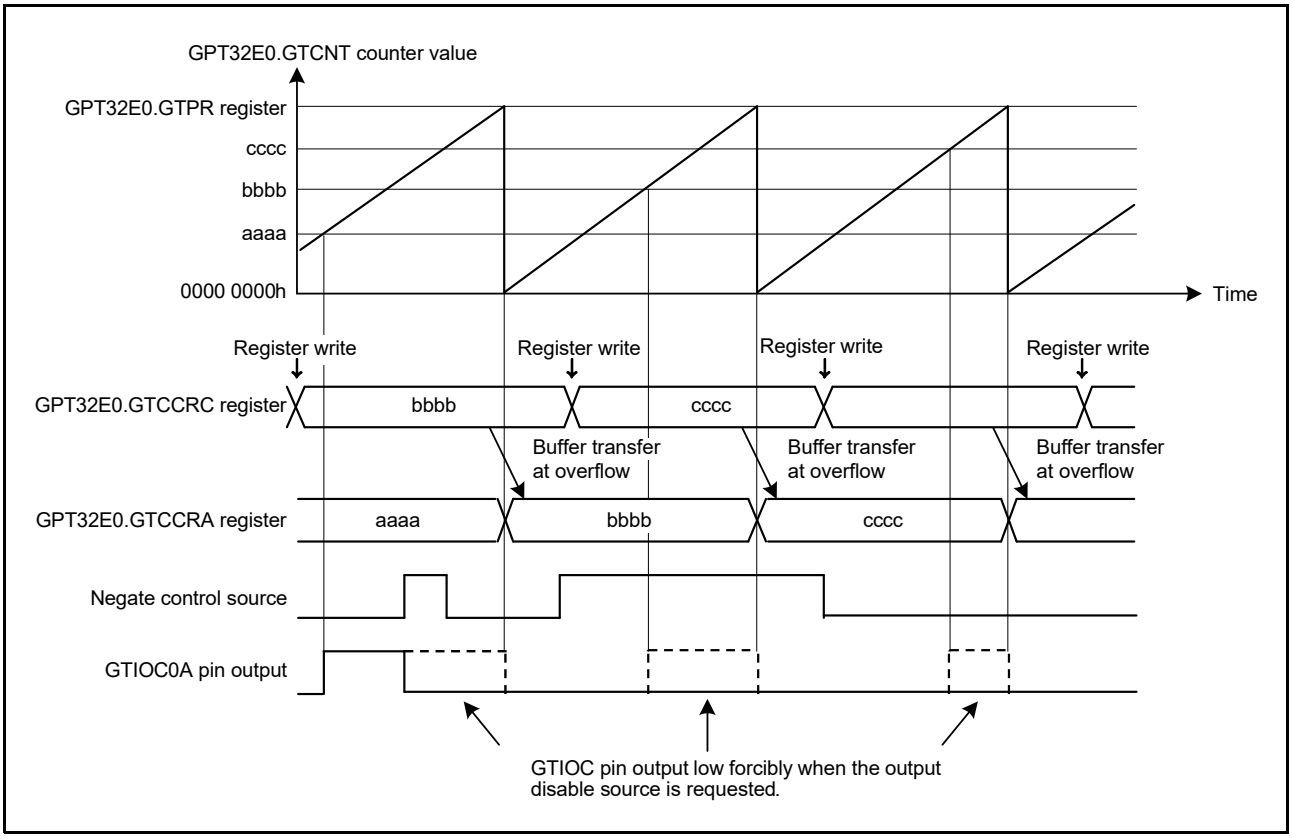


Figure 12.90 Example of GTIOC pin output disable control operation with saw-wave up-counting, buffer operation, active level 1, high output at GTCCRA compare match, low output at cycle end, and low output at output disable

12.8.4 Output Protection Function for GTIOC Pin Output

In preparation for incorrect settings of the GTCCRA register (settings outside the range of 0 < GTCCRA < GTPR), the output protection function for the GTIOC pin output (disabling function) is activated when the automatic dead time setting (GTDTCR.TDE = 1) is made in triangle-wave mode. The status of the output protection function can be read from GTSOS.SOS[1:0].

Figure 12.91 shows the output protection function state transition.

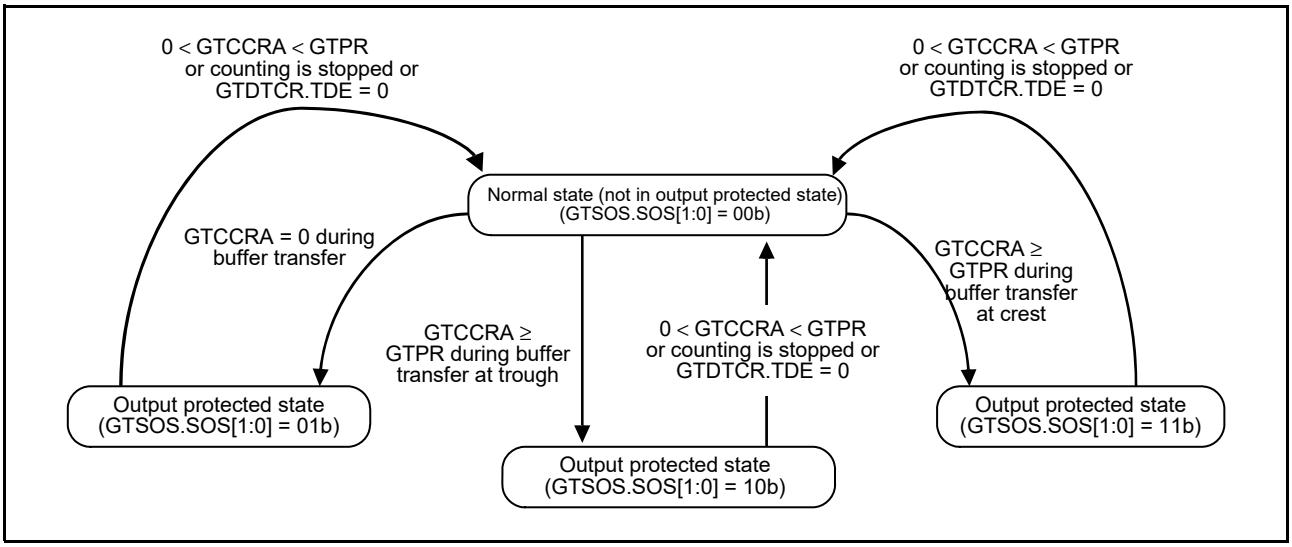
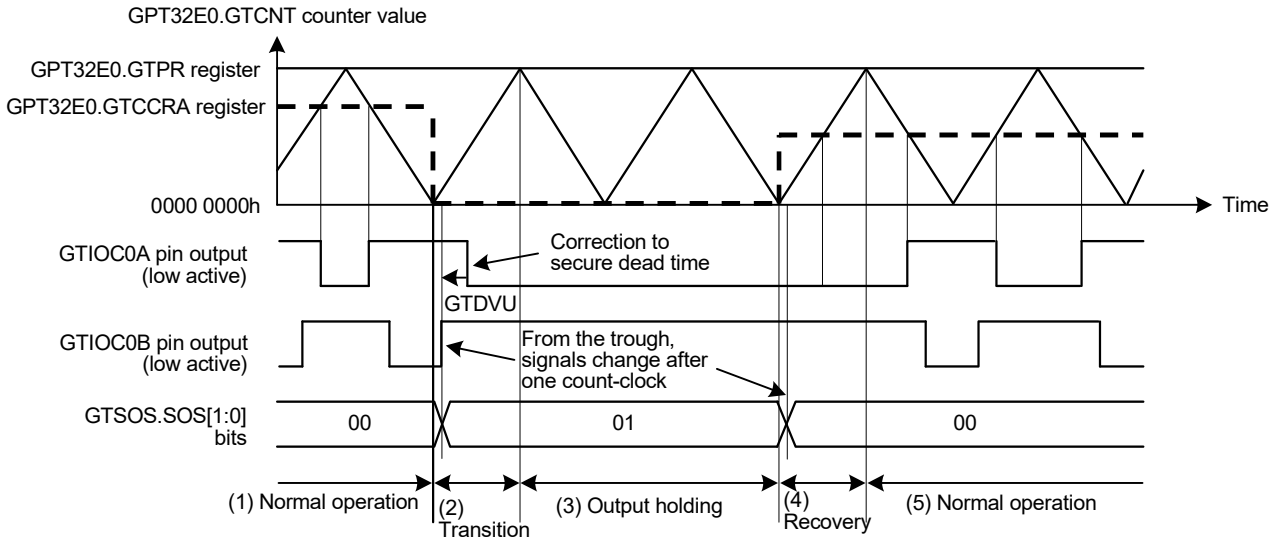


Figure 12.91 Output protection function

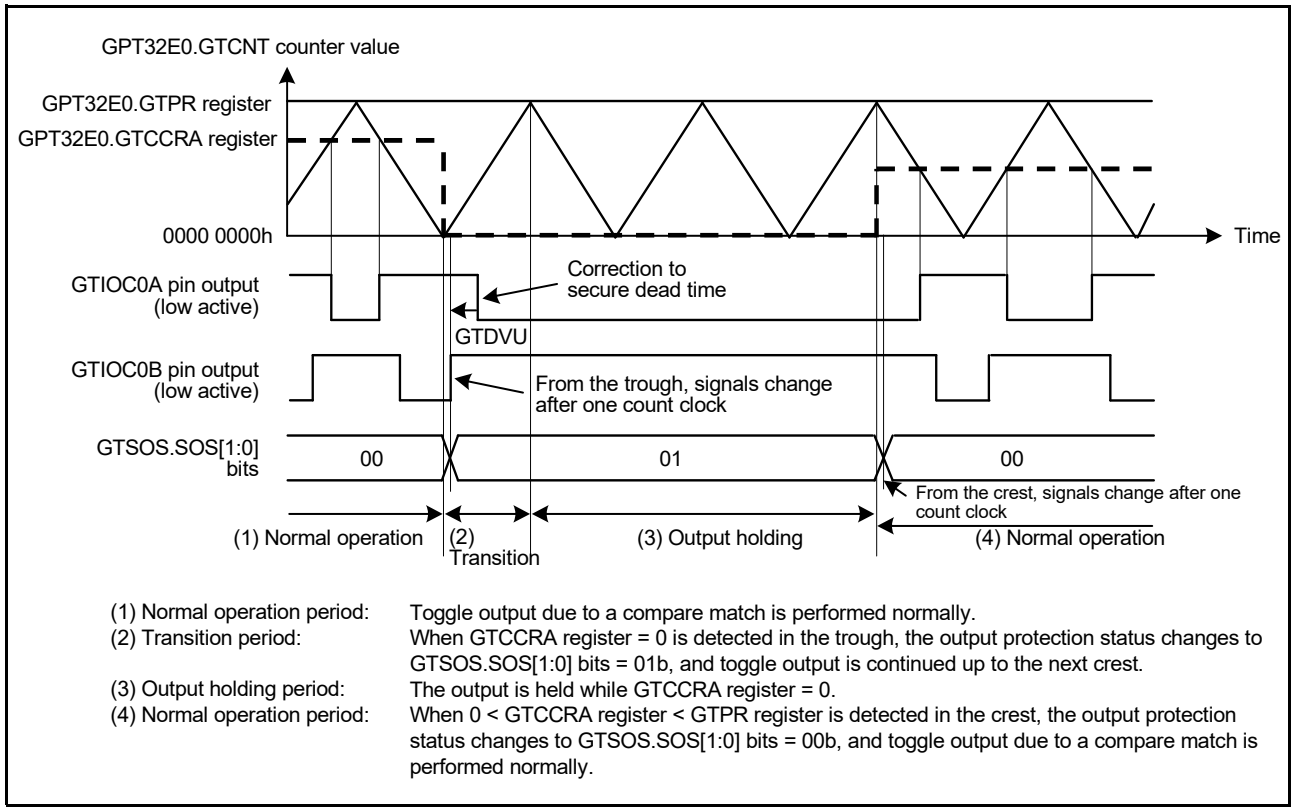
12.8.4.1 Output Protection Function When the GTCCRA Register is Set to 0 during Buffer Transfer

Figure 12.92 and Figure 12.93 show examples of output protection function operation when the GTCCRA register is set to 0 during buffer transfer at troughs, and Figure 12.94 and Figure 12.95 show examples when the GTCCRA register is set to 0 during buffer transfer at crests.



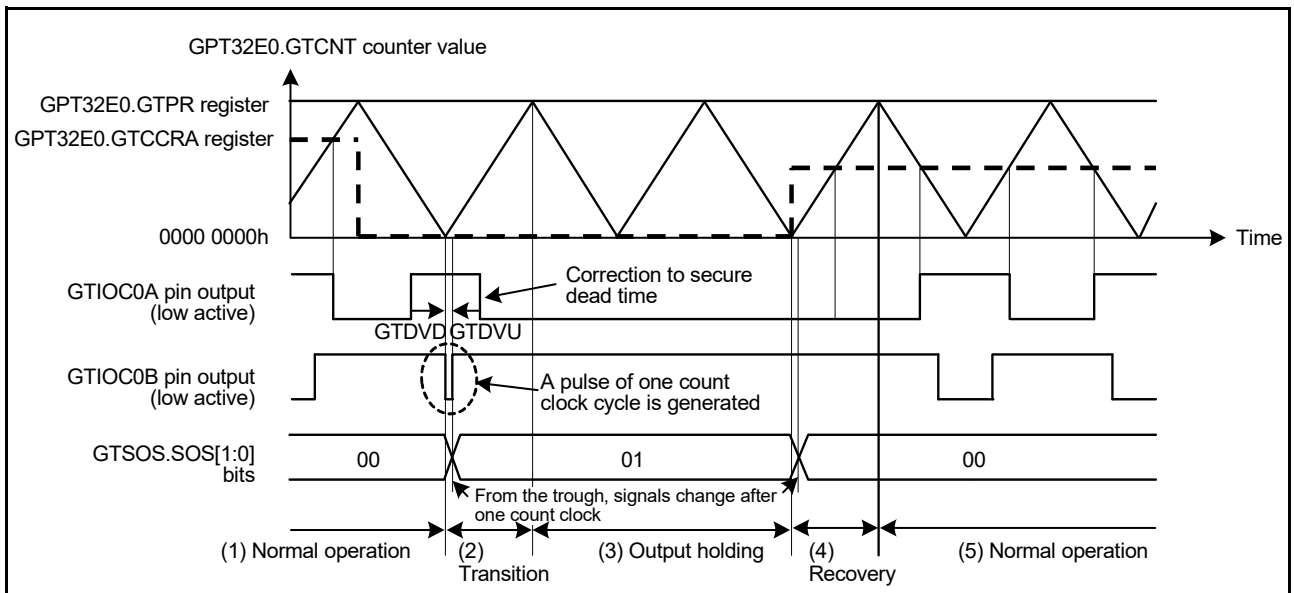
(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Transition period: When GTCCRA register = 0 is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 01b, and toggle output is continued up to the next crest
(3) Output holding period: The output is held while GTCCRA register = 0.
(4) Recovery period: When 0 < GTCCRA register < GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and holding output is maintained up to the next crest.
(5) Normal operation period: Toggle output due to a compare match is performed normally.

Figure 12.92 Example of output protection operation when GTCCRA is set to 0 during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low



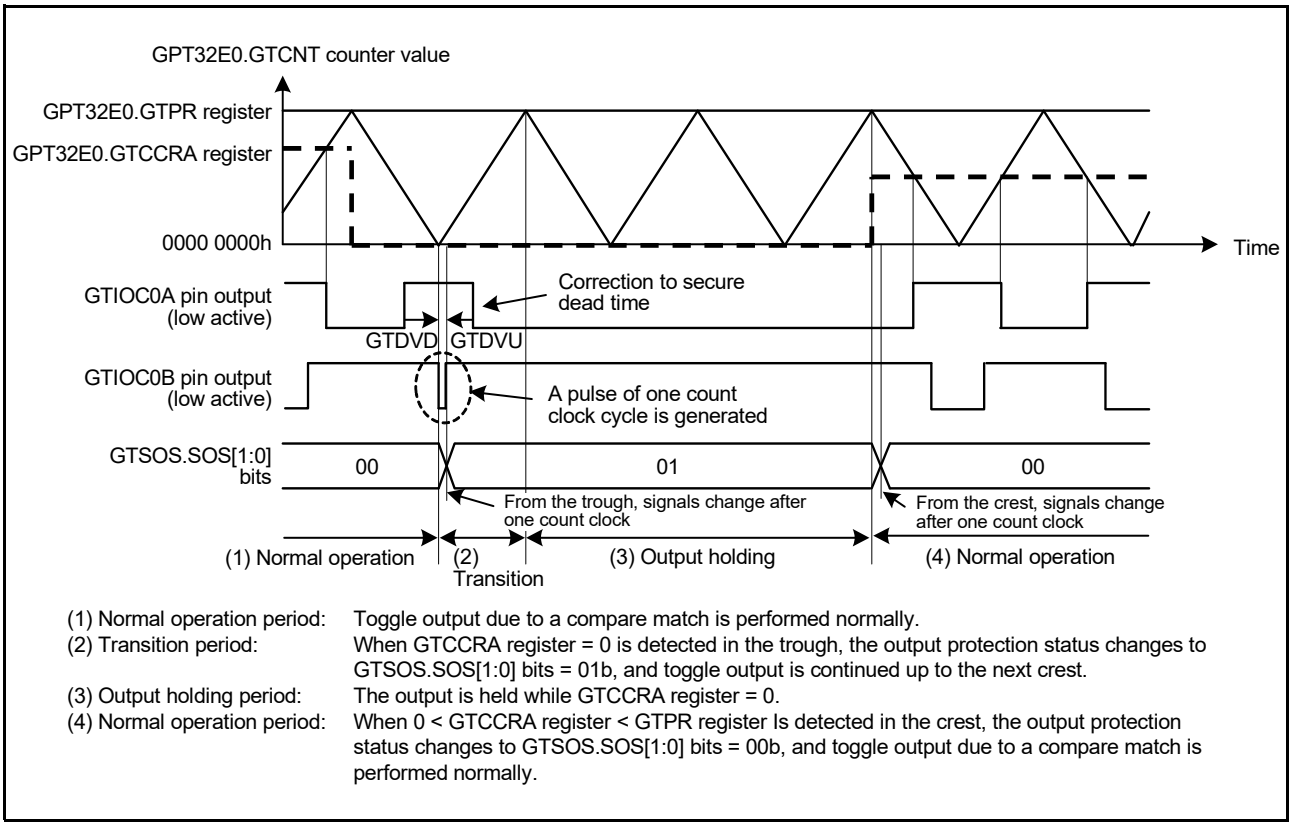
(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Transition period: When GTCCRA register = 0 is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 01b, and toggle output is continued up to the next crest.
(3) Output holding period: The output is held while GTCCRA register = 0.
(4) Normal operation period: When 0 < GTCCRA register < GTPR register is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and toggle output due to a compare match is performed normally.

Figure 12.93 Example of output protection operation when GTCCRA is set to 0 during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at crests, and active-low



(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Transition period: When GTCCRA register = 0 is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 01b, and toggle output is continued up to the next crest.
(3) Output holding period: The output is held while GTCCRA register = 0.
(4) Recovery period: When 0 < GTCCRA register < GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and holding output is maintained up to the next crest.
(5) Normal operation period: Toggle output due to a compare match is performed normally.

Figure 12.94 Example of output protection operation when GTCCRA is set to 0 during buffer transfer at crests, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low



(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Transition period: When GTCCRA register = 0 is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 01b, and toggle output is continued up to the next crest.
(3) Output holding period: The output is held while GTCCRA register = 0.
(4) Normal operation period: When 0 < GTCCRA register < GTPR register Is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and toggle output due to a compare match is performed normally.

Figure 12.95 Example of output protection operation when GTCCRA is set to 0 during buffer transfer at crests, with 0 < GTCCRA < GTPR restored during buffer transfer at crests, and active-low

12.8.4.2 Output Protection Function When GTCCRA ≥ GTPR is Set during Buffer Transfer at Troughs

Figure 12.96 and Figure 12.97 show examples of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs.

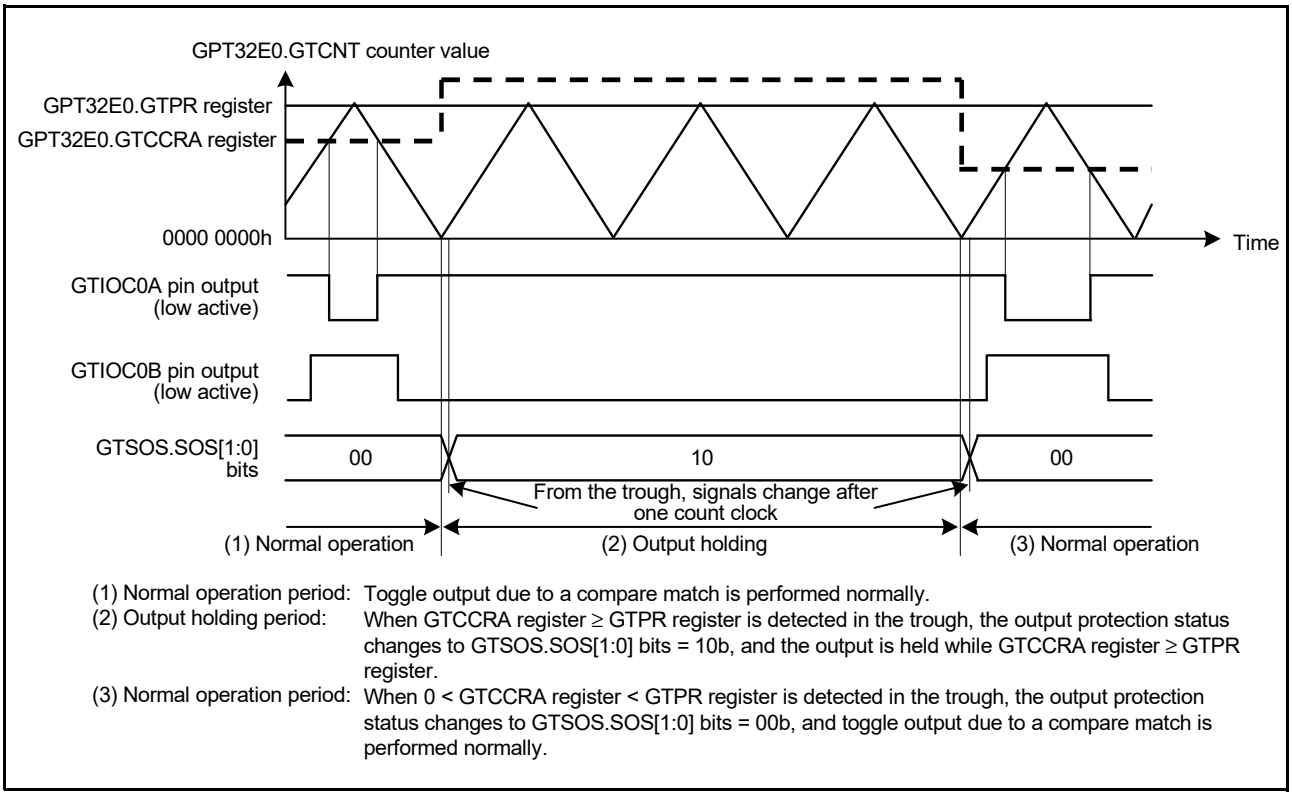
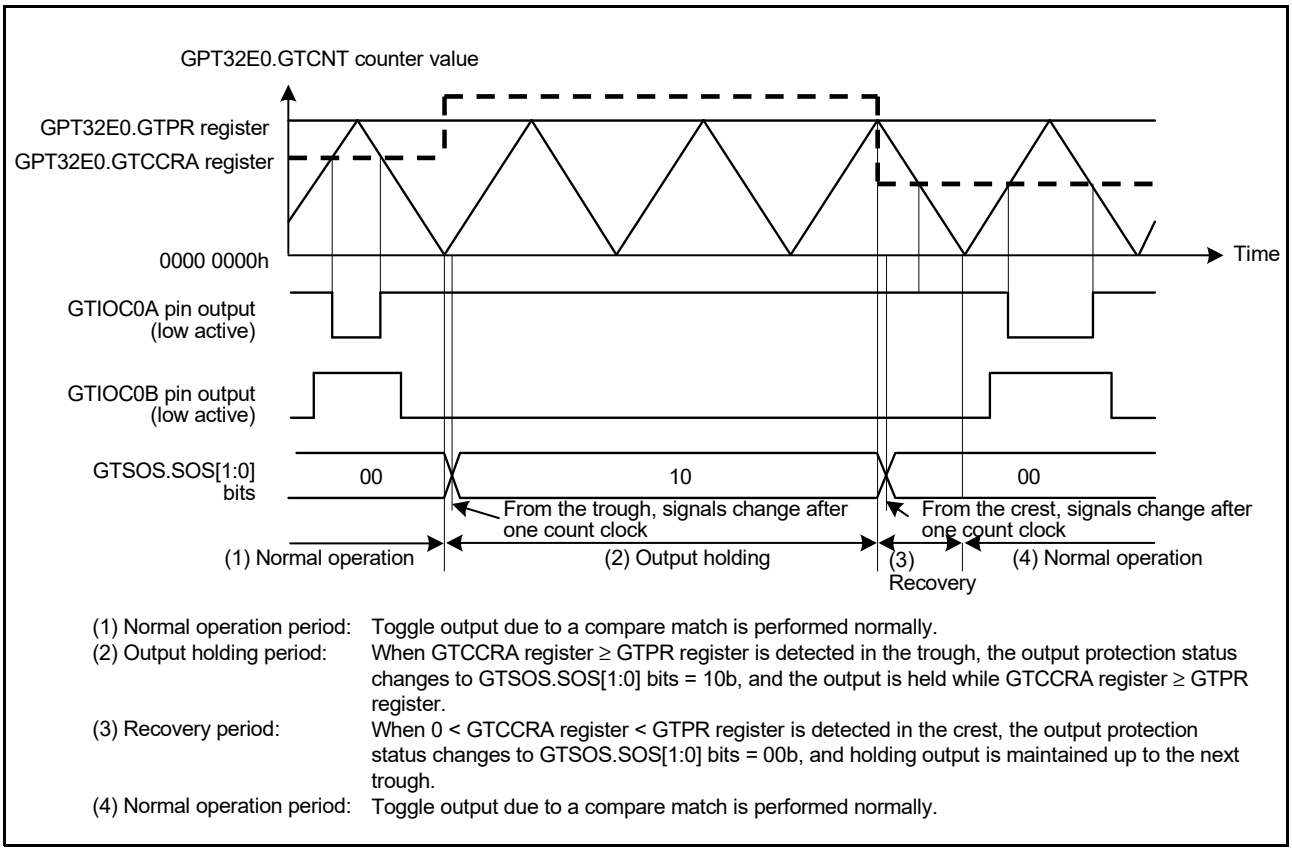


Figure 12.96 Example of output protection operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low



(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Output holding period: When GTCCRA register ≥ GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 10b, and the output is held while GTCCRA register ≥ GTPR register.
(3) Recovery period: When 0 < GTCCRA register < GTPR register is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and holding output is maintained up to the next trough.
(4) Normal operation period: Toggle output due to a compare match is performed normally.

Figure 12.97 Example of output protection operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at crests, and active-low

12.8.4.3 Output Protection Function When GTCCRA ≥ GTPR is Set during Buffer Transfer at Crests

Figure 12.98 and Figure 12.99 show examples of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at crests.

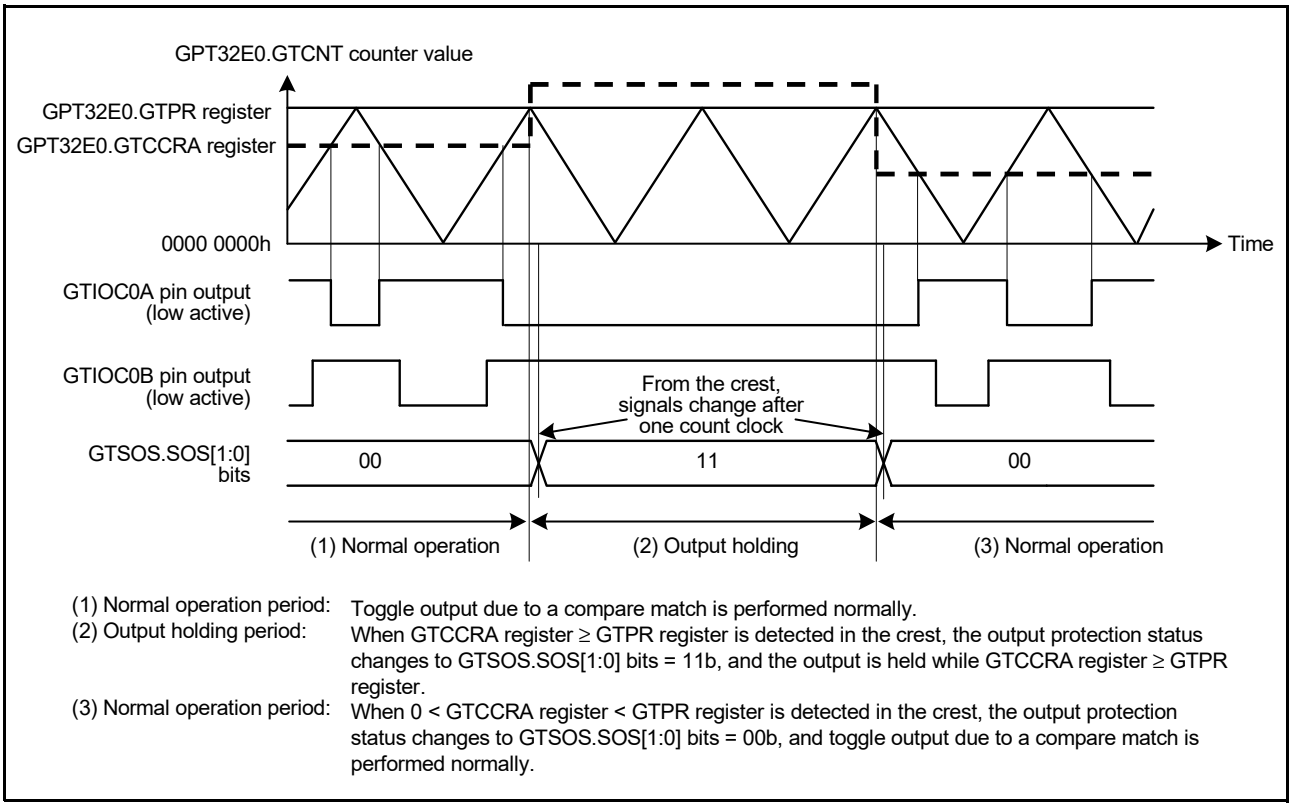


Figure 12.98 Example of output protection operation when GTCCRA ≥ GTPR is set during buffer transfer at crests, with 0 < GTCCRA < GTPR restored during buffer transfer at crests, and active- low

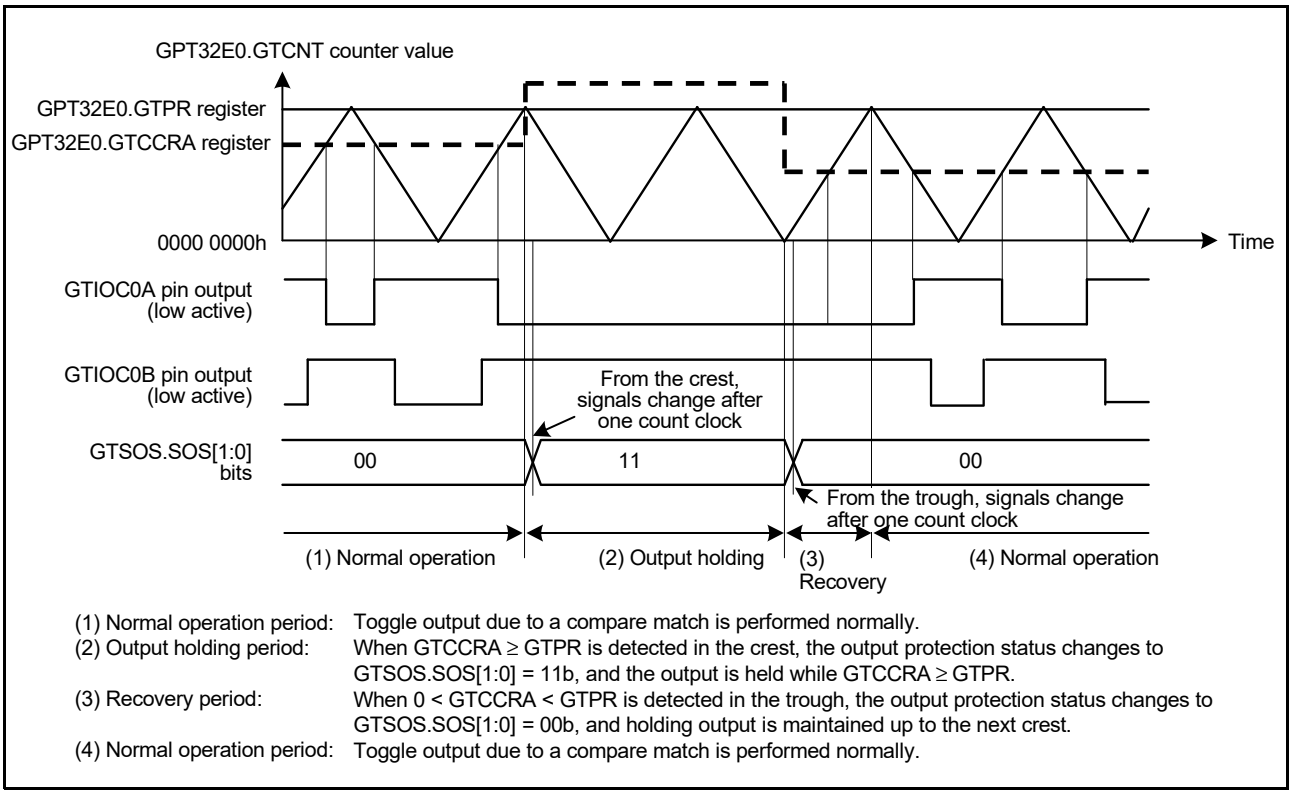


Figure 12.99 Example of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at crests, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low

12.8.4.4 Restricted Specification of Output Protection Function

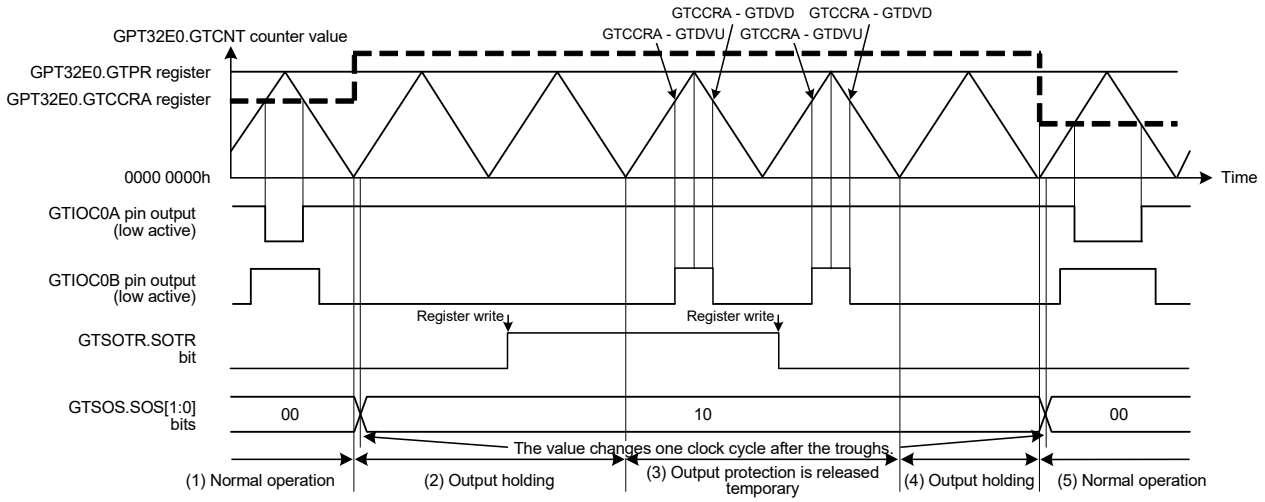
The output protection function deactivates the level of one of the positive and negative outputs, even if an incorrect value is set in the GTCCRA register during counting (a setting outside the range of 0 < GTCCRA < GTPR). However, it will not work properly unless the following conditions are satisfied.

- At count start: The value of the GTCCRA register must be set within the range of (0 < GTCCRA < GTPR).
- During buffer transfer at crest: GTCCRA - GTDVD < GTPR - 1 has to be satisfied..
- During buffer transfer at trough: If GTCCRA ≥ GTPR, GTCCRA - GTDVU > 1 has to be satisfied.

12.8.4.5 Temporary cancellation of Output Protection Function

When the GTSOTR.SOTR bit is set to 1 with GTSOS.SOS[1:0] bits equal to 10b (showing output protection state by GTCCRA ≥ GTPR during buffer transfer at troughs), the output protection function for GTIOCB pin is temporarily canceled. GTSOS.SOS[1:0] bits retain the value of 10b even when the output protection function is canceled. When the SOTR bit is set to 0, the output protection function for GTIOCB pin resumes.

Figure 12.100 shows examples of temporary cancellation of output protection function operation when the GTCCRA ≥ GTPR is set during buffer transfer at troughs.



(1) Normal operation period:	Toggle output due to a compare match is performed normally.
(2) Output holding period:	When GTCCRA register ≥ GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 10b, and the output is held while GTCCRA register ≥ GTPR register. If GTSOTR.SOTR is set to 1 during the output holding period, the GTIOCB output is holding until the first trough after the setting.
(3) Output protection releasing period:	After GTSOTR.SOTR is set to 1, the GTIOCB output is performed normally at the first trough after the setting. If GTSOTR.SOTR is set to 0 during the output protection releasing period, the output is holding at the first trough after the setting.
(4) Output holding period:	Until 0 < GTCCRA register < GTPR register is detected in the trough, the output is holding.
(5) Normal operation period:	When 0 < GTCCRA register < GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and toggle output due to a compare match is performed normally.

Figure 12.100 Example of temporary cancellation of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low

12.9 Initialization Method of Output Pins

12.9.1 Pin Settings after Reset

The GPT registers are initialized at a reset. Start counting after selecting the port mode, setting GTIOR.OAE and GTIOR.OBE bits, and outputting the GPT function to external pins.

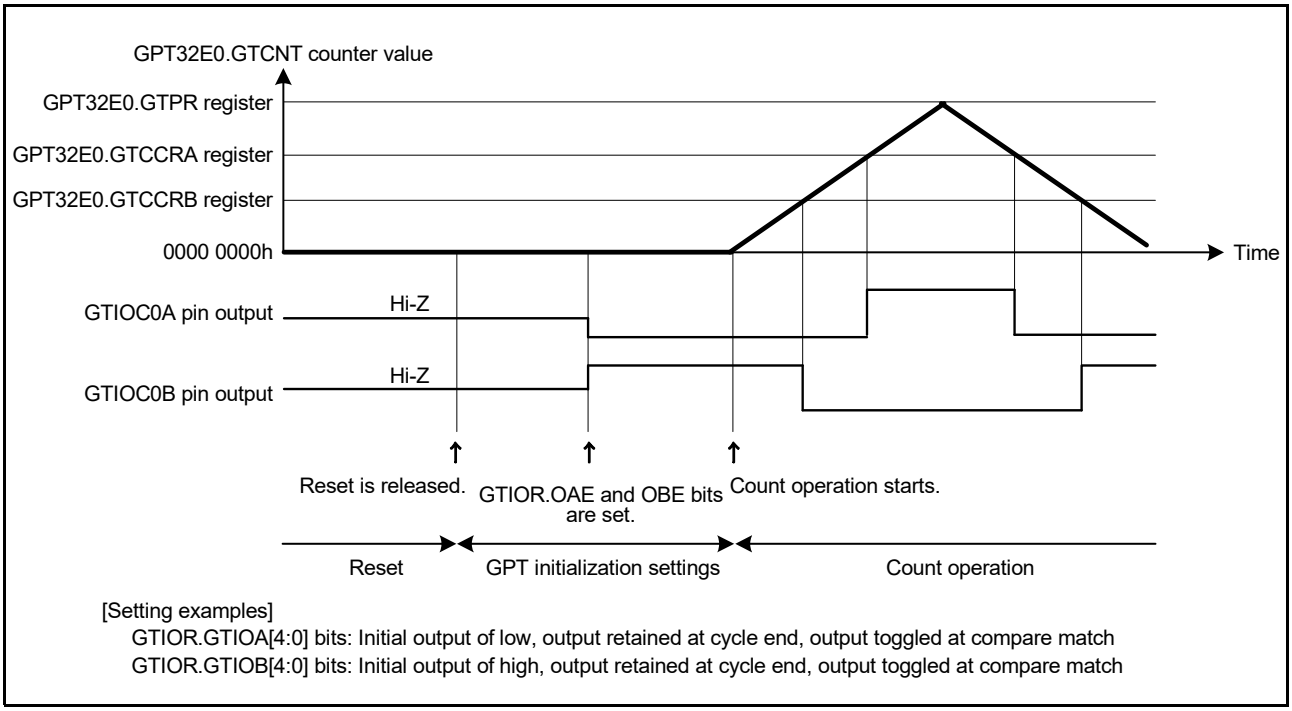


Figure 12.101 Example of pin settings after reset

12.9.2 Pin Initialization Caused by Error during Operation

If an error occurs during GPT operation, the following four types of pin processing can be performed before pin initialization:

- Set the OAHLD and OBHLD bits in GTIOR to 1 and retain the outputs at count stop.
- Set the OAHLD and OBHLD bits in GTIOR to 0, specify arbitrary output values in OADFLT and OBDFLT in GTIOR, and output the arbitrary values on count stop.
- Set the pin to output an arbitrary value as a general output port by setting the PDR, PODR, and PMR registers of the I/O port in advance. Set the OAE and OBE bits in GTIOR to 0 and the control bit associated with the pin in the PMR to 0, to allow arbitrary values to be output from the pin set as a general output port when an error occurs.
- Drive the output to a high impedance state using the POEG function.

When the automatic dead time setting is made, clear the GTDTCR.TDE bit to 0 after counting stops. When counting stops, only the values of registers that are changed by a GPT external source change. If counting resumes, operation continues from where it stops. If counting stops, registers must be initialized before counting starts.

12.10 Usage Notes

12.10.1 Settings for the Module-Stop Function

GPT operation can be disabled or enabled using the standby control register. The GPT module is initially stopped after reset. Releasing the module-stop state enables access to the registers. For details, see section 52, Power-Down Modes.

12.10.2 GTCCRn Settings during Compare Match Operation (n = A to F)

(1) When automatic dead time setting is made in triangle-wave PWM mode

The GTCCRA register must satisfy the following conditions: GTDVU < GTCCRA, GTDVD < GTCCRA, and GTCCRA < GTPR.
When the setting of GTCCRA = 0 or GTCCRA ≥ GTPR is made during count operation, the output protection function is activated.
However, it will not work properly unless the following conditions are satisfied.

- At count start: The value of the GTCCRA register must be set within the range of (0 < GTCCRA < GTPR).
- During buffer transfer at crest: GTCCRA - GTDVD < GTPR - 1 has to be satisfied..
- During buffer transfer at through: If GTCCRA ≥ GTPR, GTCCRA - GTDVU > 1 has to be satisfied.

For details, see section 12.8.4, Output Protection Function for GTIOC Pin Output.

(2) When automatic dead time setting is not made in triangle-wave PWM mode

The GTCCRA register must be set within the range of 0 < GTCCRA < GTPR. If GTCCRA = 0 or GTCCRA = GTPR is set, a compare match occurs within the cycle only when GTCCRA = 0 or GTCCRA = GTPR is satisfied. When GTCCRA > GTPR, no compare match occurs.
Similarly, GTCCRB must be set within the range of 0 < GTCCRB < GTPR. If GTCCRB = 0 or GTCCRA = GTPR is set, a compare match occurs within the cycle only when GTCCRB = 0 or GTCCRB = GTPR is satisfied. When GTCCRB > GTPR, no compare match occurs.

(3) When automatic dead time setting is made in saw-wave one-shot pulse mode

The GTCCRC and GTCCRD registers must be set to satisfy the following constraints. If the constraints are not satisfied, correct output waveforms with secured dead time might not be obtained.

- In up-counting: GTCCRC < GTCCRD, GTCCRC > GTDVU, GTCCRD < GTPR - GTDVD
- In down-counting: GTCCRC > GTCCRD, GTCCRC < GTPR – GTDVU, GTCCRD > GTDVD

(4) When automatic dead time setting is not made in saw-wave one-shot pulse mode

The GTCCRC and GTCCRD registers must be set to satisfy the following constraints. If the constraints are not satisfied, two compare matches do not occur and pulse output cannot be performed.

- In up-counting: 0 < GTCCRC < GTCCRD < GTPR
- In down-counting: GTPR > GTCCRC > GTCCRD > 0

Similarly, GTCCRE and GTCCRF must be set to satisfy the following constraints. If the constraints are not satisfied, two compare matches do not occur and pulse output cannot be performed.

- In up-counting: 0 < GTCCRE < GTCCRF < GTPR
- In down-counting: GTPR > GTCCRE > GTCCRF > 0

(5) In saw-wave PWM mode

The GTCCRA register must be set with the range of 0 < GTCCRA < GTPR. If GTCCRA = 0 or GTCCRA = GTPR is set, a compare match occurs within the cycle only when GTCCRA = 0 or GTCCRA = GTPR is satisfied. If GTCCRA > GTPR is set, no compare match occurs.
Similarly, GTCCRB must be set with the range of 0 < GTCCRB < GTPR. If GTCCRB = 0 or GTCCRB = GTPR is set, a

compare match occurs within the cycle only when GTCCRB = 0 or GTCCRB = GTPR is satisfied. If GTCCRB > GTPR is set, no compare match occurs.

12.10.3 Setting Range for the GTCNT Counter

The GTCNT counter register must be set with the range of 0 ≤ GTCNT ≤ GTPR.

12.10.4 Starting and Stopping the GTCNT Counter

The control timing of starting and stopping the GTCNT counter by the GTCR.CST bit synchronizes the count clock that is selected in GTCR.TPCS[2:0]. When GTCR.CST is updated, the GTCNT counter starts/stops after a count clock that is selected in GTCR.TPCS[2:0]. Therefore, an event generated before the GTCNT counter actually starts is ignored. Note, however, that an event might be accepted or an interrupt might occur after the GTCR.CST bit is set to 0.

12.10.5 Priority On Conflicts

(1) GTCNT register

Table 12.22 shows a priority order of events updating GTCNT register.

Table 12.22 Priority order of sources updating GTCNT

Source updating GTCNT	Priority order
Writing by CPU (Writing to GTCNT/GTCLR)	High ↑
Clear by hardware sources set in GTCSR	
Count up or down by hardware sources set in GTUPSR/GTDNSR	
Count operation	Low

If up-counting and down-counting by hardware sources occur at the same time, the GTCNT counter value does not change. When there is a conflict between updating the GTCNT register and reading by the CPU, pre-update data is read.

(2) GTCR.CST bit

When there is a conflict between starting/stopping by hardware sources set in the GTSSR/GTPSR registers and writing by the CPU (writing to GTCR/GTSTR/GTSTP registers), writing by CPU has a priority over starting/stopping by hardware sources.
When there is a conflict between starting by hardware sources set in the GTSSR register and stopping by hardware sources set in GTPSR register, the GTCR.CST bit value does not change. Where there is a conflict between updating the GTCR.CST bit and reading by the CPU, pre-update data is read.

(3) GTCCRn registers (n = A to F)

When there is a conflict between input capture/buffer transfer operation and writing to GTCCRn registers, writing to GTCCRn registers has a priority over input capture/buffer transfer operation. When there is a conflict between input capture and writing to the counter register by the CPU or updating the counter register by hardware sources, the pre-update counter value is captured. Where there is a conflict between updating the GTCCRn registers and reading by the CPU, pre-update data is read.

(4) GTPR registers

When there is a conflict between buffer transfer operation and writing to the GTPR register, writing to GTPR register has

a priority over buffer transfer operation. When there is a conflict between updating GTPR register and reading by the CPU, pre-update data is read.

(5) GTADTRn registers (n = A, B)

When there is a conflict between buffer transfer operation and writing to the GTADTRn registers, writing to the GTADTRn registers has priority over buffer transfer operation. Where there is a conflict between updating GTADTRn registers and reading by the CPU, pre-update data is read.

(6) GTDVn registers (n = U, D)

When there is a conflict between buffer transfer operation and writing to GTDVn registers, writing to GTDVn registers has priority over buffer transfer operation. When there is a conflict between updating GTDVn registers and reading by the CPU, pre-update data is read.

13. Port Output Enable for GPT (POEG)

13.1 Overview

The output pins of the general PWM timer (GPT) can be disabled by using the port output enabling function for the GPT (POEG). Specifically, either of the following ways can be used.

- Input level detection of the GTETRGA to GTETRGD pins
- Output-disable request from the GPT
- Register settings

The GTETRGA to GTETRGD pins can also be used as GPT external trigger input pins.
Table 13.1 lists the POEG specifications, Figure 13.1 shows the block diagram, and Table 13.2 lists the input pins.

Table 13.1 POEG specifications

Parameter	Specifications
Output-disable control through input level detection	GPT output pins can be disabled when a GTETRGA to GTETRGD rising edge or high level is sampled after polarity and filter selection
Output-disable request from the GPT	• When the GTIOCA pin and the GTIOCB pin are driven to an active level simultaneously, the GPT generates an output-disable request to the POEG. Through reception of these requests, the POEG can control whether the GTIOCA and GTIOCB pins are output-disabled • GPT output pins can be set to be disabled when the GPT output pins detect a dead time error or short circuit detection between the output terminals
Output-disable control by software (registers)	GPT output pins can be disabled by modifying the register settings
Interrupt	• Allows output-disable control by input level detection • Allows output-disable requests from the GPT
External trigger output to the GPT (count start, count stop, count clear, up-count, down-count, or input capture function)	GTETRGA to GTETRGD signals can be output to the GPT after polarity and filter selection
Noise filtering	• Three times sampling for every P1φ1, P1φ8, P1φ32, or P1φ128 can be set for any of the input pins GTETRGA to GTETRGD • Positive or negative polarity can be selected for any of the input pins, GTETRGA to GTETRGD • Signal state after polarity and filter selection can be monitored

GTETRGn (n = A to D) in the subsequent descriptions indicates the GTETRGA to GTETRGD pins. The signal input from each of the pins corresponds to the POEGGn register for which n matches the n of the pin.

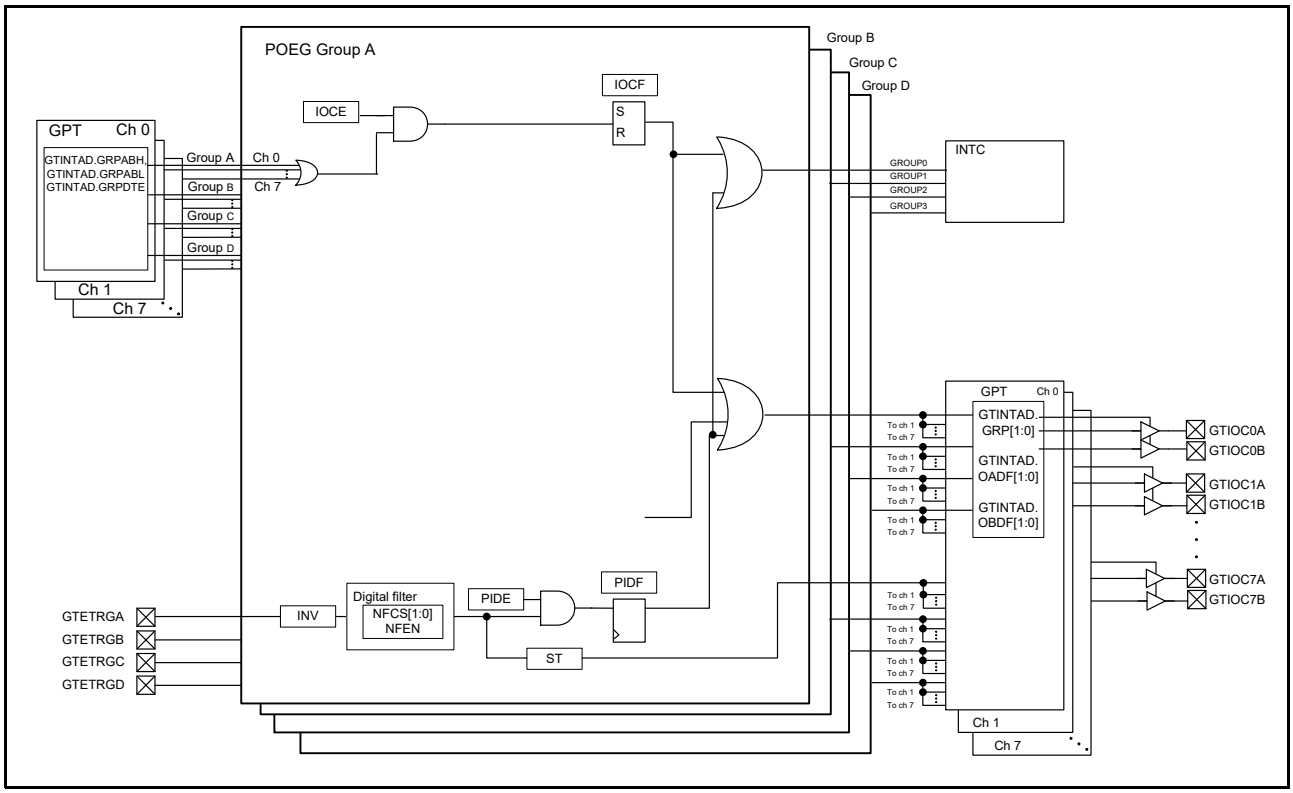


Figure 13.1 POEG block diagram

Table 13.2 POEG input pins

Pin name	I/O	Description
GTETRGA	Input	GPT output pin output-disable request signal and GPT external trigger input pin A
GTETRGB	Input	GPT output pin output-disable request signal and GPT external trigger input pin B
GTETRGC	Input	GPT output pin output-disable request signal and GPT external trigger input pin C
GTETRGD	Input	GPT output pin output-disable request signal and GPT external trigger input pin D

13.2 Register Descriptions

Table 13.3 shows the register configuration.

Table 13.3 Register configuration

Register Name	Abbreviation	Address	Access size
POEG group A setting register	POEGGA	H'E804_4000	32
POEG group B setting register	POEGGB	H'E804_4800	32
POEG group C setting register	POEGGC	H'E804_5000	32
POEG group D setting register	POEGGD	H'E804_5800	32

13.2.1 POEG Group n Setting Register (POEGGn) (n = A to D)

Address(es): POEG.POEGGA H'E804_4000, POEG.POEGGB H'E804_4800, POEG.POEGGC H'E804_5000, POEG.POEGGD H'E804_5800

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	NFCS[1:0]		NFEN	INV	—	—	—	—	—	—	—	—	—	—	—	ST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	IOCE	PIDE	SSF	—	IOCF	PIDF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	PIDF	Port Input Detection Flag	0: No output-disable request from the GTETRGn pin occurred 1: Output-disable request from the GTETRGn pin occurred.	R(/W)*1
b1	IOCF	Detection Flag for GPT Output-Disable Request	0: No output-disable request from GPT disable request occurred 1: Output-disable request from GPT disable request occurred.	R(/W)*1
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	SSF	Software Stop Flag	0: No output-disable request from software occurred 1: Output-disable request from software occurred.	R/W
b4	PIDE	Port Input Detection Enable	0: Disable output-disable requests from the GTETRGn pins 1: Enable output-disable requests from the GTETRGn pins.	R/W*2
b5	IOCE	Enable for GPT Output-Disable Request	0: Disable output-disable requests from GPT disable request 1: Enable output-disable requests from GPT disable request.	R/W*2
b15 to b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b16	ST	GTETRGn Input Status Flag	0: GTETRGn input after filtering was 0 1: GTETRGn input after filtering was 1.	R
b27 to b17	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b28	INV	GTETRGn Input Reverse	0: Input GTETRGn as-is 1: Input GTETRGn in reverse.	R/W
b29	NFEN	Noise Filter Enable	0: Disable noise filtering 1: Enable noise filtering.	R/W
b31, b30	NFCS[1:0]	Noise Filter Clock Select	b1 b0 0 0: Sample GTETRGn pin input level three times every P1φ 0 1: Sample GTETRGn pin input level three times every P1φ8 1 0: Sample GTETRGn pin input level three times every P1φ32 1 1: Sample GTETRGn pin input level three times every P1φ128.	R/W

Note 1. Only 0 can be written, to clear the flag.
Note 2. Can be modified only once after a reset.

The POEGGA to POEGGD registers control the output-disable state of the GPT pins, interrupts, and the external trigger input to the GPT. In the descriptions, POEGGn represents all of the POEGGA to POEGGD registers.

13.3 Output-Disable Control Operation

The output of the GTIOCxA and GTIOCxB pins can be disabled when any of the following conditions are satisfied.

- Input level or edge detection of the GTETRGn pins
  When POEGGn.PIDE is 1, the POEGGn.PIDF flag is set to 1.
- Output-disable request from the GPT
  When POEGGn.IOCE is 1, the POEGGn.IOCF flag is set to 1.
  The output-disable requests enabled by GRPDTE, GRPABH, and GRPABL bits of the GTINTAD register in the GPT are applied to the group selected by GRP[1:0] bits of the GTINTAD register.
- SSF bit setting
  When POEGGn.SSF is set to 1.

The state of the GTIOCxA and the GTIOCxB pins when the output is disabled is controlled by the GPT module. For details, see the descriptions of the GTINTAD.GRP[1:0], GTIOR.OADF[1:0], and GTIOR.OBDF[1:0] bits of the general PWM timer (GPT).

13.3.1 Pin Input Level Detection Operation

If the input conditions set in POEGGn.PIDE, POEGGn.NFCS[1:0], POEGGn.NFEN, and POEGGn.INV occur on the GTETRGn pins, the GPT output pins are output-disabled.

13.3.1.1 Digital Filter

Figure 13.2 shows high-level detection by the digital filter. When a high level associated with the POEGGn.INV polarity setting is detected three times consecutively with the sampling clock selected in POEGGn.NFCS[1:0] and POEGGn.NFEN, the detected level is recognized as high, and the GPT output pins are output-disabled. If even one low level is detected during this interval, the detected level is not recognized as high. In addition, in an interval where the sampling clock is not being output, changes of the levels on the GTETRGn pins are ignored.

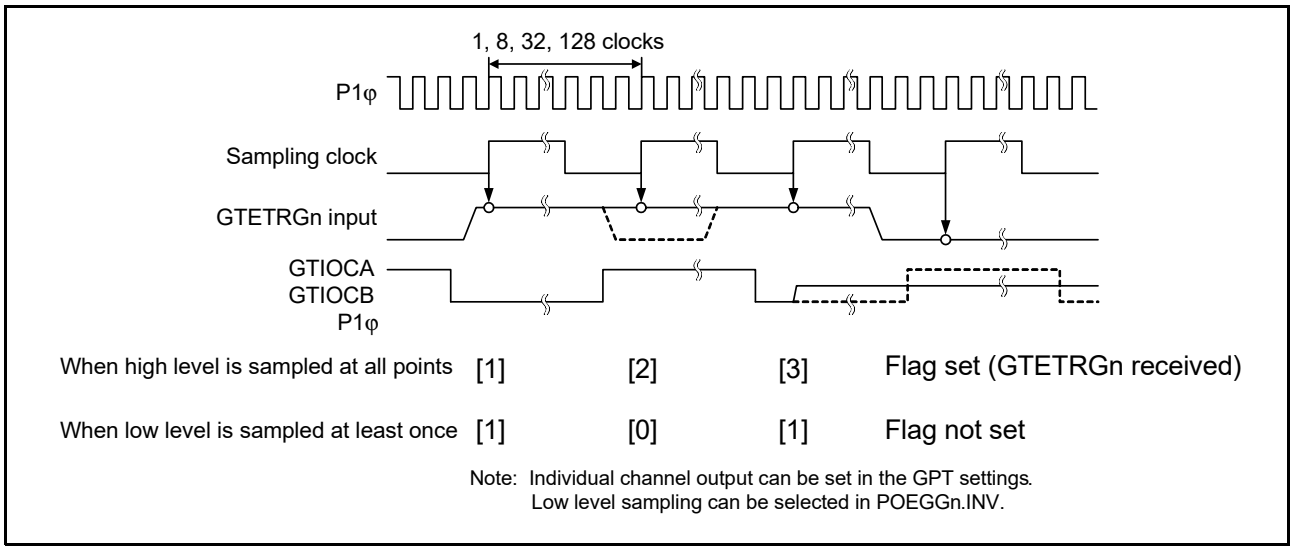


Figure 13.2 Example of digital filter operation

13.3.2 Output-Disable Requests from the GPT

For details on this operation, see the description of section 12.8.3, GTIOC Pin Output Negate Control in section 12, General PWM Timer (GPT).

13.3.3 Output-Disable Control Using Registers

The GPT output pins can be directly controlled by writing to the software stop flag, POEGGn.SSF.

13.3.4 Release from Output-Disable

To release the GPT output pins placed in the output-disable state, either return them to their initial state with a reset or clear all of the following:

- POEGGn.PIDF flag
- POEGGn.IOCF flag
- POEGGn.SSF flag.

Writing 0 to the POEGGn.PIDF flag is ignored (the flag is not cleared) if the external input pins, GTETRGn, are not disabled and the POEGGn.ST bit is not set to 0.
Writing 0 to the POEGGn.IOCF flag is valid (the flag is cleared) only if all of the GTST.DTEF, GTST.OABHF, and GTST.OABLF flags in the GPT are set to 0.
Figure 13.3 shows the release timing for output-disable. The output-disable is released at the beginning of the next count cycle of the GPT after the flag is cleared.

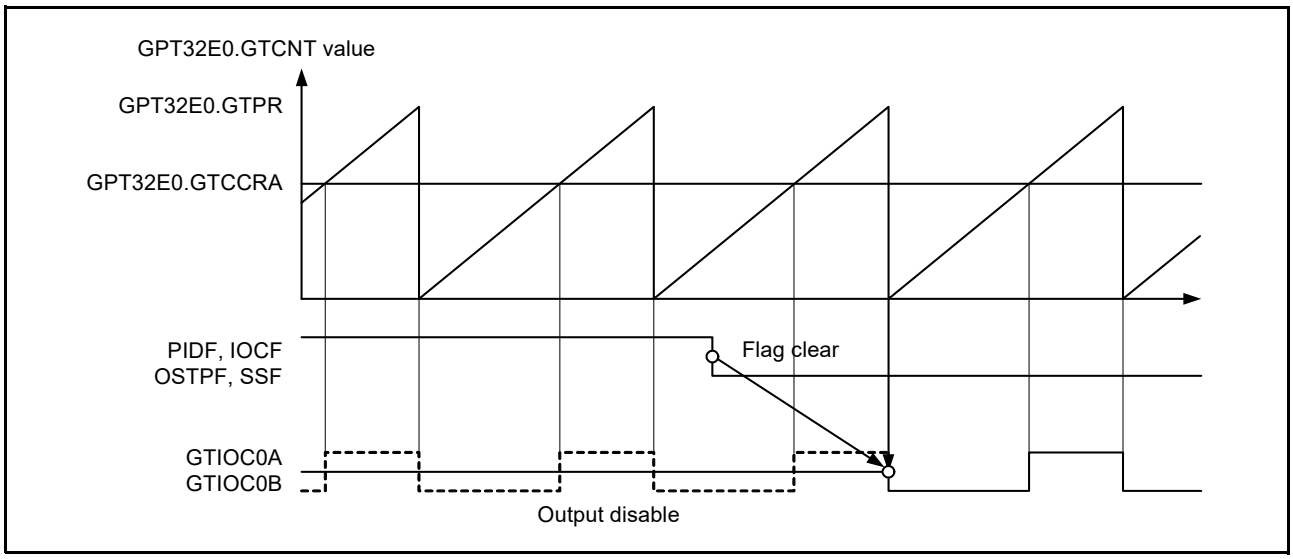


Figure 13.3 Output-disable release timing for GPT pin outputs

13.4 Interrupt Sources

The POEG generates an interrupt request when triggered by these sources:

- Output-disable control by input level detection
- Output-disable request from the GPT
- Output-disable control by the registers.

Table 13.4 lists the conditions for interrupt requests.

Table 13.4 Interrupt sources and conditions

Interrupt source	Symbol	Associated flag	Trigger conditions
POEG group A interrupt	GROUP0	POEGGA.IOCF	Output-disable request from a GPT disable request occurred
		POEGGA.PIDF	Output-disable request from the GTETRGA pin occurred
POEG group B interrupt	GROUP1	POEGGB.IOCF	Output-disable request from a GPT disable request occurred
		POEGGB.PIDF	Output-disable request from the GTETRGB pin occurred
POEG group C interrupt	GROUP2	POEGGC.IOCF	Output-disable request from a GPT disable request occurred
		POEGGC.PIDF	Output-disable request from the GTETRGC pin occurred
POEG group D interrupt	GROUP3	POEGGD.IOCF	Output-disable request from a GPT disable request occurred
		POEGGD.PIDF	Output-disable request from the GTETRGD pin occurred

13.5 External Trigger Output to the GPT

The POEG outputs the GTETRGn signals to the GPT as the GPT operation trigger signal for the following:

- Count start
- Count stop
- Count clear
- Up-count
- Down-count
- Input capture.

For the POEGGn.INV polarity setting signal, when the same level is input three times continuously with the sampling clock selected in POEGGn.NFCS[1:0] and POEGGn.NFEN, that value is output. Set the control registers the same as for the input level detection operation described in section 13.3.1, Pin Input Level Detection Operation. The state after filtering can be monitored in POEGGn.ST.

Figure 13.4 shows the output timing of an external trigger to the GPT.

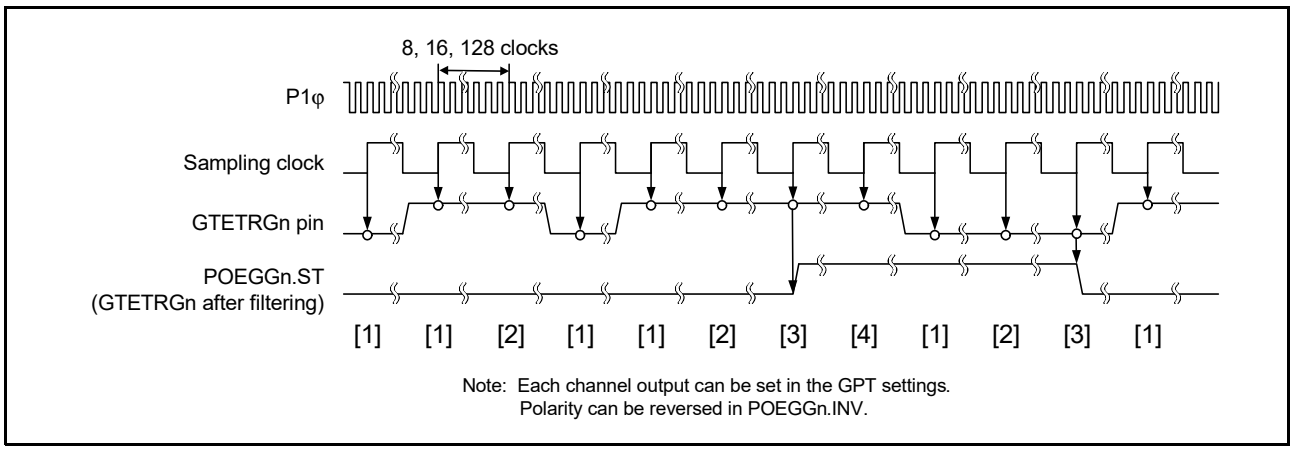


Figure 13.4 Output timing of external trigger to GPT

13.6 Usage Notes

13.6.1 Transition to Software Standby Mode

Do not invoke the Software Standby mode when using the POEG. In this mode, the POEG stops and therefore output-disable of the pins cannot be controlled.

13.6.2 Specifying Pins Associated with the GPT

The POEG controls output-disable only when a pin is associated with the GPT in the PMR register and PmnPFS.PSEL settings. When the pin is specified as a general I/O pin, the POEG does not perform output-disable control.

14. OS Timer

14.1 Functional Overview

The OS timer has the following features.

- Two operating modes
  - Interval timer mode
  - Free-running comparison mode
- Choice between startup of DMA by compare match and generation of interrupt

14.1.1 Features of OSTM

Channels

This product has the following number of channels of the OS timer.

Table 14.1 Channels of OS Timer

OS Timer	
Number of Channels	3
Name	OSTMn

Note: n = 0, 1, 2

Meaning of n

Throughout this section, the individual channels of the OS timer are identified by the index "n" (n = 0, 1, 2), for example OSTMnTO for the OS timer n output register.

Register address

The register addresses of the OS timer are given as offsets from the individual base addresses <OSTMn_base>.
The register base addresses of each OSTMn are listed in the following table.

Table 14.2 Register Base Addresses

Base Address Name	Base Address
<OSTM0_base>	$E803\_B000_H$
<OSTM1_base>	$E803\_C000_H$
<OSTM2_base>	$E803\_D000_H$

Interrupts

The OS timers can generate the following interrupt requests.

Table 14.3 OSTMn Interrupt Requests

OSTMn Signal	Function	Startup of Direct Memory Access Controller
OSTM0TINT	OSTM0 interrupt	√
OSTM1TINT	OSTM1 interrupt	√
OSTM2TINT	OSTM2 interrupt	√

14.2 Registers

The OS timer is controlled by the following registers.

14.2.1 Registers Overview

The list of OSTMn (n = 0, 1, 2) registers and the memory addresses are as follows.
For the base addresses, see Table 14.2.
For the actual addresses, the offset values indicated in the following table are added to the base addresses.

Register Name	Function	R/W	Reset Value	Access Unit (bit) 8	16	32	Address
OSTMnCMP	OSTM compare register	R/W	$0000\ 0000_H$	—	—	√	<OSTMn_base> + 00_H
OSTMnCNT	OSTM counter register	R	$FFFF\ FFFF_H$	—	—	√	<OSTMn_base> + 04_H
OSTMnTE	OSTM count enable status register	R	00_H	√	—	—	<OSTMn_base> + 10_H
OSTMnTS	OSTM count start trigger register	W	00_H	√	—	—	<OSTMn_base> + 14_H
OSTMnTT	OSTM count stop trigger register	W	00_H	√	—	—	<OSTMn_base> + 18_H
OSTMnCTL	OSTM control register	R/W	00_H	√	—	—	<OSTMn_base> + 20_H

14.2.2 Details of OSTM Registers

14.2.2.1 OSTMnCMP — OSTM Compare Register

Depending on the mode of operation, this register holds the start value for the down-counter or the value for comparison with that of the counter.

Access: This register is readable/writable in 32-bit units.
Address: <OSTMn_base>
Initial value: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OSTMnCMP[31:16]															
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTMnCMP[15:0]															
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.4 Contents of the OSTMnCMP Register

Bit Position	Bit Name	Function
31 to 0	OSTMnCMP[31:0]	• In interval timer mode: start value of the down-counter • In free-running comparison mode: value for comparison

14.2.2.2 OSTMnCNT — OSTM Counter Register

This register indicates the counter value of the timer.

Access: This register is readable in 32-bit units.
Address: OSTMn_base> + 4_H
Initial value: The initial value depends on the operating mode of the OS timer. Refer to Table 14.6.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OSTMnCNT[31:16]															
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTMnCNT[15:0]															
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.5 Contents of the OSTMnCNT Register

Bit Position	Bit Name	Function
31 to 0	OSTMnCNT[31:0]	32-bit counter value

The following table shows the correspondence between operating mode, counting direction and initial value. The initial value is the value read from the counter after a change to the operating mode.

Table 14.6 Correspondence between Operating Mode, Counting Direction and Initial Value

Timer Operating Mode	OSTMnCTL.OSTMnMD1	Counting Direction	Initial value
Interval timer mode	0 *1	Down	FFFF FFFF$_H$
Free-running comparison mode	1	Up	0000 0000$_H$

Note 1. Value after reset

14.2.2.3 OSTMnTE — OSTM Count Enable Status Register

This register indicates whether the counter is enabled or disabled.

Access: This register is readable in 8-bit units.
Address: <OSTMn_base>+ 10_H
Initial value: 00_H

Bit:	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	OSTMnTE
R/W:	R	R	R	R	R	R	R	R

Table 14.7 Contents of the OSTMnTE Register

Bit Position	Bit Name	Function
7 to 1	Reserved	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTMnTE	This bit indicates whether the counter is enabled or disabled. 0: Counter disabled 1: Counter enabled This bit is set to 1 in response to OSTMnTS.OSTMnTS being set to 1. This bit is reset to 0 in response to OSTMnTT.OSTMnTT being set to 1.

Note: When OSTMnTE = 0, the counter retains its value.
If the counter is restarted, it
- restarts counting down from the value in the OSTMnCMP register if it is in interval timer mode or
- restarts counting up from the counter value 0000 0000_H if it is in free running comparison mode.

14.2.2.4 OSTMnTS — OSTM Count Start Trigger Register

This register starts the counter.

Access: This register is writable in 8-bit units. It is always read as 00_H.
Address: <OSTMn_base>+ 14_H
Initial value: 00_H

Bit:	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	OSTMnTS
R/W:	R	R	R	R	R	R	R	W

Table 14.8 Contents of the OSTMnTS Register

Bit Position	Bit Name	Function
7 to 1	Reserved	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTMnTS	This bit starts the counter. 0: This setting has no effect. 1: Starts the counter and sets OSTMnTE.OSTMnTE = 1. • In interval timer mode, a forced restart is executed if this bit is set while OSTMnTE.OSTMnTE = 1. • In free-running comparison mode, setting this bit is ignored as long as OSTMnTE.OSTMnTE = 1.

14.2.2.5 OSTMnTT — OSTM Count Stop Trigger Register

This register stops the counter.

Access: This register is writable in 8-bit units. It is always read as 00_H.

Address: <OSTMn_base>+ 18_H

Initial value: 00_H

Bit:	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	OSTMnTT
R/W:	R	R	R	R	R	R	R	W

Table 14.9 Contents of the OSTMnTT Register

Bit Position	Bit Name	Function
7 to 1	Reserved	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTMnTT	Stops the counter. 0: This setting has no effect. 1: Stops the counter and clears the OSTMnTE.OSTMnTE bit.

14.2.2.6 OSTMnCTL — OSTM Control Register

This register specifies the operating mode for the counter and controls enabling/disabling of OSTMnTINT interrupt requests when counting starts.

Access: This register is readable/writable in 8-bit units. Writing to this register is only possible if the counter is disabled (OSTMnTOE.OSTMnTOE = 0).

Address: <OSTMn_base>+ 20_H

Initial value: 00_H

Bit:	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	OSTMnMD1	OSTMnMD0
R/W:	R	R	R	R	R	R	R/W	R/W

Table 14.10 Contents of the OSTMnCTL Register

Bit Position	Bit Name	Function
7 to 2	Reserved	Reserved These bits are always read as 0. The write value should always be 0.
1	OSTMnMD1	Specifies the operating mode for the counter. 0: Interval timer mode 1: Free-running comparison mode
0	OSTMnMD0	Controls enabling/disabling of OSTMnTINT interrupt requests when counting starts. 0: Disables the interrupts when counting starts. 1: Enables the interrupts when counting starts.

14.3 Functional Description

Each OS timer is a 32-bit timer/counter.
The settings for operating mode specify the direction of counting (up or down) and the generation of interrupt requests.

14.3.1 Block Diagram

The following block diagram shows the main components of OSTM.

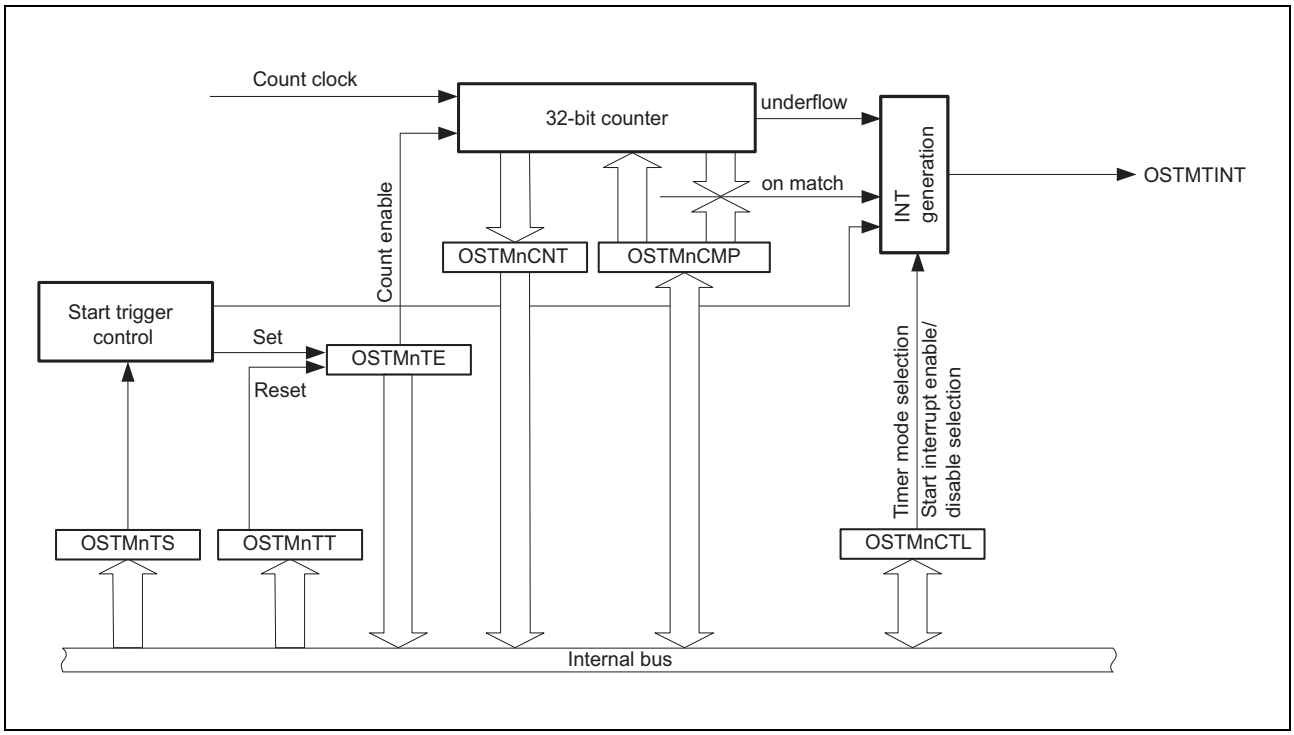


Figure 14.1 Block Diagram of OSTM

14.3.2 Count Clock

The count clock of OSTMn is P1φ.

14.3.3 Generation of Interrupt Request

An OSTMnTINT interrupt request is generated whenever the counter reaches 0000 0000_H (in interval timer mode) or matches the comparison value (in free-running comparison mode).
An interrupt request can also be generated on starting and restarting of the counter. This is controlled by the OSTMnCTL.OSTMnMD0 bit.
This operation is shown in the following figure.

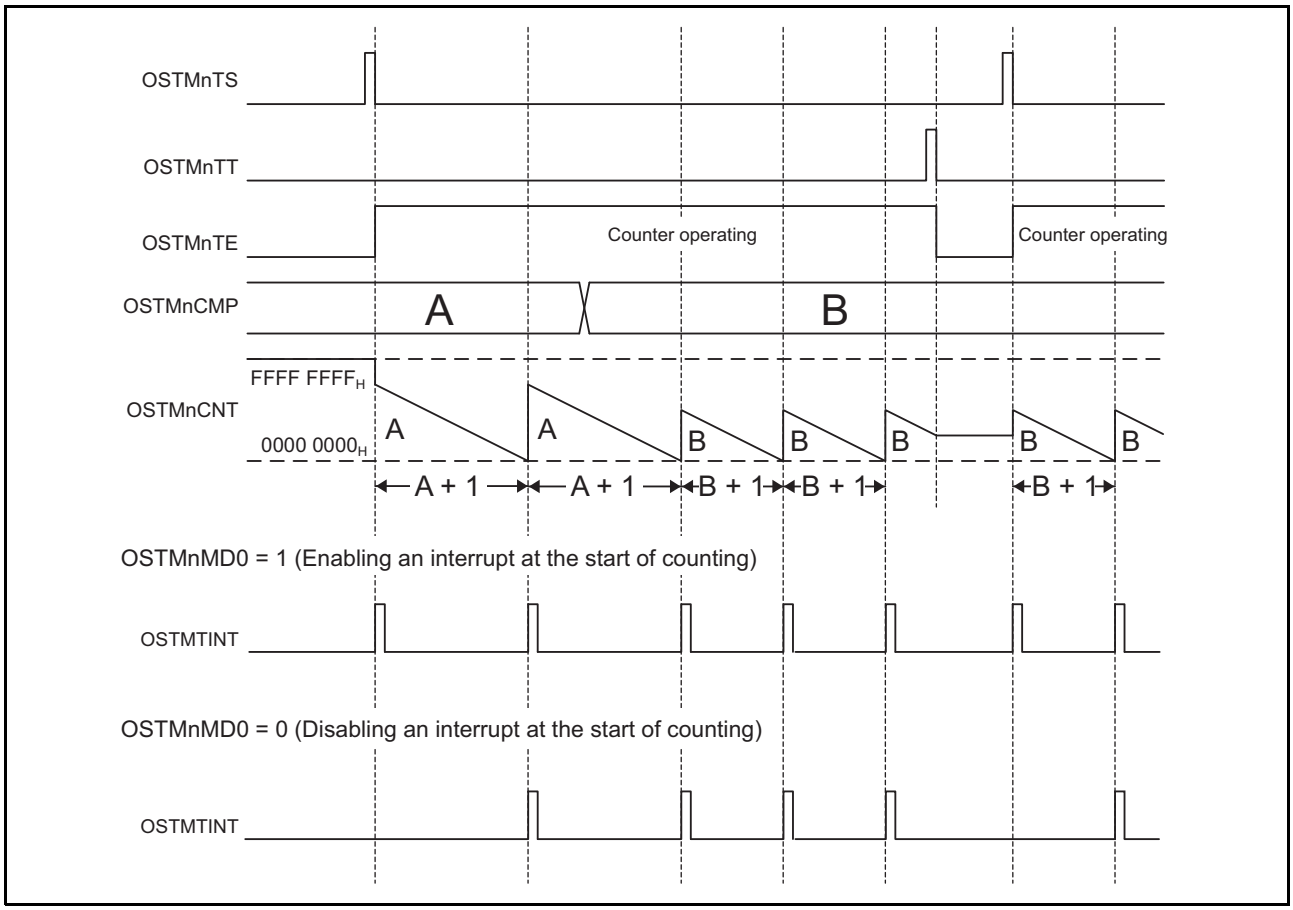


Figure 14.2 Generating an Interrupt when Counting Starts (in Interval Timer Mode)

14.3.4 Starting and Stopping the Timer

The OS timer is started and stopped as follows.

Starting the timer

The timer is started in either of the following way:

- setting the OSTMnTS.OSTMnTSF bit to 1

Status bit OSTMnTE.OSTMnTE is set to 1.
The counter starts to count up or down in accord with the settings for operating mode.

Stopping the timer

Setting the OSTMnTT.OSTMnTT bit to 1 stops the timer.
This also clears the OSTMnTE.OSTMnTE status flag.

14.3.5 Interval Timer Mode

Select the interval timer mode when an OS timer is to be used as a reference timer for generating interrupt requests at a fixed interval.

14.3.5.1 Basic Operation in Interval Timer Mode

In interval timer mode, the timer counts down from the value specified in the OSTMnCMP register. An OSTMnTINT interrupt request is generated when the counter reaches 0000 0000_H.
Select interval timer mode by setting OSTMnCTL.OSTMnMD1 = 0.
New values can be written to the OSTMnCMP register at any time. If it is rewritten during count operation, the counter loads the new OSTMnCMP value when the next 0000 0000_H is reached.

Cycles of OSTMnTINT output

The cycle of OSTMnTINT output is as follows.

- OSTMnTINT generation cycle = counter-clock cycle × (OSTMnCMP + 1)

The following figure shows the basic operation of OSTM when counter-start interrupts is enabled in interval timer mode.

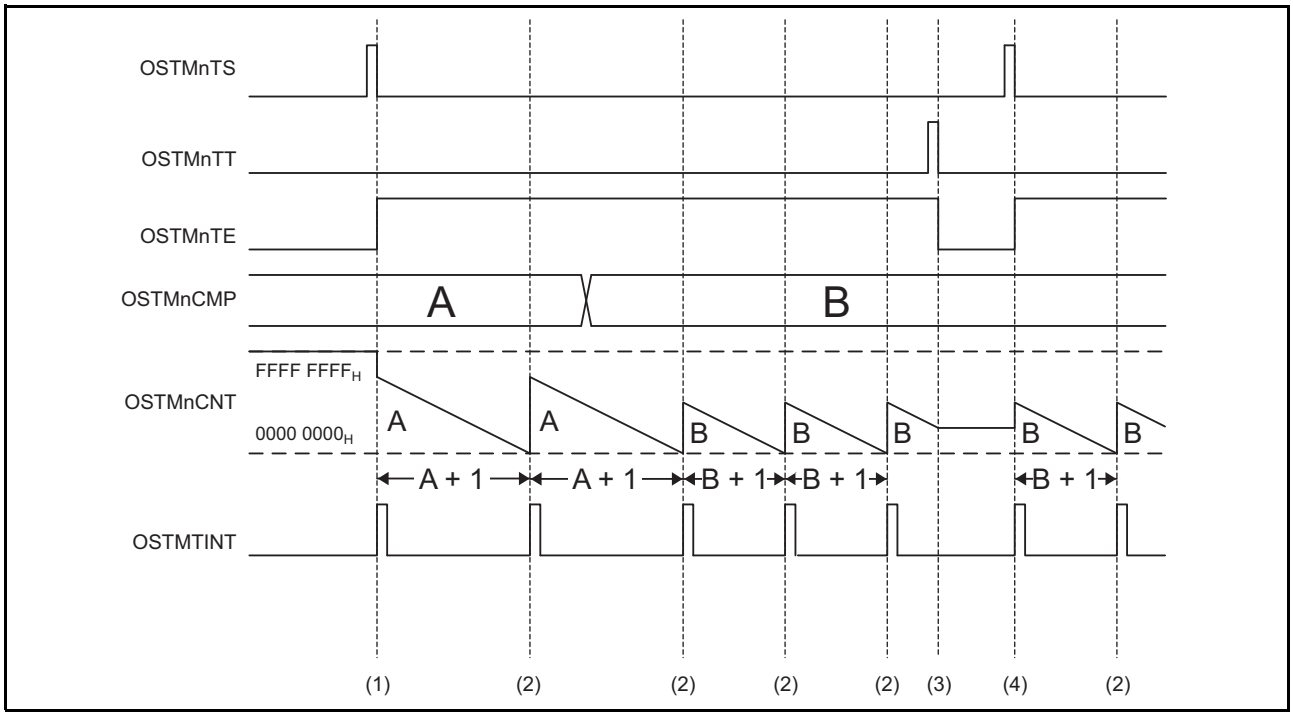


Figure 14.3 Timing Diagram of OSTM in Interval Timer Mode

The timing diagram above shows the following:

1. The counter starts counting when OSTMnTS.OSTMnTS = 1. The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter.
   The counter starts counting down from the value of OSTMnCMP.
   If OSTMnCTL.OSTMnMD0 is 1, OSTMTINT interrupt requests are generated at the start of counting. The OSTMnCNT register contains the current value as the counter.
2. When the counter reaches 0000 0000_H, an OSTMTINT interrupt request is generated. The counter loads the new start value from OSTMnCMP and continues counting down.
3. When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter. The counter retains its current value until it is restarted.
4. When counting is restarted (OSTMnTS.OSTMnTS = 1), the counter loads the new start value from OSTMnCMP and starts counting down.

Forced restart

The counter is forcibly restarted by setting OSTMnTS.OSTMnTS = 1 during counting.
The counter loads the start value from the OSTMnCMP register and continues to count down.
The following figure shows the forced restart of the OS Timer in interval timer mode, with counter-start interrupts enabled (OSTMnCTL.OSTMnMD0 = 1).

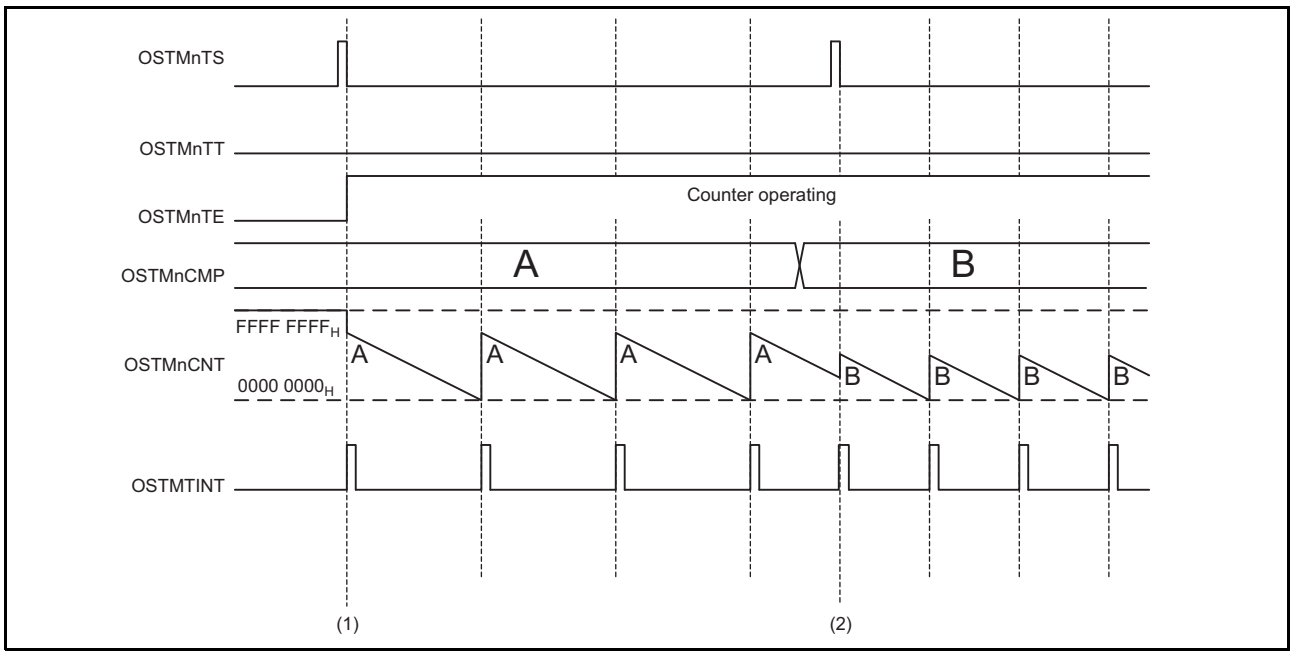


Figure 14.4 Timing Diagram of Forced Restart in Interval Timer Mode

Operations shown in the above timing diagram are as follows.

1. The counter is started and stopped as described under Figure 14.3, Timing Diagram of OSTM in Interval Timer Mode.
2. Setting OSTMnTS.OSTMnTS = 1 restarts the counter while counting is in progress (i.e. while OSTMnTE.OSTMnTE = 1).
   The counter immediately restarts counting down, starting with the current value of OSTMnCMP.
   When OSTMnCTL.OSTMnMD0 = 1, an OSTMTINT interrupt request is generated when counting starts.

14.3.5.2 Operation when OSTMnCMP = 0000 0000_H

When OSTMnCMP = 0000 0000_H, OSTM behaves as follows.

- When the counter is enabled, the OSTMTINT interrupt request is always set to 1.

The following figure shows operations of OSTM when OSTMnCMP = 0000 0000_H, and counter-start interrupts are enabled.

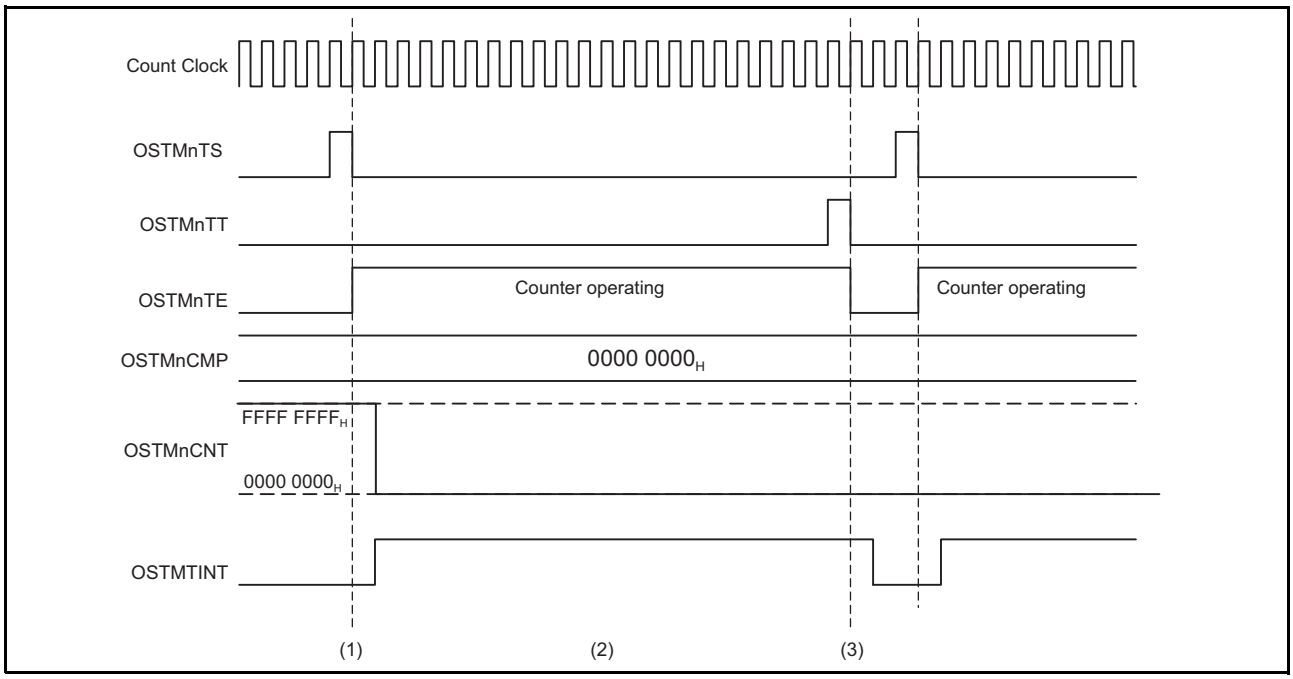


Figure 14.5 Timing Diagram when OSTMnCMP = 0000 0000H in Interval Timer Mode

The timing diagram above shows the following operations:

1. The counter is reloaded with the value in OSTMnCMP as soon as it starts counting, so the value 0000 0000_H is retained in OSTMnCMP.
2. The OSTMTINT interrupt request is continuously asserted.
3. After the counter stops, the OSTMTINT interrupt request signal is deasserted.
4. When interrupts on starting of the counter are disabled, no interrupt is generated when counting starts.

14.3.6 Free-Running Comparison Mode

14.3.6.1 Basic Operation in Free-Running Comparison Mode

In free-running comparison mode, the counter counts up from 0000 0000_H to FFFF $FFFF_H$. An OSTMnTINT interrupt request is output when the current value of the counter matches the value of the OSTMnCMP register. The free-running comparison mode is selected by setting the OSTMnCTL.OSTMnMD1 bit to 1.
New values can be written to the OSTMnCMP register at any time.
The following figure shows the basic operation of OSTM in free-run compare mode with the start of counting enabled (OSTMnCTL.OSTMnMD0 = 1).

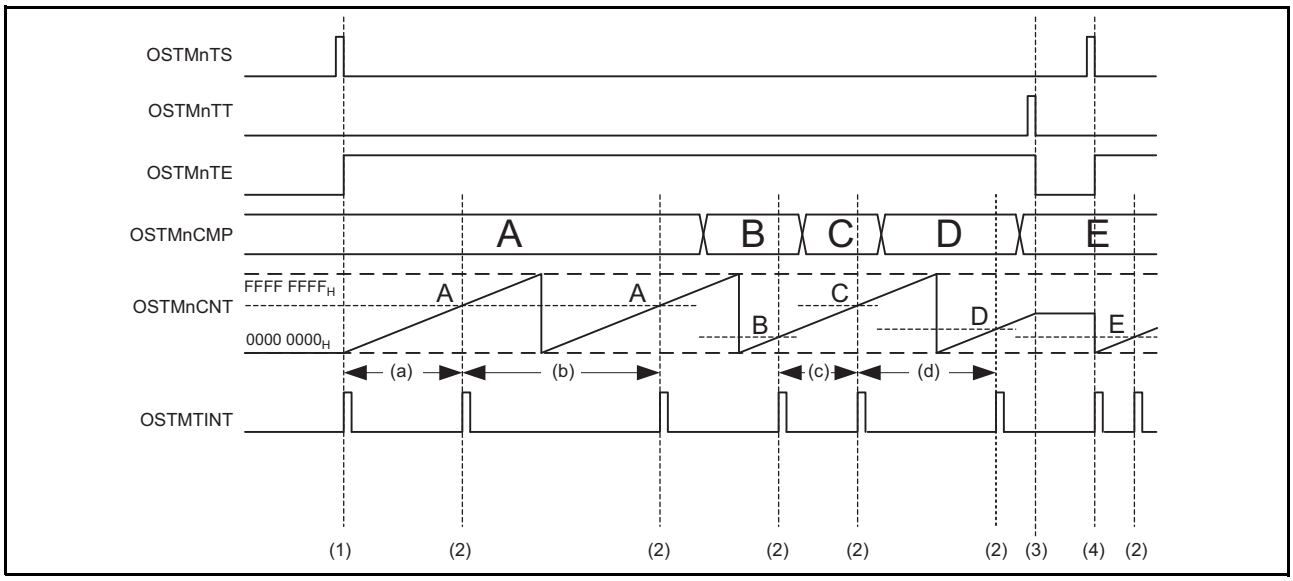


Figure 14.6 Timing Diagram of OSTM in Free-Run Compare Mode

The timing diagram above shows the following:

1. The counter starts counting when OSTMnTS.OSTMnTS = 1.
   The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter. The counter counts up from 0000 0000_H to FFFF $FFFF_H$. The OSTMnCNT register is the counter, so it contains the current value.
   When OSTMnCTL.OSTMnMD0 = 1, an OSTMTINT interrupt request is generated at the start of counting.
2. When the current counter value matches the value in the OSTMnCMP register, an OSTMTINT interrupt request is generated.
3. When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter.
   The counter retains its current value until it is restarted.
4. Counting by the counter restarts from 0000 0000_H when OSTMnTS.OSTMnTS = 1.

OSTMTINT period

The OSTMTINT generation period is different at the start of counting and depends on the old and new compare values if OSTMnCMP is rewritten during operation.

Table 14.11 OSTMTINT Generation Timing

Old Value for Comparison	New Value for Comparison	Counter Value at Time of Rewriting	Period of OSTMTINT Generation	Label in Timing Diagram
Counter starts			(A + 1) × counter clock period	(a)
A	A	No rewriting	(FFFF FFFF$_H$ + 1) × counter clock period	(b)
B	C > B	B < counter value < C	(C – B) × counter clock period	(c)
C	D < C	Counter value > D,C	(FFFF FFFF$_H$ – C + D + 1) × counter clock period	(d)

Forced restart

Forced restarting does not proceed during counting even if the OSTMnTS.OSTMnTS bit is set.
The counter ignores the attempted setting and continues counting.

14.3.6.2 Operation when OSTMnCMP = 0000 0000$_H$

The following figure shows the operation of OSTM when OSTMnCMP = 0000 0000$_H$, and counter-start interrupts are enabled (OSTMnCTL.OSTMnMD0 = 1).

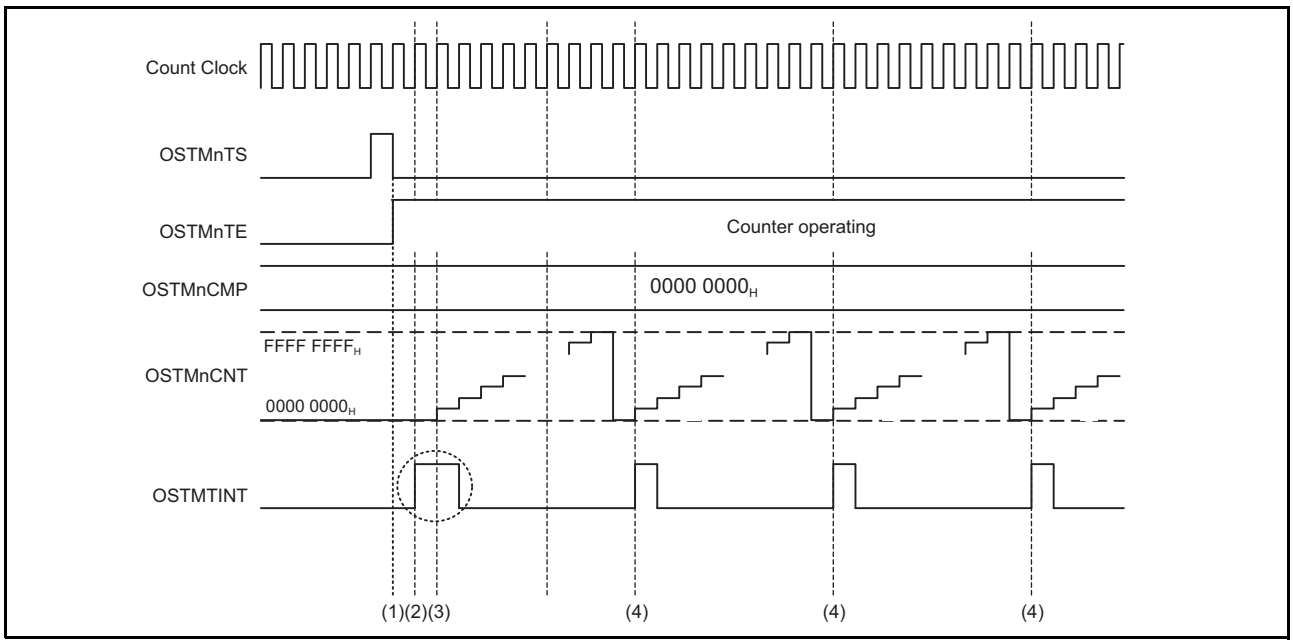


Figure 14.7 Timing Diagram when OSTMnCMP = 0000 0000$_H$ in Free-Run Compare Mode

The timing diagram above shows the following operations.

1. Once the counter starts, it counts up from 0000 0000$_H$ to FFFF FFFF$_H$.
2. An OSTMTINT interrupt request is generated when counting starts.
3. If the current counter value matches OSTMnCMP, an OSTMTINT interrupt request is generated.
   If OSTMnCMP = 0000 0000$_H$ in the above case, OSTMTINT is generated over two clock cycles.
4. Every (FFFF FFFF$_H$ + 1) clock cycles the OSTMTINT interrupt request is asserted.

When interrupts on starting of the counter are disabled, no interrupt is generated when counting starts.

15. Watchdog Timer

This LSI includes the watchdog timer, which externally outputs an overflow signal (WDTOVF#) on overflow of the counter when the value of the counter has not been updated because of a system malfunction. This module can simultaneously generate an internal reset signal for the entire LSI.
In a similar manner, this module can also generate an internal reset signal in response to a CPU parity error and output the parity error signal (PERROUT#) externally.
This module is a single channel timer that counts up the clock oscillation settling period when the system leaves software standby mode. It can also be used as a general watchdog timer or interval timer.

15.1 Features

- Can be used to ensure the clock oscillation settling time
  This module is used in leaving software standby mode.
- Can switch between watchdog timer mode and interval timer mode.
- Outputs WDTOVF# signal in watchdog timer mode
  When the counter overflows in watchdog timer mode, the WDTOVF# signal is output externally. It is possible to select whether to reset the LSI internally when this happens. The internal reset signal is used as the power-on reset signal.
- Interrupt generation in interval timer mode
  An interval timer interrupt (ITI) is generated when the counter overflows.
- Choice of 16 counter input clocks
  Sixteen clocks (P0φ × 1 to P0φ × 1/4194304) that are obtained by dividing the peripheral clock can be selected.
- Generation of an internal reset signal or interrupt signal in response to a CPU parity error.
  This module generates an internal reset signal for the entire LSI and outputs the PERROUT# signal externally when generation of the internal reset signal is selected as a behavior in response to a CPU parity error.
  This internal reset signal is equivalent to the power-on reset. Similarly, this module generates an interrupt signal(CA9PEI) when the interrupt signal is selected.
- Retention of the CPU parity error sources
  The parity error sources are retained in this module and are not cleared by the internal reset.

Figure 15.1 shows a block diagram.

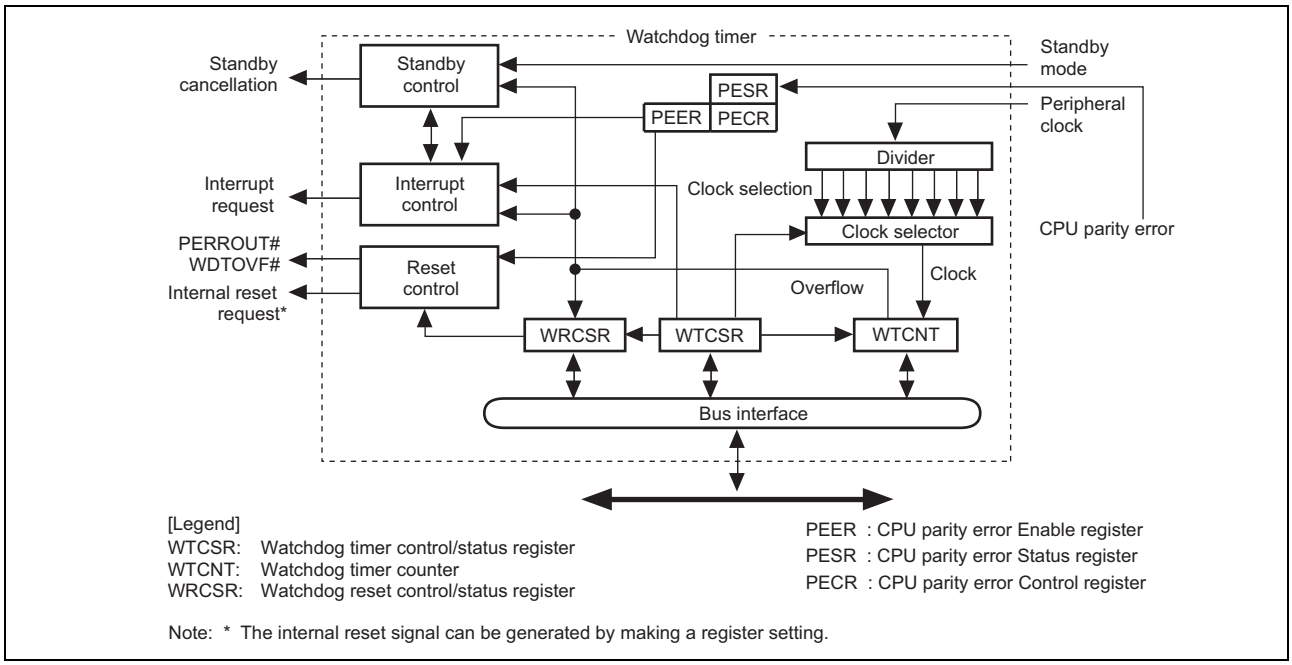


Figure 15.1 Block Diagram

15.2 Input/Output Pin

Table 15.1 shows the pin configuration.

Table 15.1 Pin Configuration

Pin Name	Symbol	I/O	Function
Watchdog timer overflow	WDTOVF#	Output	Outputs the counter overflow signal in watchdog timer mode
CPU parity error	PERROUT#	Output	This signal goes to the active (low) level when a CPU parity error is detected.

15.3 Register Descriptions

Table 15.2 shows the register configuration.

Table 15.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Watchdog timer counter	WTCNT	R/W	H'00	H'FCFE7002	16*
Watchdog timer control/status register	WTCSR	R/W	H'10	H'FCFE7000	16*
Watchdog reset control/status register	WRCSR	R/W	H'1F	H'FCFE7004	16*
CPU parity error enable register	PEER	R/W	H'00	H'FCFE7020	16*
CPU parity error control register	PECR	R/W	H'00	H'FCFE7022	16*
CPU parity error status register	PESR	R/W	H'00	H'FCFE7024	16*

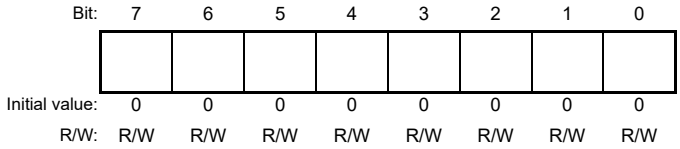
Note: * For the access size, see section 15.3.7, Notes on Register Access. Use 8-bit access to read.

15.3.1 Watchdog Timer Counter (WTCNT)

WTCNT is an 8-bit readable/writable register that is incremented by cycles of the selected clock signal. When an overflow occurs, it generates a watchdog timer overflow signal (WDTOVF#) in watchdog timer mode and an interrupt in interval timer mode.

Use 16-bit access to write to WTCNT, writing H'5A in the upper byte. Use 8-bit access to read from WTCNT.

Note: The method for writing to WTCNT differs from that for other registers to prevent erroneous writes. See section 15.3.7, Notes on Register Access for details.



15.3.2 Watchdog Timer Control/Status Register (WTCSR)

WTCSR is an 8-bit readable/writable register composed of bits to select the clock used for the count, overflow flags, and timer enable bit.

When used to count the clock oscillation settling time for canceling software standby mode, it retains its value after counter overflow.

Use 16-bit access to write to WTCSR, writing H'A5 in the upper byte. Use 8-bit access to read from WTCSR.

Note: The method for writing to WTCSR differs from that for other registers to prevent erroneous writes. See section 15.3.7, Notes on Register Access for details.

Bit:	7	6	5	4	3	2	1	0
	IOVF	WT/IT#	TME	—	CKS[3:0]			
Initial value:	0	0	0	1	0	0	0	0
R/W:	R/(W)	R/W	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	IOVF	0	R/(W)	Interval Timer Overflow Indicates that WTCNT has overflowed in interval timer mode. This flag is not set in watchdog timer mode. 0: No overflow 1: WTCNT overflow in interval timer mode [Clearing condition] • When 0 is written to IOVF after reading IOVF
6	WT/IT#	0	R/W	Timer Mode Select Selects whether to use this module as a watchdog timer or an interval timer. 0: Use as interval timer 1: Use as watchdog timer Note: When the WTCNT overflows in watchdog timer mode, the WDTOVF# signal is output externally. If this bit is modified when this module is running, incrementation may not be performed correctly.
5	TME	0	R/W	Timer Enable Starts and stops timer operation. Clear this bit to 0 when using this module in software standby mode or when changing the clock frequency. 0: Timer disabled Incrementation stops and WTCNT value is retained 1: Timer enabled
4	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	CKS[3:0]	000	R/W	Clock Select These bits select the clock to be used for the WTCNT count from the sixteen types obtainable by dividing the peripheral clock (P0φ). The overflow period that is shown inside the parenthesis in the table is the value when the peripheral clock (P0φ) is 33.00 MHz.

	Clock Ratio	Overflow Cycle
0000:	1 × P0 × φ	(7.75 µs)
0001:	1/64 × P0φ	(496.48 µs)
0010:	1/128 × P0φ	(992.96 µs)
0011:	1/256 × P0φ	(1.98 ms)
0100:	1/512 × P0φ	(3.97 ms)
0101:	1/1024 × P0φ	(7.94 ms)
0110:	1/4096 × P0φ	(31.77 ms)
0111:	1/16384 × P0φ	(127.10 ms)
1000:	1/32768 × P0φ	(254.20 ms)
1001:	1/65536 × P0φ	(508.40 ms)
1010:	1/131072 × P0φ	(1.01 s)
1011:	1/262144 × P0φ	(2.03 s)
1100:	1/524288 × P0φ	(4.06 s)
1101:	1/1048576 × P0φ	(8.13 s)
1110:	1/2097152 × P0φ	(16.26 s)
1111:	1/4194304 × P0φ	(32.53 s)

Note: If bits CKS[3:0] are modified when this module is running, the incrementation may not be performed correctly. Ensure that these bits are modified only when this module is not running.

15.3.3 Watchdog Reset Control/Status Register (WRCSR)

WRCSR is an 8-bit readable/writable register that controls output of the internal reset signal generated by watchdog timer counter (WTCNT) overflow.

Note: The method for writing to WRCSR differs from that for other registers to prevent erroneous writes. See section 15.3.7, Notes on Register Access for details.

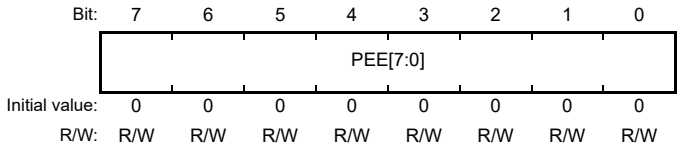
Bit:	7	6	5	4	3	2	1	0
	WOVF	RSTE	—	—	—	—	—	—
Initial value:	0	0	0	1	1	1	1	1
R/W:	R/(W)	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	WOVF	0	R/(W)	Watchdog Timer Overflow Indicates that the WTCNT has overflowed in watchdog timer mode. This bit is not set in interval timer mode. 0: No overflow 1: WTCNT has overflowed in watchdog timer mode [Clearing condition] • When 0 is written to WOVF after reading WOVF
6	RSTE	0	R/W	Reset Enable Selects whether to generate a signal to reset the LSI internally if WTCNT overflows in watchdog timer mode. In interval timer mode, this setting is ignored. 0: Not reset when WTCNT overflows* 1: Reset when WTCNT overflows Note: * LSI not reset internally, but WTCNT and WTCSR reset within this module.
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4 to 0	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.

15.3.4 CPU Parity Error Enable Register (PEER)

PEER is an 8-bit readable/writable register that controls the internal reset signal and interrupt generation due to CPU parity error.

Note: The method for writing to PEER differs from that for other registers to prevent erroneous writes. See section 15.3.7, Notes on Register Access for details.

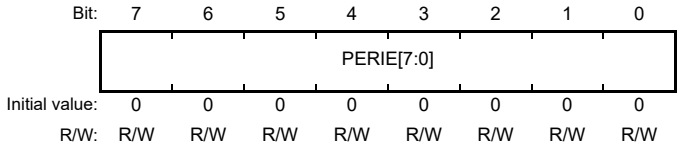


Bit	Bit Name	Initial Value	R/W	Description
7 to 0	PEE[7:0]	H'00	R/W	CPU parity error enable Each bit controls the operation of PERIE[7:0] (internal reset or interrupt generation). For the contents of the CPU parity error, refer to the CPU parity error status register (PESR). 0: Disable the internal reset or interrupt generation 1: Enable the internal reset or interrupt generation

15.3.5 CPU Parity Error Control Register (PECR)

PECR is an 8-bit readable/writable register that selects interrupt or internal reset signal generation due to CPU parity error. The selected operation is performed when the PEE bit of the CPU parity error enable register (PEER) is enabled.

Note: The method for writing to PECR differs from that for other registers to prevent erroneous writes. See section 15.3.7, Notes on Register Access for details.



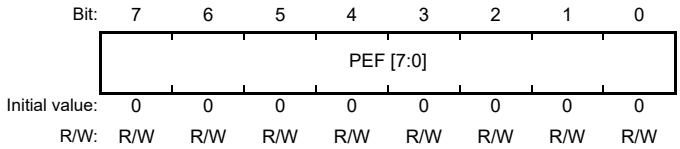
Bit	Bit Name	Initial Value	R/W	Description
7 to 0	PERIE[7:0]	H'00	R/W	CPU parity error mode select Select internal reset or interrupt generation in case of CPU parity error. It is valid only when the PEE bit of the CPU parity error enable register (PEER) is 1. 0: Select interrupt 1: Select internal reset When internal reset is selected, this module outputs the PERROUT# signal externally.

15.3.6 CPU Parity Error Status Register (PESR)

PESR is an 8-bit readable/0-writable register that holds each source due to CPU parity error. To clear each source, write 0 after reading.

It is not affected by the setting of the CPU parity error enable register (PEER).

Note: The method for writing to PESR differs from that for other registers to prevent erroneous writes. See section 15.3.7, Notes on Register Access for details.



Bit	Bit Name	Initial Value	R/W	Description
7	PEF[7]	H'00	R/(W)	CPU parity error 7 flag Indicates that BTAC parity error has occurred. When an error occurs, it operates corresponding to PEE[7], PERIE[7]. [Clear condition] • Write 0 after reading.
6	PEF[6]	H'00	R/(W)	CPU parity error 6 flag Indicates that GHB parity error has occurred. When an error occurs, it operates corresponding to PEE[6], PERIE[6]. [Clear condition] • Write 0 after reading.
5	PEF[5]	H'00	R/(W)	CPU parity error 5 flag Indicates that Instruction Cache Tag RAM parity error has occurred. When an error occurs, it operates corresponding to PEE[5], PERIE[5]. [Clear condition] • Write 0 after reading.
4	PEF[4]	H'00	R/(W)	CPU parity error 4 flag Indicates that Instruction cache data RAM parity error has occurred. When an error occurs, it operates corresponding to PEE[4], PERIE[4]. [Clear condition] • Write 0 after reading.
3	PEF[3]	H'00	R/(W)	CPU parity error 3 flag Indicates that Main TLB parity error has occurred. When an error occurs, it operates corresponding to PEE[3], PERIE[3]. [Clear condition] • Write 0 after reading.
2	PEF[2]	H'00	R/(W)	CPU parity error 2 flag Indicates that Data cache External cache attribute RAM parity error has occurred. When an error occurs, it operates corresponding to PEE[2], PERIE[2]. [Clear condition] • Write 0 after reading.
1	PEF[1]	H'00	R/(W)*	CPU parity error 1 flag Indicates that Data cache tag RAM parity error has occurred. When an error occurs, it operates corresponding to PEE[1], PERIE[1]. [Clear condition] • Write 0 after reading.
0	PEF[0]	H'00	R/(W)*	CPU parity error 0 flag Indicates that Data cache data RAM parity error has occurred. When an error occurs, it operates corresponding to PEE[0], PERIE[0]. [Clear condition] • Write 0 after reading.

15.3.7 Notes on Register Access

The watchdog timer counter (WTCNT), watchdog timer control/status register (WTCSR), and watchdog reset control/status register (WRCSR) are more difficult to write to than other registers. The procedures for reading or writing to these registers are given below.

(1) Writing to WTCNT and WTCSR

These registers must be written by a 16-bit transfer instruction. They cannot be written by an 8- or 32-bit transfer instruction.
When writing to WTCNT, set the upper byte to H'5A and transfer the lower byte as the write data, as shown in Figure 15.2. When writing to WTCSR, set the upper byte to H'A5 and transfer the lower byte as the write data. This transfer procedure writes the lower byte data to WTCNT or WTCSR.

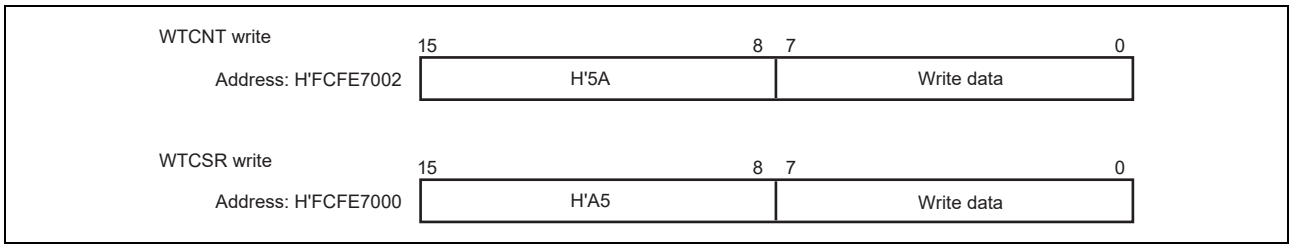


Figure 15.2 Writing to WTCNT and WTCSR

(2) Writing to WRCSR

WRCSR must be written by a 16-bit access to address H'FCFE7004. It cannot be written by 8- or 32-bit transfer instructions.

Procedures for writing 0 to WOVF (bit 7) and for writing to RSTE (bit 6) are different, as shown in Figure 15.3.

To write 0 to the WOVF bit, the write data must be H'A5 in the upper byte and H'00 in the lower byte. This clears the WOVF bit to 0. The RSTE bit is not affected. To write to the RSTE bit, the upper byte must be H'5A and the lower byte must be the write data. The value of bit 6 of the lower byte is transferred to the RSTE bit. The WOVF bit is not affected.

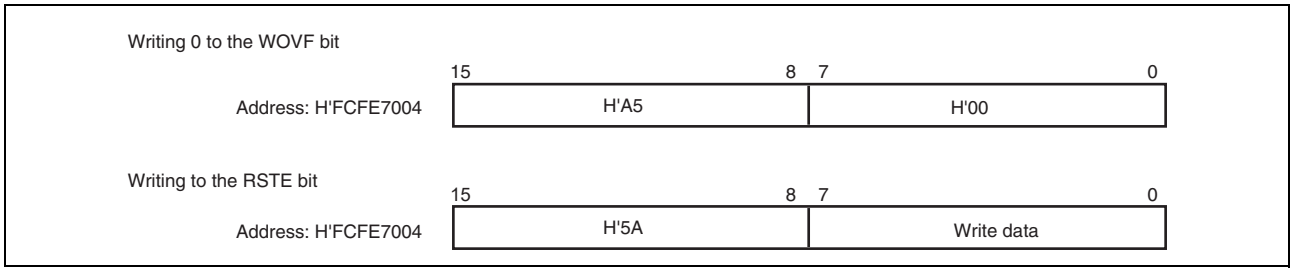


Figure 15.3 Writing to WRCSR

(3) Reading from WTCNT, WTCSR, and WRCSR

WTCNT, WTCSR, and WRCSR are read in a method similar to other registers. WTCSR is allocated to address H'FCFE7000, WTCNT to address H'FCFE7002, and WRCSR to address H'FCFE7004. Eight-bit transfer instructions must be used for reading from these registers.

(4) Writing to PEER, PECR, and PESR

These registers must be written by a 16-bit transfer instruction. They cannot be written by an 8- or 32-bit transfer instruction.
When writing to PEER, PECR, or PESR, set the upper byte to H'5A and transfer the lower byte as the write data, as shown in Figure 15.4.
This transfer procedure writes the lower byte data to PEER, PECR, or PESR.

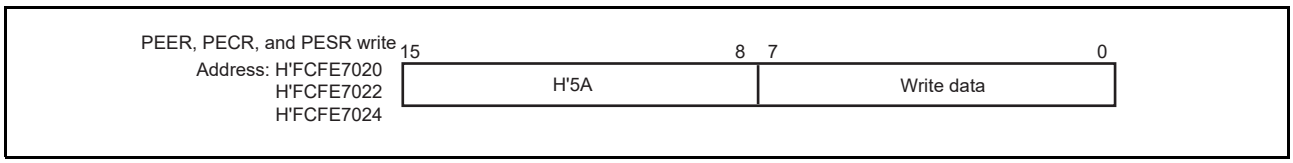


Figure 15.4 Writing to PEER, PECR, and PESR

(5) Reading from PEER, PECR, and PESR

Reading can be done in the same way as general registers. Eight-bit transfer instructions must be used for reading from these registers.

15.4 Usage

15.4.1 Canceling Software Standby Mode

This module can be used to cancel software standby mode with an interrupt such as an NMI interrupt. The procedure is described below. (This module does not operate when resets are used for canceling, so keep the RES# pin low until clock oscillation settles.)

1. Before making a transition to software standby mode, always clear the TME bit in WTCSR to 0. When the TME bit is 1, an erroneous reset or interval timer interrupt may be generated when the count overflows.
2. Set the type of count clock used in the CKS[3:0] bits in WTCSR and the initial value of the counter in WTCNT. Set these values so that the time till count overflow becomes at least the same as the clock oscillation settling time.*1
3. After setting the STBY and DEEP bits of the standby control register 1 (STBCR1: see section 52, Power-Down Modes) to 1 and 0 respectively, the execution of a WFI instruction puts the system in software standby mode and clock operation then stops.
4. This module starts counting by detecting the edge change of the NMI signal.
5. When the module count overflows, the clock pulse generator starts supplying the clock and this LSI resumes operation. The WOVF flag in WRCSR is not set when this happens.

Note 1. When the setting of the RTC1XT and RTC0XT bits in the RTCXTALSEL register is 0, the EXTAL clock stops oscillating when the chip is placed on software standby. Set the CKS[3:0] bits in the WTCSR and the initial value of WTCNT to obtain an interval that is greater than the sum of the on-chip oscillation circuit oscillation settling time (t_{ROSC}) of the EXTAL and the internal PLL circuit oscillation settling time (t_{POSC}).
When the setting of either the RTC1XT or RTC0XT bits is 1, the EXTAL clock continues oscillating even on software standby. In this case, set the CKS[3:0] bits in the WTCSR and the initial value of WTCNT to obtain an interval that is greater than the internal PLL circuit oscillation settling time (t_{POSC}).

15.4.2 Using Watchdog Timer Mode

1. Set the WT/IT# bit in WTCSR to 1, the type of count clock in the CKS[3:0] bits in WTCSR, whether this LSI is to be reset internally or not in the RSTE bit in WRCSR and the initial value of the counter in WTCNT.
2. Set the TME bit in WTCSR to 1 to start the count in watchdog timer mode.
3. While operating in watchdog timer mode, rewrite the counter periodically to H'00 to prevent the counter from overflowing.
4. When the counter overflows, this module sets the WOVF flag in WRCSR to 1, and the WDTOVF# signal is output externally (Figure 15.5). The WDTOVF# signal can be used to reset the system. The WDTOVF# signal is output for 64 × P0φ clock cycles.
5. If the RSTE bit in WRCSR is set to 1, a signal to reset the inside of this LSI can be generated simultaneously with the WDTOVF# signal. The internal reset signal is output for 128 × P0φ clock cycles.
6. When an overflow reset of this module is generated simultaneously with a reset input on the RES# pin, the RES# pin reset takes priority, and the WOVF bit in WRCSR is cleared to 0.

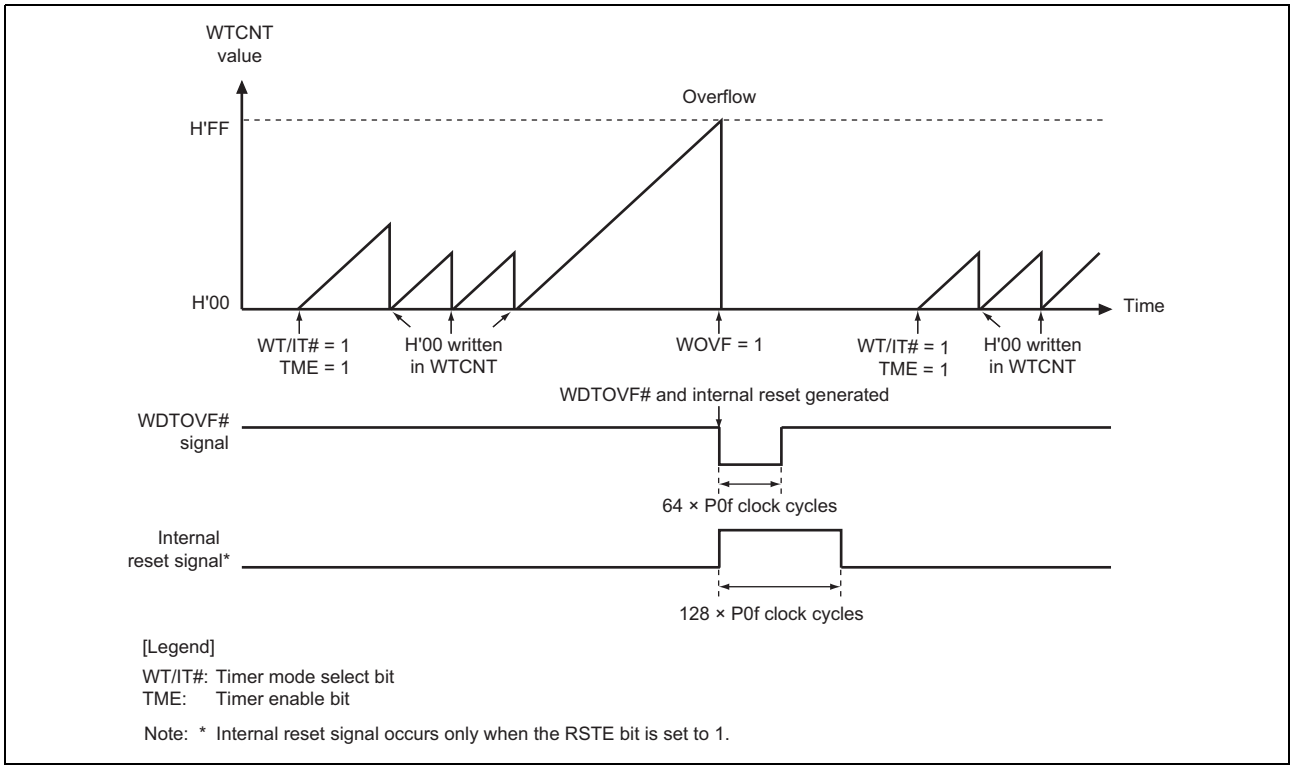


Figure 15.5 Operation in Watchdog Timer Mode

15.4.3 Using Interval Timer Mode

When operating in interval timer mode, interval timer interrupts are generated at every overflow of the counter. This enables interrupts to be generated at set periods.

1. Clear the WT/IT# bit in WTCSR to 0, set the type of count clock in the CKS[3:0] bits in WTCSR, and set the initial value of the counter in WTCNT.
2. Set the TME bit in WTCSR to 1 to start the count in interval timer mode.
3. When the counter overflows, this module sets the IOVF bit in WTCSR to 1 and an interval timer interrupt request is sent to the interrupt controller. The counter then resumes counting.

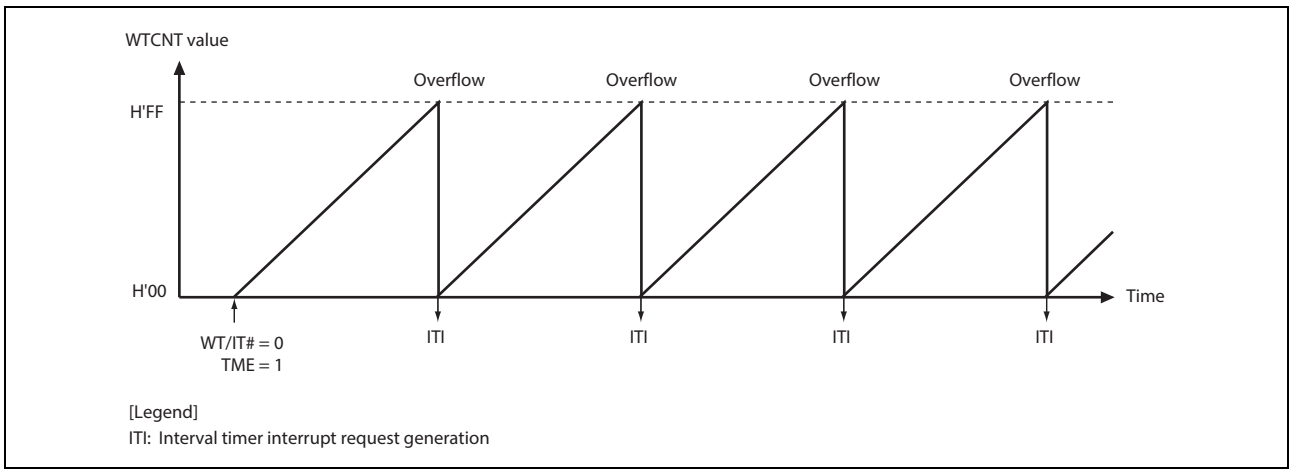


Figure 15.6 Operation in Interval Timer Mode

15.4.4 CPU Parity Error Operation

When CPU parity error corresponding to each PEE bit of the CPU parity error control register (PECR) occurs, an interrupt or internal reset can be generated.

- When PEE = 1 and PERIE = 0, generate an interrupt
  An interrupt can be generated when a CPU parity error occurs. The source of error can be confirmed by the CPU parity error status register (PESR).

- When PEE = 1 and PERIE = 1, generate an internal reset
  An internal reset is generated and PERROUT# is output externally of this module when a CPU parity error occurs. An internal reset signal is output for 128 × P0φ clock period, and the PERROUT# signal is output for 64 × P0φ clock period.
  When the RES# pin input reset signal and the parity error reset of this module occur at the same time, the RES# pin reset takes precedence and the status bit (PEF) of the PESR is cleared to 0.

15.5 Usage Notes

Pay attention to the following points when using this module in either the interval timer or watchdog timer mode.

15.5.1 Timer Variation

After timer operation has started, the period from the power-on reset point to the first incrementation timing of WTCNT varies depending on the time period that is set by the TME bit of WTCSR. The shortest such time period is thus one cycle of the peripheral clock, P0φ, while the longest is the result of frequency division according to the value in the CKS[3:0] bits. The timing of subsequent incrementation is in accord with the selected frequency division ratio. Accordingly, this time difference is referred to as timer variation.
This also applies to the timing of the first incrementation after WTCNT has been written to during timer operation.

15.5.2 Prohibition Against Setting H'FF to WTCNT

When the value in WTCNT reaches H'FF, this module assumes that an overflow has occurred. Accordingly, when H'FF is set in WTCNT, an interval timer interrupt or reset will occur immediately, regardless of the current clock selection by the CKS[3:0] bits.

15.5.3 Interval Timer Overflow Flag

When the value in WTCNT is H'FF, the IOVF flag in WTCSR cannot be cleared.

Only clear the IOVF flag when the value in WTCNT has either become H'00 or been changed to a value other than H'FF.

15.5.4 System Reset by WDTOVF# Signal

If the WDTOVF# signal is input to the RES# pin of this LSI, this LSI cannot be initialized correctly.

Avoid input of the WDTOVF# signal to the RES# pin of this LSI through glue logic circuits. To reset the entire system with the WDTOVF# signal, use the circuit shown in Figure 15.7.

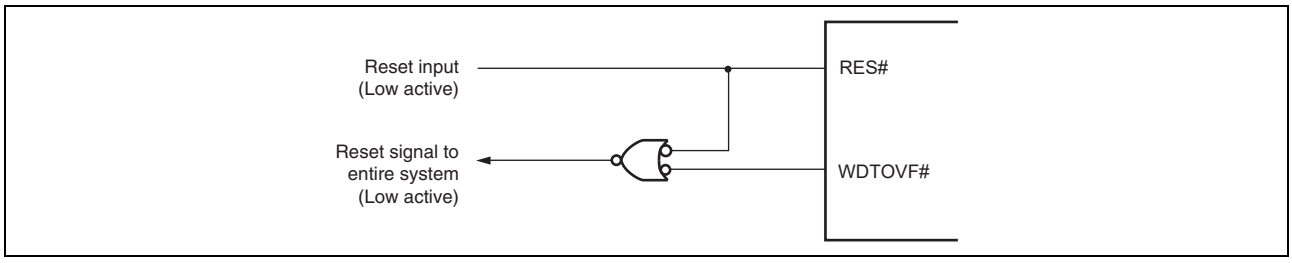


Figure 15.7 Example of System Reset Circuit Using WDTOVF# Signal

15.5.5 Internal Reset in Watchdog Timer Mode

When an internal reset is generated due to an overflow of the watchdog timer counter (WTCNT) in watchdog timer mode, the watchdog reset control/status register (WRCSR) is not initialized, so the WOVF bit retains the value 1. As long as the WOVF bit is 1, an internal reset will not be generated even if the WTCNT overflows again.

16. Realtime Clock (RTC)

This module has two channels of real time clock for secure and non secure.

Note: For details of secure/non secure access, refer to sections 5.8, AXI Protocol Control Signals, 5.8.1 (3), 5.8.2, 5.8.3, and 5.8.4.

Register address

All RTCn register addresses are given as values obtained by adding offsets to the register base address <RTCn_base> for each channel. The base address <RTCn_base> of each RTCn is listed in the following table.

Table 16.1 RTC Base Address

Base Address Name	Base Address
<RTC0_base>	FCFF D000H
<RTC1_base>	FCFF 1000H

Each channel is identified by the index "n". (n = 0: secure, 1: non-secure)

16.1 Features

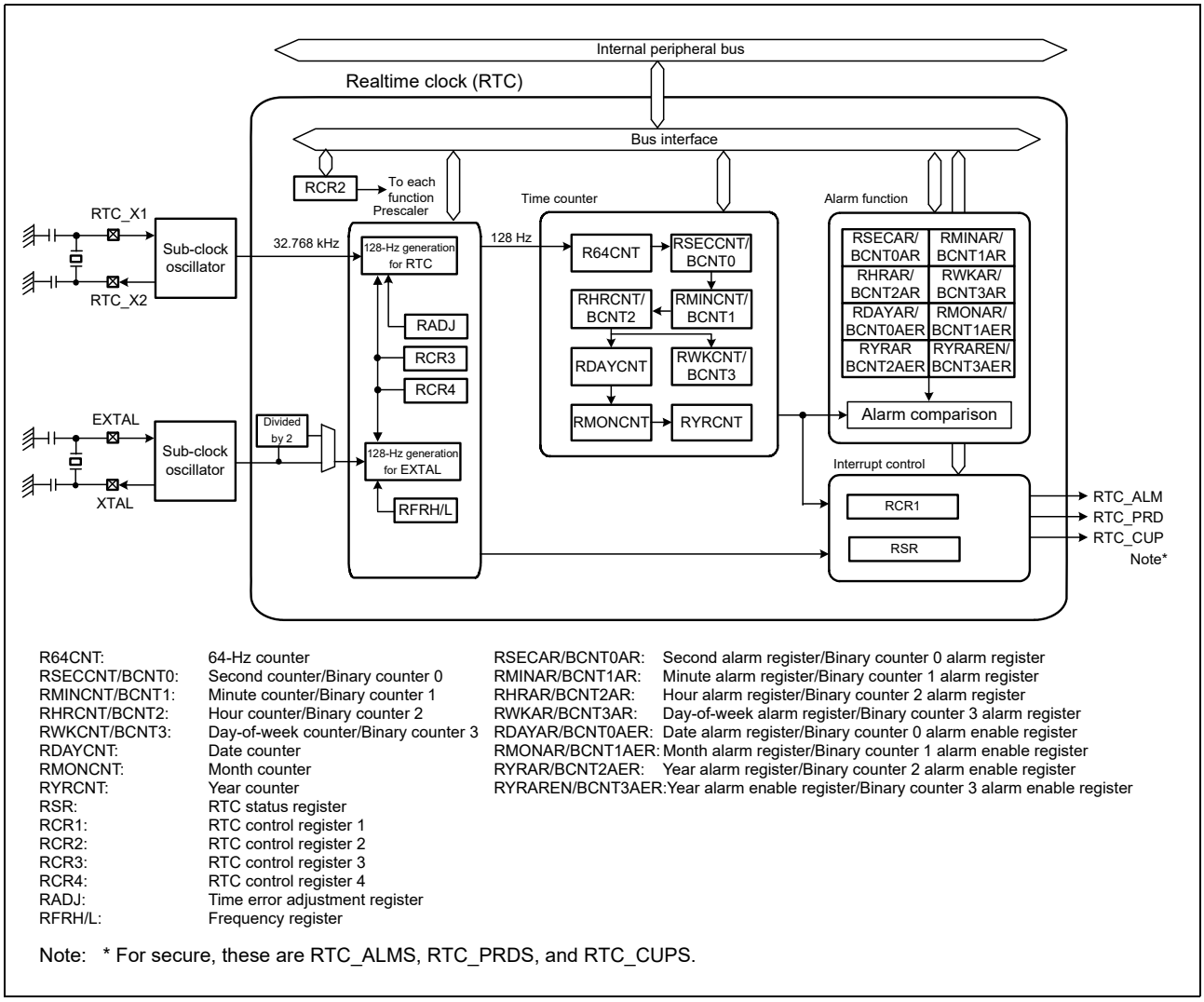
The RTC has two counting modes, calendar count mode and binary count mode, that are used by switching register settings. For calendar count mode, the RTC has a 100 year calendar from 2000 to 2099 and automatically adjusts dates for leap years. For binary count mode, the RTC counts seconds and retains the information as a serial value. Binary count mode can be used for calendars other than the Gregorian (Western) calendar.
The RTC uses a 128-Hz clock acquired by dividing the count source by a prescaler. Year, month, date, day-of-week, a.m./p.m. (in 12-hour mode), hour, minute, second, or 32-bit binary are counted in 1/128 second units.

Table 16.2 lists the specifications of the RTC, Figure 16.1 shows a block diagram of the RTC, and Table 16.3 shows the pin configuration of the RTC.

Table 16.2 RTC specifications

Item	Description
Count mode	Calendar count mode/binary count mode
Count source*1	Realtime clock (RTC_X1) or Internal clock (EXTAL)
Clock and calendar functions	• Calendar count mode Year, month, date, day of week, hour, minute, second are counted, BCD display 12 hours/24 hours mode switching function 30 seconds adjustment function (a number less than 30 is rounded down to 00 seconds, and 30 seconds or more are rounded up to 1 minute) Automatic adjustment function for leap years • Binary count mode Count seconds in 32 bits, binary display • Common to both modes Start/stop function The sub-second digit is displayed in binary units (1 Hz, 2 Hz, 4 Hz, 8 Hz, 16 Hz, 32 Hz, or 64 Hz). Clock error correction function
Interrupts	• Alarm interrupt (RTC_ALM) As an alarm interrupt condition, selectable for comparison with the following: Calendar count mode: Year, month, date, day-of-week, hour, minute, or second can be selected Binary count mode: Each bit of the 32-bit binary counter • Periodic interrupt (RTC_PRD) 2 seconds, 1 second, 1/2 second, 1/4 second, 1/8 second, 1/16 second, 1/32 second, 1/64 second, 1/128 second, or 1/256 second can be selected as an interrupt period. • Carry interrupt (RTC_CUP) An interrupt is generated at either of the following conditions: - When a carry from the 64-Hz counter to the second counter is generated. - When the 64-Hz counter is changed and R64CNT is read at the same time. • Recovery from deep standby is possible by an alarm interrupt.

Note 1. Satisfy the frequency of the peripheral module clock (clkp0d) ≥ the frequency of the count source clock.



R64CNT:	64-Hz counter
RSECCNT/BCNT0:	Second counter/Binary counter 0
RMINCNT/BCNT1:	Minute counter/Binary counter 1
RHRCNT/BCNT2:	Hour counter/Binary counter 2
RWKCNT/BCNT3:	Day-of-week counter/Binary counter 3
RDAYCNT:	Date counter
RMONCNT:	Month counter
RYRCNT:	Year counter
RSR:	RTC status register
RCR1:	RTC control register 1
RCR2:	RTC control register 2
RCR3:	RTC control register 3
RCR4:	RTC control register 4
RADJ:	Time error adjustment register
RFRH/L:	Frequency register

RSECAR/BCNT0AR:	Second alarm register/Binary counter 0 alarm register
RMINAR/BCNT1AR:	Minute alarm register/Binary counter 1 alarm register
RHRAR/BCNT2AR:	Hour alarm register/Binary counter 2 alarm register
RWKAR/BCNT3AR:	Day-of-week alarm register/Binary counter 3 alarm register
RDAYAR/BCNT0AER:	Date alarm register/Binary counter 0 alarm enable register
RMONAR/BCNT1AER:	Month alarm register/Binary counter 1 alarm enable register
RYRAR/BCNT2AER:	Year alarm register/Binary counter 2 alarm enable register
RYRAREN/BCNT3AER:	Year alarm enable register/Binary counter 3 alarm enable register

Note: * For secure, these are RTC_ALMS, RTC_PRDS, and RTC_CUPS.

Figure 16.1 RTC block diagram

Table 16.3 Pin configuration of RTC

Name	Pin name	I/O	Function
Crystal resonator terminal / External Clock for Realtime clock	RTC_X1	Input	Connect a 32.768 kHz Crystal Resonator for this module. Alternatively, External Clock can be input to the RTC_X1.
	RTC_X2	Output	
Crystal oscillator for internal clock / External clock,	EXTAL	Input	Connect a crystal resonator used for internal operation. For details, see section 6, Clock Pulse Generator.
	XTAL	Output	

16.2 Register Descriptions

Write or read from the RTC registers in accordance with section 16.5.4, Notes When Writing to and Reading from Registers.

If the value in an RTC register after a reset is given as x (undefined bits) in the list, it is not initialized by a reset.

When RTC enters the reset state or a low power consumption state during counting operations, for example while the RCR2.START bit is 1, the year, month, day of the week, date, hours, minutes, seconds, and 64-Hz counters continue to operate.

Note: A reset generated while writing to a register might destroy the register value. In addition, do not allow the chip to enter software standby mode immediately after setting any of these registers. For details, see section 16.5.3, Transitions to Low Power Consumption Modes after Setting Registers.

Table 16.4 lists the register configuration.

Table 16.4 Register Configuration

Register Name	Symbol	Value after reset*1	Address	Access size
64-Hz Counter	R64CNT	xxh	RTCn_BASE+00h	8
Second Counter	RSECCNT	xxh	RTCn_BASE+02h	8
Binary Counter 0	BCNT0			
Minute Counter	RMINCNT	xxh	RTCn_BASE+04h	8
Binary Counter 1	BCNT1			
Hour Counter	RHRCNT	xxh	RTCn_BASE+06h	8
Binary Counter 2	BCNT2			
Day-of-Week Counter	RWKCNT	xxh	RTCn_BASE+08h	8
Binary Counter 3	BCNT3			
Day Counter	RDAYCNT	xxh	RTCn_BASE+0Ah	8
Month Counter	RMONCNT	xxh	RTCn_BASE+0Ch	8
Year Counter	RYRCNT	00xxh	RTCn_BASE+0Eh	16
Second Alarm Register	RSECAR	xxh	RTCn_BASE+10h	8
Binary Counter 0 Alarm Register	BCNT0AR			
Minute Alarm Register	RMINAR	xxh	RTCn_BASE+12h	8
Binary Counter 1 Alarm Register	BCNT1AR			
Hour Alarm Register	RHRAR	xxh	RTCn_BASE+14h	8
Binary Counter 2 Alarm Register	BCNT2AR			
Day-of-Week Alarm Register	RWKAR	xxh	RTCn_BASE+16h	8
Binary Counter 3 Alarm Register	BCNT3AR			
Day Alarm Register	RDAYAR	xxh	RTCn_BASE+18h	8
Binary Counter 0 Alarm Enable Register	BCNT0AER			
Month Alarm Register	RMONAR	xxh	RTCn_BASE+1Ah	8
Binary Counter 1 Alarm Enable Register	BCNT1AER			
Year Alarm Register	RYRAR	00xxh	RTCn_BASE+1Ch	16
Binary Counter 2 Alarm Enable Register	BCNT2AER			
Year Alarm Enable Register	RYRAREN	xxh	RTCn_BASE+1Eh	8
Binary Counter 3 Alarm Enable Register	BCNT3AER			
RTC Status Register	RSR	00h*2	RTCn_BASE+20h	8
RTC Control Register 1	RCR1	xxh	RTCn_BASE+22h	8
RTC Control Register 2	RCR2	xxh	RTCn_BASE+24h	8
RTC Control Register 3	RCR3	0xh	RTCn_BASE+26h	8
RTC Control Register 4	RCR4	0xh	RTCn_BASE+28h	8
Frequency Register H/L	RFRH	000xh	RTCn_BASE+2Ah	16
	RFRL	xxxxh	RTCn_BASE+2Ch	16
Time Error Adjustment Register	RADJ	xxh	RTCn_BASE+2Eh	8

Note 1. If the value in an RTC register after a reset is given as X (undefined bits), it is not initialized by a reset or the deep standby mode. When transitioning to the reset state or low power consumption state during count operation (RCR2.START bit is 1); the year, month, day, day of week, hour, minute, and second counters continue to operate in the 64-Hz counter mode or the calendar count mode, and the binary counter continues to operate in the binary count mode.
A reset generated while writing to a register might destroy the register value. In addition, do not allow the chip to enter Software Standby mode immediately after setting any of these registers. For details, see section 16.5.3, Transitions to Low Power Consumption Modes after Setting Registers.

Note 2. The value read from this register after a reset might vary from 01h to 07h.

16.2.1 64-Hz Counter (R64CNT)

Address(es): RTC.R64CNT FCFF D000h [secure], FCFF 1000h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	—	F1HZ	F2HZ	F4HZ	F8HZ	F16HZ	F32HZ	F64HZ
Value after reset:	0	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b0	F64HZ	64 Hz	Indicate the state between 1 Hz and 64 Hz of the sub-second digit.	R
b1	F32HZ	32 Hz		R
b2	F16HZ	16 Hz		R
b3	F8HZ	8 Hz		R
b4	F4HZ	4 Hz		R
b5	F2HZ	2 Hz		R
b6	F1HZ	1 Hz		R
b7	—	Reserved	This bit is read as 0. Writing to this bit has no effect.	R

R64CNT is used in both calendar count mode and in binary count mode. The 64-Hz counter (R64CNT) generates the period for a second by counting up periods of the 128-Hz clock. The state in the sub-second range can be confirmed by reading this counter.

This counter is set to 00h by an RTC software reset or an execution of a 30-second adjustment. To read this counter, follow the procedure in section 16.3.5, Reading 64-Hz Counter and Time.

16.2.2 Second Counter (RSECCNT) / Binary Counter 0 (BCNT0)

(1) In calendar count mode:

Address(es): RTC.RSECCNT FCFF D002h [secure], FCFF 1002h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	—	SEC10[2:0]			SEC1[3:0]			
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b3 to b0	SEC1[3:0]	1-Second Count	Counts from 0 to 9 every second. When a carry is generated, 1 is added to the tens place.	R/W
b6 to b4	SEC10[2:0]	10-Second Count	Counts from 0 to 5 for 60-second counting.	R/W
b7	—	Reserved	Set this bit to 0. It is read as the set value.	R/W

RSECCNT sets and counts the BCD-coded second value. It counts carries generated once per second in the 64-Hz counter.

The setting range is decimal 00 to 59. The RTC does not operate normally if any other value is set. Before writing to this register, be sure to stop the count operation using the START bit in RCR2.

To read this counter, follow the procedure in section 16.3.5, Reading 64-Hz Counter and Time.

(2) In binary count mode:

Address(es): RTC.BCNT0 FCFF D002h [secure], FCFF 1002h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	BCNT[7:0]							
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

BCNT0 is a read/write 32-bit binary counter b7 to b0. The 32-bit binary counter performs count operation by a carry generated for each second of the 64-Hz counter. Before writing to this register, be sure to stop the count operation using the START bit in RCR2. To read this counter, follow the procedure in section 16.3.5, Reading 64-Hz Counter and Time.

16.2.3 Minute Counter (RMINCNT) / Binary Counter 1 (BCNT1)

(1) In calendar count mode:

Address(es): RTC.RMINCNT FCFF D004h [secure], FCFF 1004h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	—	MIN10[2:0]			MIN1[3:0]			
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b3 to b0	MIN1[3:0]	1-Minute Count	Counts from 0 to 9 every minute. When a carry is generated, 1 is added to the tens place.	R/W
b6 to b4	MIN10[2:0]	10-Minute Count	Counts from 0 to 5 for 60-minute counting.	R/W
b7	—	Reserved	Set this bit to 0. It is read as the set value.	R/W

RMINCNT sets and counts the BCD-coded minute value. It counts carries generated once per minute in the second counter.

A value from 00 through 59 (in BCD) can be specified. If a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation using the START bit in RCR2. To read this counter, follow the procedure in section 16.3.5, Reading 64-Hz Counter and Time.

(2) In binary count mode:

Address(es): RTC.BCNT1 FCFF D004h [secure], FCFF 1004h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	BCNT[15:8]							
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

BCNT1 is a readable/writable 32-bit binary counter b15 to b8. The 32-bit binary counter performs count operation by a carry generated for each second of the 64-Hz counter. Before writing to this register, be sure to stop the count operation using the START bit in RCR2. To read this counter, follow the procedure in section 16.3.5, Reading 64-Hz Counter and Time.

16.2.4 Hour Counter (RHRCNT) / Binary Counter 2 (BCNT2)

(1) In calendar count mode:

Address(es): RTC.RHRCNT FCFF D006h [secure], FCFF 1006h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	—	PM	HR10[1:0]		HR1[3:0]			
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b3 to b0	HR1[3:0]	1-Hour Count	Counts from 0 to 9 once every hour. When a carry is generated, 1 is added to the tens place.	R/W
b5, b4	HR10[1:0]	10-Hour Count	Counts from 0 to 2 once per carry from the ones place.	R/W
b6	PM	PM	Time counter setting for a.m./p.m. 0: a.m. 1: p.m.	R/W
b7	—	Reserved	Set this bit to 0. It is read as the set value.	R/W

RHRCNT sets and counts the BCD-coded hour value. It counts carries generated once every hour in the minute counter. The specifiable time differs according to the setting in the hours mode bit (RCR2.HR24):

- When the RCR2.HR24 bit is 0 – from 00 to 11 (in BCD)
- When the RCR2.HR24 bit is 1 – from 00 to 23 (in BCD).

If a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation using the START bit in RCR2. The PM bit is only enabled when the RCR2.HR24 bit is 0. Otherwise, the setting in the PM bit has no effect. To read this counter, follow the procedure in section 16.3.5, Reading 64-Hz Counter and Time.

(2) In binary count mode:

Address(es): RTC.BCNT2 FCFF D006h [secure], FCFF 1006h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	BCNT[23:16]							
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

BCNT2 is a read/write 32-bit binary counter b23 to b16. The 32-bit binary counter performs count operation by a carry generated for each second of the 64-Hz counter. Before writing to this register, be sure to stop the count operation using the START bit in RCR2. To read this counter, follow the procedure in section 16.3.5, Reading 64-Hz Counter and Time.

16.2.5 Day-of-Week Counter (RWKCNT) / Binary Counter 3 (BCNT3)

(1) In calendar count mode:

Address(es): RTC.RWKCNT FCFF D008h [secure], FCFF 1008h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	DAYW[2:0]		
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b2 to b0	DAYW[2:0]	Day-of-Week Counting	b2 b0 0 0 0: Sunday 0 0 1: Monday 0 1 0: Tuesday 0 1 1: Wednesday 1 0 0: Thursday 1 0 1: Friday 1 1 0: Saturday 1 1 1: Setting prohibited.	R/W
b7 to b3	—	Reserved	Set these bits to 0. They are read as the set value.	R/W

RWKCNT sets and counts in the coded day-of-week value. It counts carries generated once per day in the hour counter. A value from 0 through 6 can be specified. If a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation using the START bit in RCR2. To read this counter, follow the procedure in section 16.3.5, Reading 64-Hz Counter and Time.

(2) In binary count mode:

Address(es): RTC.BCNT3 FCFF D008h [secure], FCFF 1008h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	BCNT[31:24]							
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

BCNT3 is a read/write 32-bit binary counter b31 to b24 that performs count operation by a carry generated for each second of the 64-Hz counter. Before writing to this register, be sure to stop the count operation using the START bit in RCR2. To read this counter, follow the procedure in section 16.3.5, Reading 64-Hz Counter and Time.

16.2.6 Day Counter (RDAYCNT)

Address(es): RTC.RDAYCNT FCFF D00Ah [secure], FCFF 100Ah [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	DATE10[1:0]		DATE1[3:0]			
Value after reset:	0	0	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b3 to b0	DATE1[3:0]	1-Day Count	Counts from 0 to 9 once per day. When a carry is generated, 1 is added to the tens place	R/W
b5, b4	DATE10[1:0]	10-Day Count	Counts from 0 to 3 once per carry from the ones place	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

RDAYCNT is used in calendar count mode to set and count the BCD-coded date value. It counts carries generated once per day in the hour counter. The count operation depends on the month and whether the year is a leap year. The value 00 in RYRCNT is regarded as year 2000. Leap years are determined according to whether the RYRCNT value is divisible by 400, 100, and 4.

A value from 01 through 31 (in BCD) can be specified. If a value outside of this range is specified, the RTC does not operate correctly. When specifying a value, the range of specifiable days depends on the month and whether the year is a leap year. Before writing to this register, be sure to stop the count operation using the START bit in RCR2. To read this counter, follow the procedure in section 16.3.5, Reading 64-Hz Counter and Time.

16.2.7 Month Counter (RMONCNT)

Address(es): RTC.RMONCNT FCFF D00Ch [secure], FCFF100Ch [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	MON10	MON1[3:0]			
Value after reset:	0	0	0	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b3 to b0	MON1[3:0]	1-Month Count	Counts from 0 to 9 once per month. When a carry is generated, 1 is added to the tens place	R/W
b4	MON10	10-Month Count	Counts from 0 to 1 once per carry from the ones place	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

RMONCNT is used in calendar count mode to set and count the BCD-coded month value. It counts carries generated once per month in the date counter.

A value from 01 through 12 (in BCD) can be specified. If a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation using the START bit in RCR2. To read this counter, follow the procedure in section section 16.3.5, Reading 64-Hz Counter and Time.

16.2.8 Year Counter (RYRCNT)

Address(es): RTC.RYRCNT FCFF D00Eh [secure], FCFF 100Eh [non secure]

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	YR10[3:0]				YR1[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b3 to b0	YR1[3:0]	1-Year Count	Counts from 0 to 9 once per year. When a carry is generated, 1 is added to the tens place.	R/W
b7 to b4	YR10[3:0]	10-Year Count	Counts from 0 to 9 once per carry from ones place.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

RYRCNT is used in calendar count mode to set and count the BCD-coded year value. It counts carries generated once per year in the month counter.

A value from 00 through 99 (in BCD) can be specified. If a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation using the START bit in RCR2. To read this counter, follow the procedure in section 16.3.5, Reading 64-Hz Counter and Time.

16.2.9 Second Alarm Register (RSECAR) / Binary Counter 0 Alarm Register (BCNT0AR)

(1) In calendar count mode:

Address(es): RTC.RSECAR FCFF D010h [secure], FCFF 1010h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	ENB	SEC10[2:0]			SEC1[3:0]			
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b3 to b0	SEC1[3:0]	1 Second	Value for the ones place of seconds	R/W
b6 to b4	SEC10[2:0]	10 Seconds	Value for the tens place of seconds	R/W
b7	ENB	ENB	0: The register value is not compared with the RSECCNT value 1: The register value is compared with the RSECCNT value.	R/W

RSECAR is an alarm register associated with the BCD-coded second counter RSECCNT. When the ENB bit is set to 1, the RSECAR value is compared with the RSECCNT value. From the following alarm registers, only those selected with the ENB bits set to 1 are compared with the associated counters:

- RSECAR
- RMINAR
- RHRAR
- RWKAR
- RDAYAR
- RMONAR
- RYRAREN.

When all the respective values match, the IR flag associated with the RTC_ALM interrupt is set to 1. The RSECAR values from 00 through 59 (in BCD) can be specified. If a value outside of this range is specified, the RTC does not operate correctly. This register is set to 00h by an RTC software reset.

(2) In binary count mode:

Address(es): RTC.BCNT0AR FCFF D010h [secure], FCFF 1010h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	BCNTAR[7:0]							
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

BCNT0AR is a read/write alarm register associated with the 32-bit binary counter b7 to b0. This register is set to 00h by an RTC software reset.

16.2.10 Minute Alarm Register (RMINAR) / Binary Counter 1 Alarm Register (BCNT1AR)

(1) In calendar count mode:

Address(es): RTC.RMINAR FCFF D012h [secure], FCFF 1012h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	ENB	MIN10[2:0]			MIN1[3:0]			
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b3 to b0	MIN1[3:0]	1 Minute	Value for the ones place of minutes	R/W
b6 to b4	MIN10[2:0]	10 Minutes	Value for the tens place of minutes	R/W
b7	ENB	ENB	0: The register value is not compared with the RMINCNT value 1: The register value is compared with the RMINCNT value.	R/W

RMINAR is an alarm register associated with the BCD-coded minute counter RMINCNT. When the ENB bit is set to 1, the RMINAR value is compared with the RMINCNT value. From the following alarm registers, only those selected with the ENB bits set to 1 are compared with the associated counters:

- RSECAR
- RMINAR
- RHRAR
- RWKAR
- RDAYAR
- RMONAR
- RYRAREN.

When all the respective values match, the IR flag associated with the RTC_ALM interrupt is set to 1. The RMINAR values from 00 through 59 (in BCD) can be specified. If a value outside of this range is specified, the RTC does not operate correctly. This register is set to 00h by an RTC software reset.

(2) In binary count mode:

Address(es): RTC.BCNT1AR FCFF D012h [secure], FCFF 1012h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	BCNTAR[15:8]							
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

BCNT1AR is a read/write alarm register associated with the 32-bit binary counter from b15 to b8. This register is set to 00h by an RTC software reset.

16.2.11 Hour Alarm Register (RHRAR) / Binary Counter 2 Alarm Register (BCNT2AR)

(1) In calendar count mode:

Address(es): RTC.RHRAR FCFF D014h [secure], FCFF 1014h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	ENB	PM	HR10[1:0]		HR1[3:0]			
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b3 to b0	HR1[3:0]	1 Hour	Value for the ones place of hours	R/W
b5, b4	HR10[1:0]	10 Hours	Value for the tens place of hours	R/W
b6	PM	PM	Time Alarm Setting for a.m./p.m. 0: a.m. 1: p.m.	R/W
b7	ENB	ENB	0: The register value is not compared with the RHRCNT value 1: The register value is compared with the RHRCNT value.	R/W

RHRAR is an alarm register associated with the BCD-coded hour counter RHRCNT. When the ENB bit is set to 1, the RHRAR value is compared with the RHRCNT value. From the following alarm registers, only those selected with the ENB bits set to 1 are compared with the associated counters:

- RSECAR
- RMINAR
- RHRAR
- RWKAR
- RDAYAR
- RMONAR
- RYRAREN.

When all the respective values match, the IR flag associated with the RTC_ALM interrupt is set to 1. The specifiable time differs according to the setting in the hours mode bit (RCR2.HR24):

- When the RCR2.HR24 bit is 0 – From 00 to 11 (in BCD)
- When the RCR2.HR24 bit is 1 – From 00 to 23 (in BCD).

If a value outside of this range is specified, the RTC does not operate correctly. When the RCR2.HR24 bit is 0, be sure to set the PM bit. When the RCR2.HR24 bit is 1, the setting in the PM bit has no effect. This register is set to 00h by an RTC software reset.

(2) In binary count mode:

Address(es): RTC.BCNT2AR FCFF D014h [secure], FCFF 1014h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	BCNTAR[23:16]							
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

BCNT2AR is a read/write alarm register associated with the 32-bit binary counter b23 to b16. This register is set to 00h by an RTC software reset.

16.2.12 Day-of-Week Alarm Register (RWKAR) / Binary Counter 3 Alarm Register (BCNT3AR)

(1) In calendar count mode:

Address(es): RTC.RWKAR FCFF D016h [secure], FCFF 1016h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	ENB	—	—	—	—	DAYW[2:0]		
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b2 to b0	DAYW[2:0]	Day-of-Week Setting	b2 b0 0 0 0: Sunday 0 0 1: Monday 0 1 0: Tuesday 0 1 1: Wednesday 1 0 0: Thursday 1 0 1: Friday 1 1 0: Saturday 1 1 1: Setting prohibited.	R/W
b6 to b3	—	Reserved	Set these bits to 0. They are read as the set value.	R/W
b7	ENB	ENB	0: The register value is not compared with the RWKCNT value 1: The register value is compared with the RWKCNT value.	R/W

RWKAR is an alarm register associated with the coded day-of-week counter RWKCNT. When the ENB bit is set to 1, the RWKAR value is compared with the RWKCNT value. From the following alarm registers, only those selected with the ENB bits set to 1 are compared with the corresponding counters:

- RSECAR
- RMINAR
- RHRAR
- RWKAR
- RDAYAR
- RMONAR
- RYRAREN.

When all the respective values all match, the IR flag associated with the RTC_ALM interrupt is set to 1. The RWKAR values from 0 through 6 (in BCD) can be specified. If a value outside of this range is specified, the RTC does not operate correctly. This register is set to 00h by an RTC software reset.

(2) In binary count mode:

Address(es): RTC.BCNT3AR FCFF D016h [secure], FCFF 1016h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	BCNTAR[31:24]							
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

BCNT3AR is a read/write alarm register associated with the 32-bit binary counter b31 to b24. This register is set to 00h by an RTC software reset.

16.2.13 Day Alarm Register (RDAYAR) / Binary Counter 0 Alarm Enable Register (BCNT0AER)

(1) In calendar count mode:

Address(es): RTC.RDAYAR FCFF D018h [secure], FCFF 1018h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	ENB	—	DATE10[1:0]		DATE1[3:0]			
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b3 to b0	DATE1[3:0]	1 Day	Value for the ones place of days	R/W
b5, b4	DATE10[1:0]	10 Days	Value for the tens place of days	R/W
b6	—	Reserved	Set this bit to 0. It is read as the set value.	R/W
b7	ENB	ENB	0: The register value is not compared with the RDAYCNT value 1: The register value is compared with the RDAYCNT value.	R/W

RDAYAR is an alarm register associated with the BCD-coded date counter RDAYCNT. When the ENB bit is set to 1, the RDAYAR value is compared with the RDAYCNT value. From the following alarm registers, only those selected with the ENB bits set to 1 are compared with the corresponding counters:

- RSECAR
- RMINAR
- RHRAR
- RWKAR
- RDAYAR
- RMONAR
- RYRAREN.

When all the respective values match, the IR flag associated with the RTC_ALM interrupt is set to 1. The RDAYAR values from 01 through 31 (in BCD) can be specified. If a value outside of this range is specified, the RTC does not operate correctly. This register is set to 00h by an RTC software reset.

(2) In binary count mode:

Address(es): RTC.BCNT0AER FCFF D018h [secure], FCFF 1018h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	ENB[7:0]							
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

BCNT0AER is a read/write register to set the alarm enable associated with the 32-bit binary counter b7 to b0. The binary counter (BCNT[31:0]) associated with the ENB[31:0] bits that are set to 1 is compared with the binary alarm register (BCNTAR[31:0]), and when all match, the IR flag associated with the RTC_ALM interrupt becomes 1. This register is set to 00h by an RTC software reset.

16.2.14 Month Alarm Register (RMONAR) / Binary Counter 1 Alarm Enable Register (BCNT1AER)

(1) In calendar count mode:

Address(es): RTC.RMONAR FCFF D01Ah [secure], FCFF 101Ah [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	ENB	—	—	MON10	MON1[3:0]			
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b3 to b0	MON1[3:0]	1 Month	Value for the ones place of months	R/W
b4	MON10	10 Months	Value for the tens place of months	R/W
b6, b5	—	Reserved	Set these bits to 0. They are read as the set value.	R/W
b7	ENB	ENB	0: The register value is not compared with the RMONCNT value 1: The register value is compared with the RMONCNT value.	R/W

RMONAR is an alarm register associated with the BCD-coded month counter RMONCNT. When the ENB bit is set to 1, the RMONAR value is compared with the RMONCNT value. From the following alarm registers, only those selected with the ENB bits set to 1 are compared with the associated counters:

- RSECAR
- RMINAR
- RHRAR
- RWKAR
- RDAYAR
- RMONAR
- RYRAREN.

When all the respective values match, the IR flag associated with the RTC_ALM interrupt is set to 1. The RMONAR values from 01 through 12 (in BCD) can be specified. If a value outside of this range is specified, the RTC does not operate correctly. This register is set to 00h by an RTC software reset.

(2) In binary count mode:

Address(es): RTC.BCNT1AER FCFF D01Ah [secure], FCFF 101Ah [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	ENB[15:8]							
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

BCNT1AER is a read/write register for setting the alarm enable associated with the 32-bit binary counter b15 to b8. The binary counter (BCNT[31:0]) associated with the ENB[31:0] bits that are set to 1 is compared with the binary alarm register (BCNTAR[31:0]), and when all match, the IR flag associated with the RTC_ALM interrupt becomes 1. This register is set to 00h by an RTC software reset.

16.2.15 Year Alarm Register (RYRAR) / Binary Counter 2 Alarm Enable Register (BCNT2AER)

(1) In calendar count mode:

Address(es): RTC.RYRAR FCFF D01Ch [secure], FCFF 101Ch [non secure]

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	YR10[3:0]				YR1[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b3 to b0	YR1[3:0]	1 Year	Value for the ones place of years	R/W
b7 to b4	YR10[3:0]	10 Years	Value for the tens place of years	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

RYRAR is an alarm register associated with the BCD-coded year counter RYRCNT. The RYRAR values from 00 through 99 (in BCD) can be specified. If a value outside of this range is specified, the RTC does not operate correctly. This register is set to 0000h by an RTC software reset.

(2) In binary count mode:

Address(es): RTC.BCNT2AER FCFF D01Ch [secure], FCFF 101Ch [non secure]

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ENB[23:16]							
Value after reset:	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x

x: Undefined

BCNT2AER is a read/write register for setting the alarm enable associated with the 32-bit binary counter b23 to b16. The binary counter (BCNT[31:0]) associated with the ENB[31:0] bits that are set to 1 is compared with the binary alarm register (BCNTAR[31:0]), and when all match, the IR flag associated with the RTC_ALM interrupt becomes 1. This register is set to 0000h by an RTC software reset.

16.2.16 Year Alarm Enable Register (RYRAREN) / Binary Counter 3 Alarm Enable Register (BCNT3AER)

(1) In calendar count mode:

Address(es): RTC.RYRAREN FCFF D01Eh [secure], FCFF 101Eh [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	ENB	—	—	—	—	—	—	—
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b6 to b0	—	Reserved	Set these bits to 0. They are read as the set value.	R/W
b7	ENB	ENB	0: The register value is not compared with the RYRCNT value. 1: The register value is compared with the RYRCNT value.	R/W

When the ENB bit in RYRAREN is set to 1, the RYRAR value is compared with the RYRCNT value. From the following alarm registers, only those selected with the ENB bits set to 1 are compared with the associated counters:

- RSECAR
- RMINAR
- RHRAR
- RWKAR
- RDAYAR
- RMONAR
- RYRAREN.

When all the respective values match, the IR flag associated with the RTC_ALM interrupt is set to 1.This register is set to 00h by an RTC software reset.

(2) In binary count mode:

Address(es): RTC.BCNT3AER FCFF D01Eh [secure], FCFF 101Eh [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	ENB[31:24]							
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

BCNT3AER is a read/write register for setting the alarm enable associated with the 32-bit binary counter b31 to b24. The binary counter (BCNT[31:0]) associated with the ENB[31:0] bits that are set to 1 is compared with the binary alarm register (BCNTAR[31:0]), and when all match, the IR flag associated with the RTC_ALM interrupt becomes 1. This register is set to 00h by an RTC software reset.

16.2.17 RTC Status Register 1 (RSR)

Address(es): RTC.RSR FCFF D020h [secure], FCFF 1020h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	PF	CF	AF
Value after reset:	0	0	0	0	0	0*1	0*1	0*1

Note 1. The value may be read as 1 after release from the reset state.

Bit	Symbol	Bit name	Description	R/W
b0	AF	Alarm Flag	0: The value of the counter does not match that of the alarm register. 1: The value of the counter matches that of the alarm register. Note: Writing 1 to this flag has no effect.	R/W
b1	CF	Carry Flag	0: A carry from the second counter/binary counter 0 or from the 64-Hz counter was not generated during reading of the 64-Hz counter. 1: A carry from the second counter/binary counter 0 or from the 64-Hz counter was generated during reading of the 64-Hz counter.	R/W
b2	PF	Periodic Interrupt Flag	0: An interrupt was not generated after the period specified by the RCR1.PES[3:0] bits. 1: An interrupt was generated after the period specified by the RCR1.PES[3:0] bits. Note: Writing 1 to this flag has no effect.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R

RSR is the flag register for the periodic interrupt, carry, and alarm. This register is common to the calendar count mode and the binary count mode.
Each flag is set to 1 when the prescaler or the clock counter matches the given interrupt setting condition. The prescaler, clock counter, and the setting register of each interrupt are not reset, so each flag may be set by the time it is read. This register is set to 00h by an RTC software reset

AF flag (Alarm Flag)

This flag is set to 1 when the counter matches the alarm time set with the alarm registers in which the ENB bit is set to 1 (calendar count mode: RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAR; binary count mode: BCNT0AR, BCNT1AR, BCNT2AR, and BCNT3AR).
[Clearing condition]

- 0 is written to the AF flag.

[Setting condition]

- The counter matches the alarm registers (only including those registers in which the ENB bit is set to 1).

CF flag (Carry Flag)

The setting of this flag being 1 indicates that a carry from the second counter/binary counter 0 or from the 64-Hz counter was generated during reading of the 64-Hz counter. A value of the counter read when this flag is 1 is not guaranteed. Be sure to read the counter again in such cases.
[Clearing condition]

- 0 is written to the CF flag.

[Setting conditions]

- A carry from the second counter/binary counter 0 or from the 64-Hz counter during reading of the 64-Hz counter
- 1 is written to the CF flag.

PF flag (Periodic Interrupt Flag)

This flag indicates that an interrupt was generated after the period specified by the RCR1.PES[3:0] bits. This flag is set to "1" when the interrupt is generated.

[Clearing condition]

- 0 is written to the PF flag.

[Setting condition]

- An interrupt after the period specified by the RCR1.PES[3:0] bits.

16.2.18 RTC Control Register 1 (RCR1)

Address(es): RTC.RCR1 FCFF D022h [secure], FCFF 1022h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	PES[3:0]				—	PIE	CIE	AIE
Value after reset:	x	x	x	x	0	x	0	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b0	AIE	Alarm Interrupt Enable	0: An alarm interrupt request is disabled 1: An alarm interrupt request is enabled.	R/W
b1	CIE	Carry Interrupt Enable	0: A carry interrupt request is disabled 1: A carry interrupt request is enabled.	R/W
b2	PIE	Periodic Interrupt Enable	0: A periodic interrupt request is disabled 1: A periodic interrupt request is enabled.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b7 to b4	PES[3:0]	Periodic Interrupt Select	b7 b4 0 1 1 0: A periodic interrupt is generated every 1/256 second*1 0 1 1 1: A periodic interrupt is generated every 1/128 second 1 0 0 0: A periodic interrupt is generated every 1/64 second 1 0 0 1: A periodic interrupt is generated every 1/32 second 1 0 1 0: A periodic interrupt is generated every 1/16 second 1 0 1 1: A periodic interrupt is generated every 1/8 second 1 1 0 0: A periodic interrupt is generated every 1/4 second 1 1 0 1: A periodic interrupt is generated every 1/2 second 1 1 1 0: A periodic interrupt is generated every 1 second 1 1 1 1: A periodic interrupt is generated every 2 seconds. Other than above: No periodic interrupts are generated.	R/W

Note 1. When EXTAL is selected (RCR4.RCKSEL = 1) while PES[3:0] = 0110b, a periodic interrupt is generated every 1/128 second.

RCR1 is used in both calendar count mode and in binary count mode. Bits AIE, PIE, and PES[3:0] are updated synchronously with the count source. When the RCR1 register is modified, check that all the bits are updated before proceeding.

AIE bit (Alarm Interrupt Enable)

This bit enables or disables alarm interrupt requests. When the counter matches the alarm time during deep standby, this LSI return from deep standby regardless of this bit set.

CIE bit (Carry Interrupt Enable)

This bit enables and disables interrupt requests when a carry to the RSECCNT/BCNT0 occurs, or when a carry to the 64-Hz counter (R64CNT) occurs while reading the 64-Hz counter.

PIE bit (Periodic Interrupt Enable)

This bit enables or disabled a periodic interrupt.

PES[3:0] bits (Periodic Interrupt Select)

These bits specify the period for the periodic interrupt. A periodic interrupt is generated with the period specified by these bits.

16.2.19 RTC Control Register 2 (RCR2)

(1) In calendar count mode:

Address(es): RTC.RCR2 FCFF D024h [secure], FCFF 1024h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	CNT MD	HR24	AADJP	AADJE	—	ADJ30	RESET	START
Value after reset:	x	x	x	x	0	0	0	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b0	START	Start	0: Prescaler and time counter are stopped 1: Prescaler and time counter operate normally.	R/W
b1	RESET	RTC Software Reset	• In writing: 0: Writing is invalid 1: The prescaler and the target registers for RTC software reset*1 are initialized • In reading: 0: In normal time operation, or an RTC software reset has completed 1: During an RTC software reset.	R/W
b2	ADJ30	30-Second Adjustment	• In writing: 0: Writing is invalid 1: 30-second adjustment is executed. • In reading: 0: In normal time operation, or 30-second adjustment has completed 1: During 30-second adjustment.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	AADJE	Automatic Adjustment Enable*2	0: Automatic adjustment is disabled 1: Automatic adjustment is enabled.	R/W
b5	AADJP	Automatic Adjustment Period Select*2	0: The RADJ.ADJ[5:0] setting value is adjusted from the count value of the prescaler every minute 1: The RADJ.ADJ[5:0] setting value is adjusted from the count value of the prescaler every 10 seconds.	R/W
b6	HR24	Hours Mode	0: The RTC operates in 12-hour mode 1: The RTC operates in 24-hour mode.	R/W
b7	CNTMD	Count Mode Select	0: Calendar count mode 1: Binary count mode.	R/W

Note 1. R64CNT, RSECAR/BCNT0AR, RMINAR/BCNT1AR, RHRAR/BCNT2AR, RWKAR/BCNT3AR, RDAYAR/BCNT0AER, RMONAR/BCNT1AER, RYRAR/BCNT2AER, RYRAREN/BCNT3AER, RADJ, RCR2.ADJ30, RCR2.AADJE, RCR2.AADJP
Note 2. When EXTAL is selected, the setting of this bit is disabled.

RCR2 is related to hours mode, automatic adjustment function, 30-second adjustment, RTC software reset, and controlling count operation.

START bit (Start)

This bit stops or restarts the prescaler or time counter operation.
The START bit is updated in synchronization with the next cycle of the count source. When the START bit is modified, check that the bit is updated before proceeding.

RESET bit (RTC Software Reset)

This bit initializes the prescaler and registers to be reset by RTC software.
When 1 is written to the RESET bit, initialization starts in synchronization with the count source. When the initialization is completed, the RESET bit is automatically set to 0. Check that this bit is set to 0 before proceeding.

ADJ30 bit (30-Second Adjustment)

This bit is for 30-second adjustment.
When 1 is written to the ADJ30 bit, the RSECCNT value is rounded down to 00 seconds if it is currently less than 30 seconds and is rounded up to 1 minute if the value is 30 seconds or greater.
The 30-second adjustment is performed in synchronization with the count source. When 1 is written to this bit, the ADJ30 bit is automatically set to 0 after the 30-second adjustment is completed. If 1 is written to the ADJ30 bit, check that the bit is set to 0 before proceeding. When the 30-second adjustment is performed, the prescaler and R64CNT are also reset. The ADJ30 bit is set to 0 by an RTC software reset.

AADJE bit (Automatic Adjustment Enable)

This bit controls (enables or disables) automatic adjustment.
Set the plus–minus bits (RADJ.PMADJ[1:0]) to 00b (adjustment is not performed) before changing the value of the AADJE bit.
The AADJE bit is set to 0 by an RTC software reset.

AADJP bit (Automatic Adjustment Period Select)

This bit selects the automatic-adjustment period.
Set the plus–minus bits (RADJ.PMADJ[1:0]) to 00b (adjustment is not performed) before changing the value of the AADJP bit.
The AADJP bit is set to 0 by an RTC software reset.

HR24 bit (Hours Mode)

This bit specifies whether the RTC operates in 12- or 24-hour mode.
Use the START bit to stop counting before changing the value of the HR24 bit. Do not stop counting (write 0 to the START bit) and change the value of the HR24 bit at the same time.

CNTMD bit (Count Mode Select)

This bit specifies whether the RTC count mode operates in calendar count mode or in binary count mode.
When setting the count mode, execute an RTC software reset and start again from the initial settings. This bit is updated synchronously with the count source, and its value is fixed before the RTC software reset is complete.
For details on initial settings, see section 16.3.1, Outline of Initial Settings of Registers after Power On.

(2) In binary count mode:

Address(es): RTC.RCR2 FCFF D024h [secure], FCFF 1024h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	CNTMD	—	AADJP	AADJE	—	—	RESET	START
Value after reset:	x	x	x	x	0	0	0	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b0	START	Start	0: The 32-bit binary counter, 64-Hz counter, and prescaler are stopped 1: The 32-bit binary counter, 64-Hz counter, and prescaler are in normal operation.	R/W
b1	RESET	RTC Software Reset	• In writing 0: Writing is invalid 1: The prescaler and the target registers for RTC software reset*1 are initialized • In reading 0: In normal time operation, or an RTC software reset has completed. 1: During an RTC software reset	R/W
b2, b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b4	AADJE	Automatic Adjustment Enable*2	0: Automatic adjustment is disabled 1: Automatic adjustment is enabled.	R/W
b5	AADJP	Automatic Adjustment Period Select*2	0: Adds or subtracts the RADJ.ADJ[5:0] bits from the prescaler count value every 32 seconds 1: Adds or subtracts the RADJ.ADJ[5:0] bits from the prescaler count value every 8 seconds	R/W
b6	—	Reserved	This bit is undefined. The write value should be 0.	R/W
b7	CNTMD	Count Mode Select	0: The calendar count mode 1: The binary count mode.	R/W

Note 1. R64CNT, RSECAR/BCNT0AR, RMINAR/BCNT1AR, RHRAR/BCNT2AR, RWKAR/BCNT3AR, RDAYAR/BCNT0AER, RMONAR/BCNT1AER, RYRAR/BCNT2AER, RYRAREN/BCNT3AER, RADJ, RCR2.ADJ30, RCR2.AADJE, RCR2.AADJP
Note 2. When EXTAL is selected, the setting of this bit is disabled.

START bit (Start)

This bit stops or restarts the prescaler or counter (clock) operation.
The START bit is updated in synchronization with the count source. When the START bit is modified, check that the bit is updated before proceeding.

RESET bit (RTC Software Reset)

This bit initializes the prescaler and registers to be reset by RTC software.
When 1 is written to the RESET bit, initialization starts in synchronization with the count source. When the initialization is completed, the RESET bit is automatically set to 0. When 1 is written to the RESET bit, check that the bit is set to 0 before proceeding.

AADJE bit (Automatic Adjustment Enable)

This bit controls (enables or disables) automatic adjustment.
Set the plus–minus bits (RADJ.PMADJ[1:0]) to 00b (adjustment is not performed) before changing the value of the AADJE bit. The AADJE bit is set to 0 by an RTC software reset.

AADJP bit (Automatic Adjustment Period Select)

This bit selects the automatic-adjustment period.
Correction period can be selected from 32 second units or 8 second units in binary count mode.
Set the plus–minus bits (RADJ.PMADJ[1:0]) to 00b (adjustment is not performed) before changing the value of the AADJP bit. The AADJP bit is set to 0 by an RTC software reset.

CNTMD bit (Count Mode Select)

This bit specifies whether the RTC count mode operates in calendar count mode or in binary count mode.
When setting the count mode, execute an RTC software reset and start again from the initial settings. This bit is updated synchronously with the count source, and its value is fixed before the RTC software reset is completed.
For details on initial settings, see section 16.3.1, Outline of Initial Settings of Registers after Power On.

16.2.20 RTC Control Register 3 (RCR3)

Address(es): RTC.RCR3 FCFF D026h [secure], FCFF 1026h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	RTCEN
Value after reset:	0	0	0	0	0	0	0	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b0	RTCEN	32kHz clock generator control bit	0: The 32-kHz clock oscillator is stopped. 1: The 32-kHz clock oscillator is operated.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R

RCR3 controls the 32-kHz clock oscillator in the clock generation circuit. For control of the 32-kHz clock oscillator, refer to section 6, Clock Pulse Generator.
This register has the same function in calendar count mode and binary count mode.
After rewriting this register, confirm that all bit values are updated and execute the next processing.

RTCEN Bit (32kHz Clock Generator Control Bit)

This register controls the operation and stop of the 32 kHz clock oscillator.
The operation and stop of the 32-kHz clock oscillator are controlled by RTCEN bits. When either bit is set in operation, the 32 kHz clock oscillator is in operation.
To use the 32-kHz clock as count source of RTC3, set the 32-kHz clock oscillator with RTCEN bit.

16.2.21 RTC Control Register 4 (RCR4)

Address(es): RTC.RCR4 FCFF D028h [secure], FCFF 1028h [non secure]

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	RCK SEL
Value after reset:	0	0	0	0	0	0	0	x

x: Undefined

Bit	Symbol	Bit name	Description	R/W
b0	RCKSEL	Count Source Select	0: 32-kHz clock is selected. 1: EXTAL clock is selected.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

RCR4 selects the count source and is used in both calendar count mode and binary count mode.
When the RCKSEL bit is set to 0, the time is counted with the 32-kHz clock. When the bit is set to 1, the time is counted with EXTAL clock.

RCKSEL bit (Count Source Select)

This bit selects the count source from the 32-kHz clock and EXTAL clock.
The count source must be selected only once before making the initial settings of the RTC registers at power on.

16.2.22 Frequency Register H/L (RFRH/L)

RFRH Register

Address(es): RTC.RFRH FCFF D02Ah [secure], FCFF 102Ah [non secure]

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RFC [16]
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x

x: Undefined

RFRL Register

Address(es): RTC.RFRL FCFF D02Ch [secure], FCFF 102Ch [non secure]

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RFC[15:0]															
Value after reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

RFRH Register

Bit	Symbol	Bit name	Description	R/W
b0	RFC[16]	Frequency Comparison Value	Set the frequency comparison value to generate a 128-Hz clock from the EXTAL frequency.	R/W
b15 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R

RFRL Register

Bit	Symbol	Bit name	Description	R/W
b15 to b0	RFC[15:0]	Frequency Comparison Value	Set the frequency comparison value to generate a 128-Hz clock from the EXTAL frequency.	R/W

RFRH and RFRL are registers for controlling the prescaler when EXTAL is selected. This register is a common feature in the calendar count mode and binary count mode.
The time counter of this module operates on a 128-Hz clock signal as the base clock. Therefore, when EXTAL is selected, EXTAL is divided by the prescaler to generate a 128-Hz clock signal. Clock mode 0, the EXTAL clock is divided only by the prescaler to generate a 128-Hz clock signal. Clock mode 1, the EXTAL clock is divided by twice, and divided by the prescaler to generate a 128-Hz clock signal. Set the frequency comparison value in the RFC[16:0] bits to generate a 128-Hz clock from the EXTAL frequency.
Change the frequency comparison value according to the EXTAL clock frequency. For calculation method, refer to the following. A value from 0000_0007h through 0001_FFFFh can be specified as the frequency comparison value. If a value outside of this range is specified, the RTC does not operate correctly.
Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2. The range of EXTAL clock frequency that can generate a 128-Hz clock is 1.024-kHz to 16.777-MHz in clock mode 0 and 2.048-kHz to 33.555-MHz in clock mode 1.
The operating frequency of the peripheral module clock and the EXTAL should be such that the peripheral module clock is greater than or equal to the EXTAL.

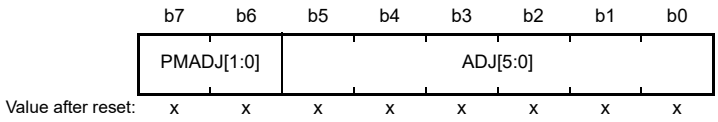
Calculation method of frequency comparison value:
Clock mode 0,
RFC[16:0] = (EXTAL clock frequency) / 128 - 1
Clock mode 1,
RFC[16:0] = (EXTAL clock frequency) / 256 - 1

Table 16.5 Setting Example

Clock Frequency			RFC Setting Value
EXTAL	Clock mode 0	10MHz	H'1312C
		12MHz	H'16E35
	Clock mode 1	20MHz	H'1312C
		24MHz	H'16E35

16.2.23 Time Error Adjustment Register (RADJ)

Address(es): RTC.RADJ FCFF D02Eh [secure], FCFF 102Eh [non secure]



x: Undefined

Bit	Symbol	Bit name	Description	R/W
b5 to b0	ADJ[5:0]	Adjustment Value	These bits specify the adjustment value from the prescaler	R/W
b7, b6	PMADJ[1:0]	Plus–Minus	b7 b6 0 0: Adjustment is not performed 0 1: Adjustment is performed by the addition to the prescaler 1 0: Adjustment is performed by the subtraction from the prescaler 1 1: Setting prohibited.	R/W

Adjustment is performed by the addition to or subtraction from the prescaler. If the automatic adjustment enable (RCR2.AADJE) bit is 0, adjustment is performed when writing to the RADJ. If the RCR2.AADJE bit is 1, adjustment is performed in the interval specified by the automatic adjustment period select (RCR2.AADJP) bit.
The current adjustment by software (disabling automatic adjustment) might be invalid if the following adjustment value is specified within 320 cycles of the count source after the register setting. To perform adjustment consecutively, wait for 320 cycles or more of the count source after the register setting, then specify the next adjustment value.
RADJ is updated in synchronization with the count source. When RADJ is modified, check that all the bits are updated before continuing with further processing. This register is set to 00h by an RTC software reset. The setting of this register is enabled only when the 32-kHz clock is selected. When EXTAL clock is selected, adjustment is not performed.

ADJ[5:0] bits (Adjustment Value)

These bits specify the adjustment value (the number of 32-kHz clock cycles) from the prescaler.

PMADJ[1:0] bits (Plus–Minus)

These bits select whether the clock is set ahead or back depending on the error-adjustment value set in the ADJ[5:0] bits.

16.3 Operation

16.3.1 Outline of Initial Settings of Registers after Power On

After the power is turned on, the initial settings for the clock setting, count mode setting, time error adjustment, time setting, alarm, interrupt, and time capture control register should be performed.

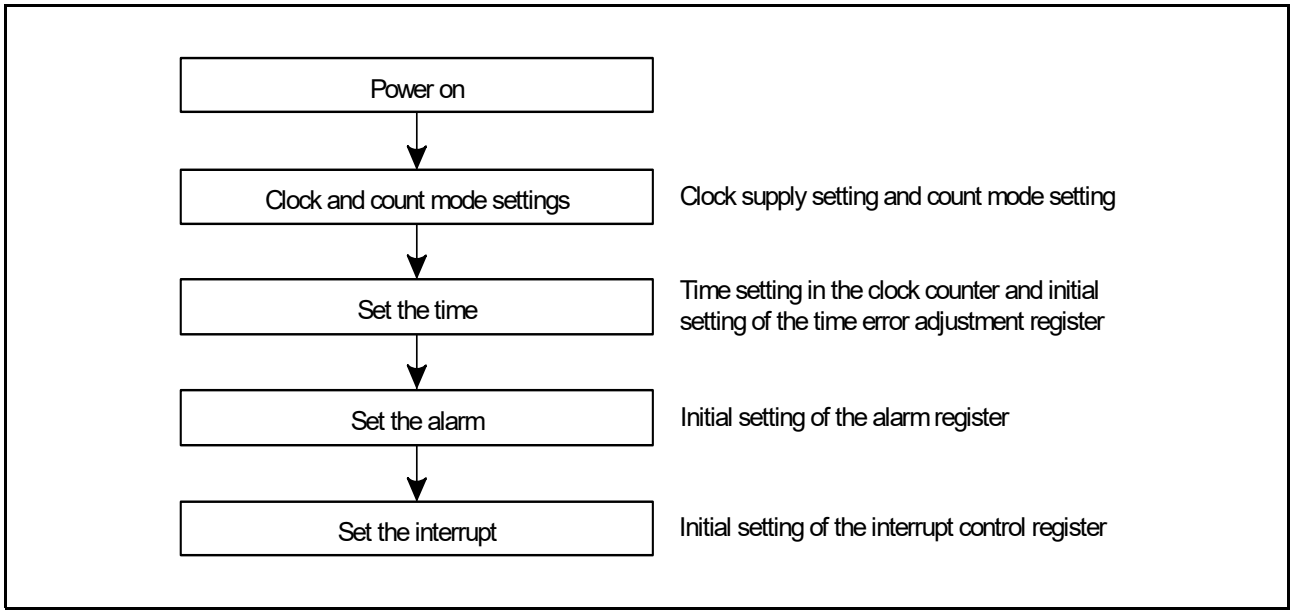
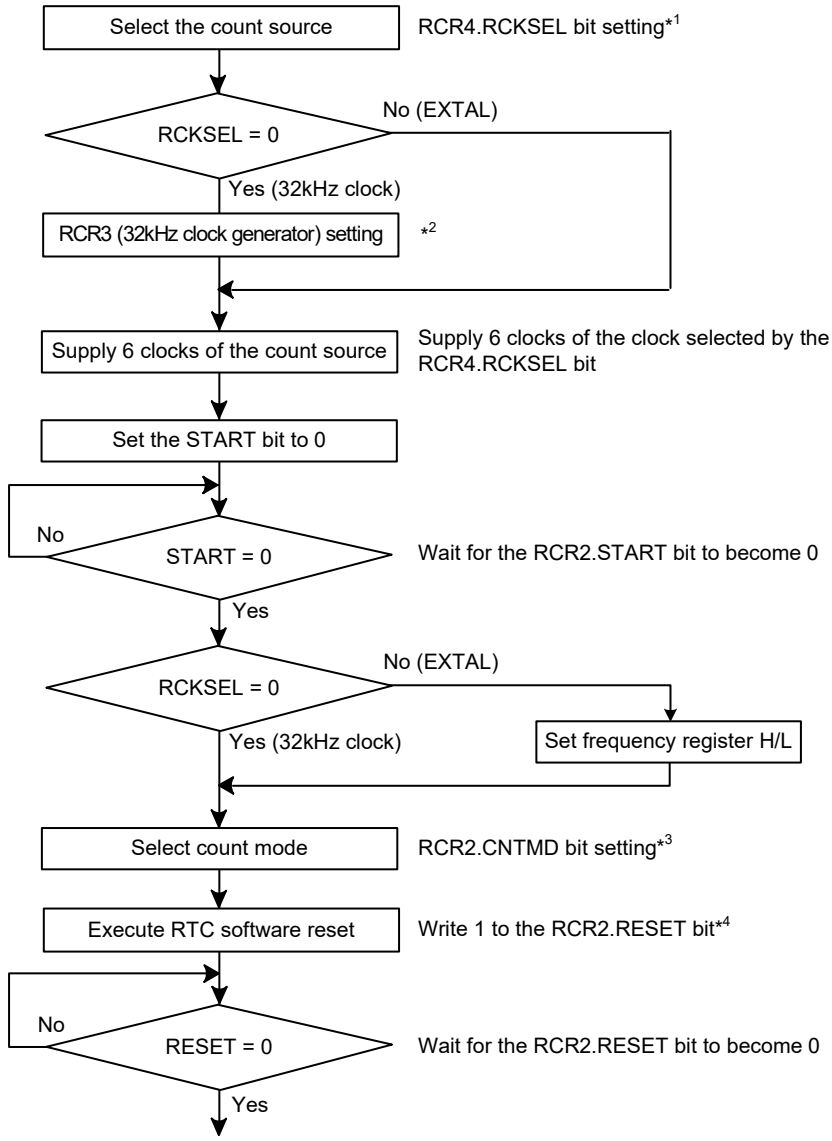


Figure 16.2 Outline of initial settings after a power on

16.3.2 Clock and Count Mode Setting Procedure

Figure 16.3 shows how to set the clock and the count mode.

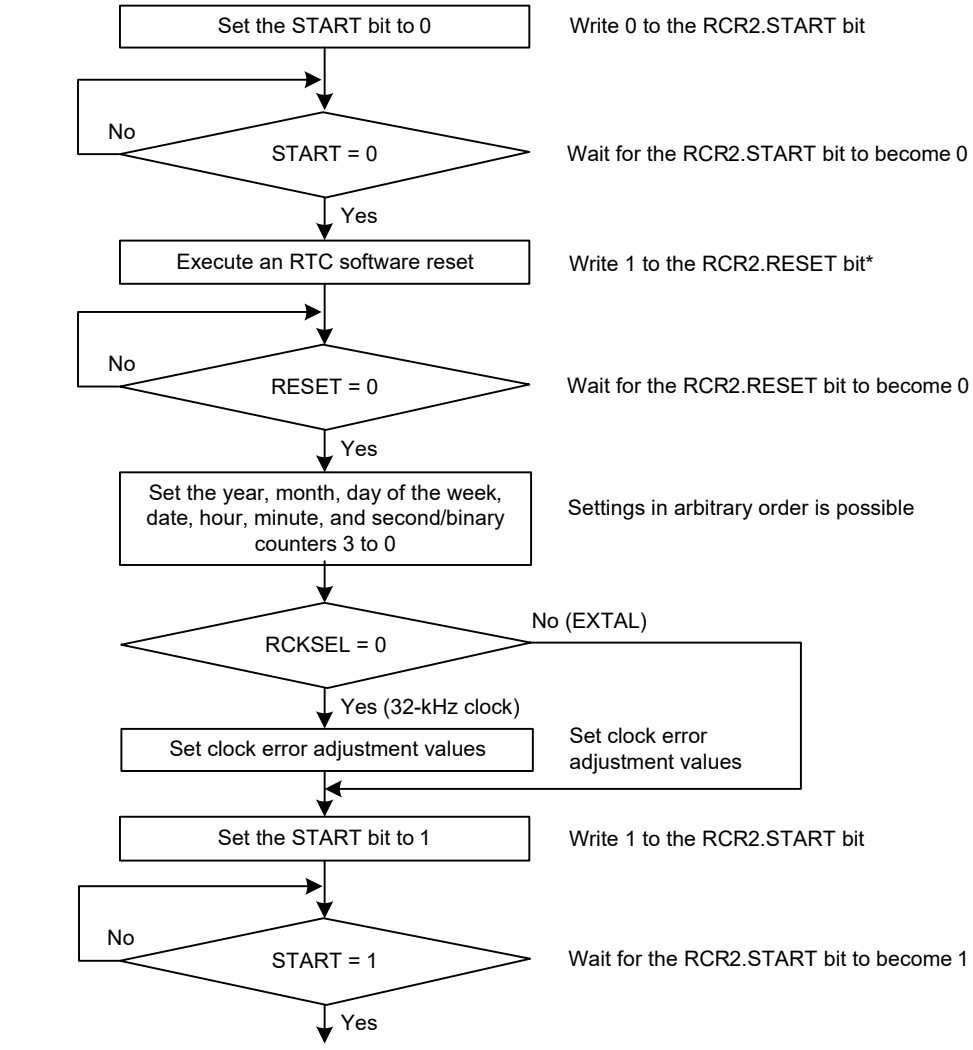


Note 1. This setting can only be made once after the power supply has been turned on.
Note 2. Waiting for the oscillation settling time (t_{ROSC}) is required after 1 has been written to RCR3.RTCEN. For the settling time, see the row with the tag “On-chip oscillation circuit oscillation settling time (RTC_X1)” in Table 56.6 of section 56, Electrical Characteristics.
Note 3. This step is not necessary if the count mode is set concurrently by setting the START bit to 0.
Note 4. A value associated with the count mode setting must be written to the RCR2.CNTMD bit.

Figure 16.3 Clock and count mode setting procedure

16.3.3 Setting the Time

Figure 16.4 shows how to set the time.



Note: This step is not necessary for the time-setting procedure because an RTC software reset is executed in the clock setting procedure of the initial settings for the power supply.

Figure 16.4 Setting the time

16.3.4 30-Second Adjustment

Figure 16.5 shows how to execute a 30-second adjustment.

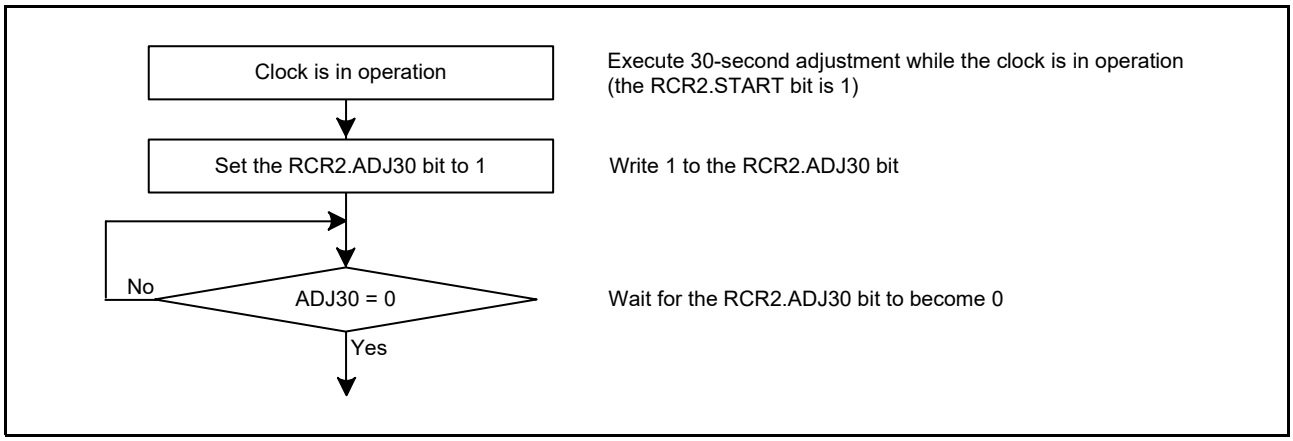


Figure 16.5 30-Second adjustment

16.3.5 Reading 64-Hz Counter and Time

Figure 16.6 shows how to read a 64-Hz counter and time.

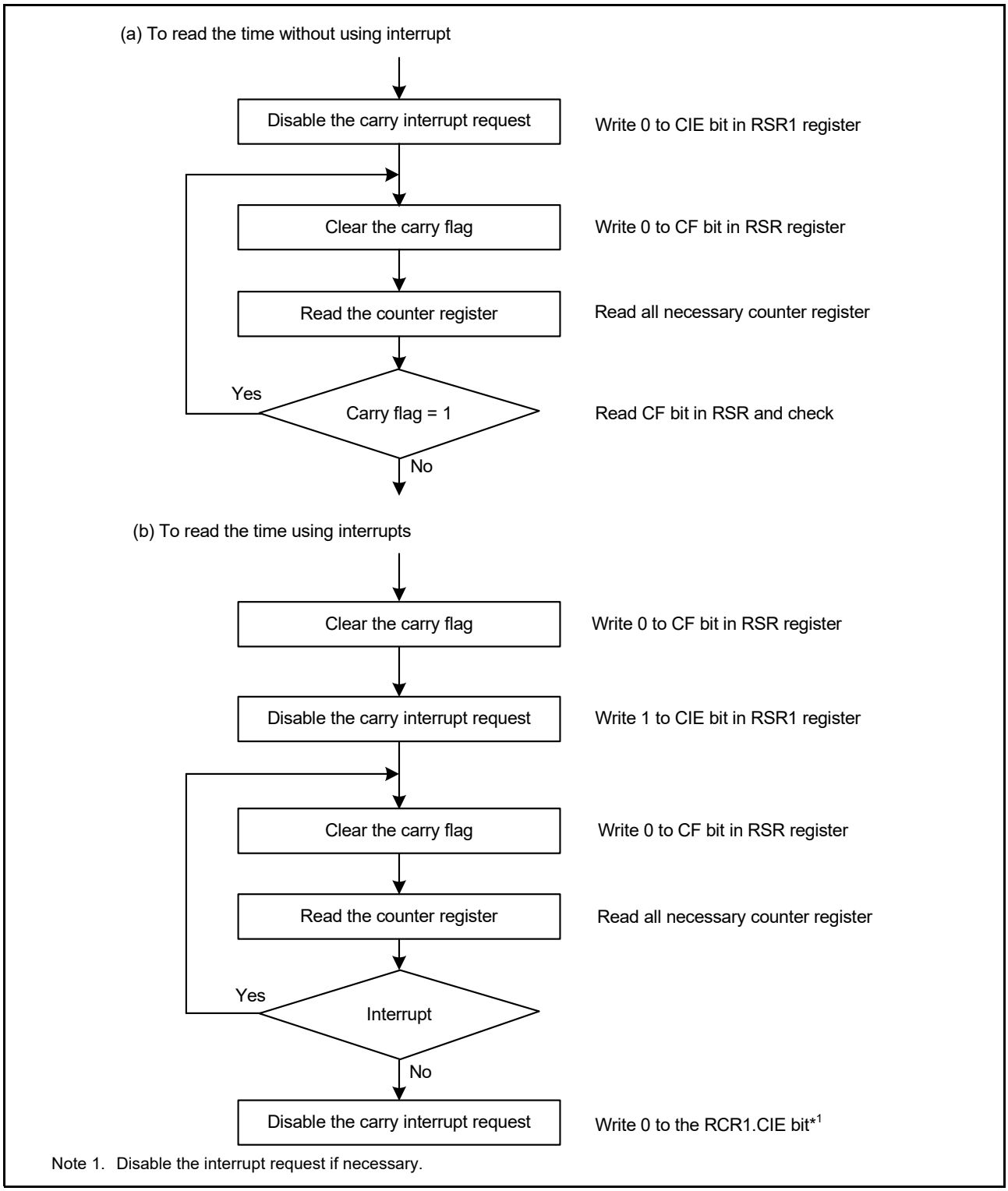


Figure 16.6 Reading time

If a carry occurs while the 64-Hz counter and time are read, the correct time is not obtained, therefore they must be read again. The procedure for reading the time without using interrupts is shown in (a) in Figure 16.6, and the procedure using carry interrupts is shown in (b). To keep the program simple, method (a) should be used in most cases.

16.3.6 Alarm Function

Figure 16.7 shows how to use the alarm function.

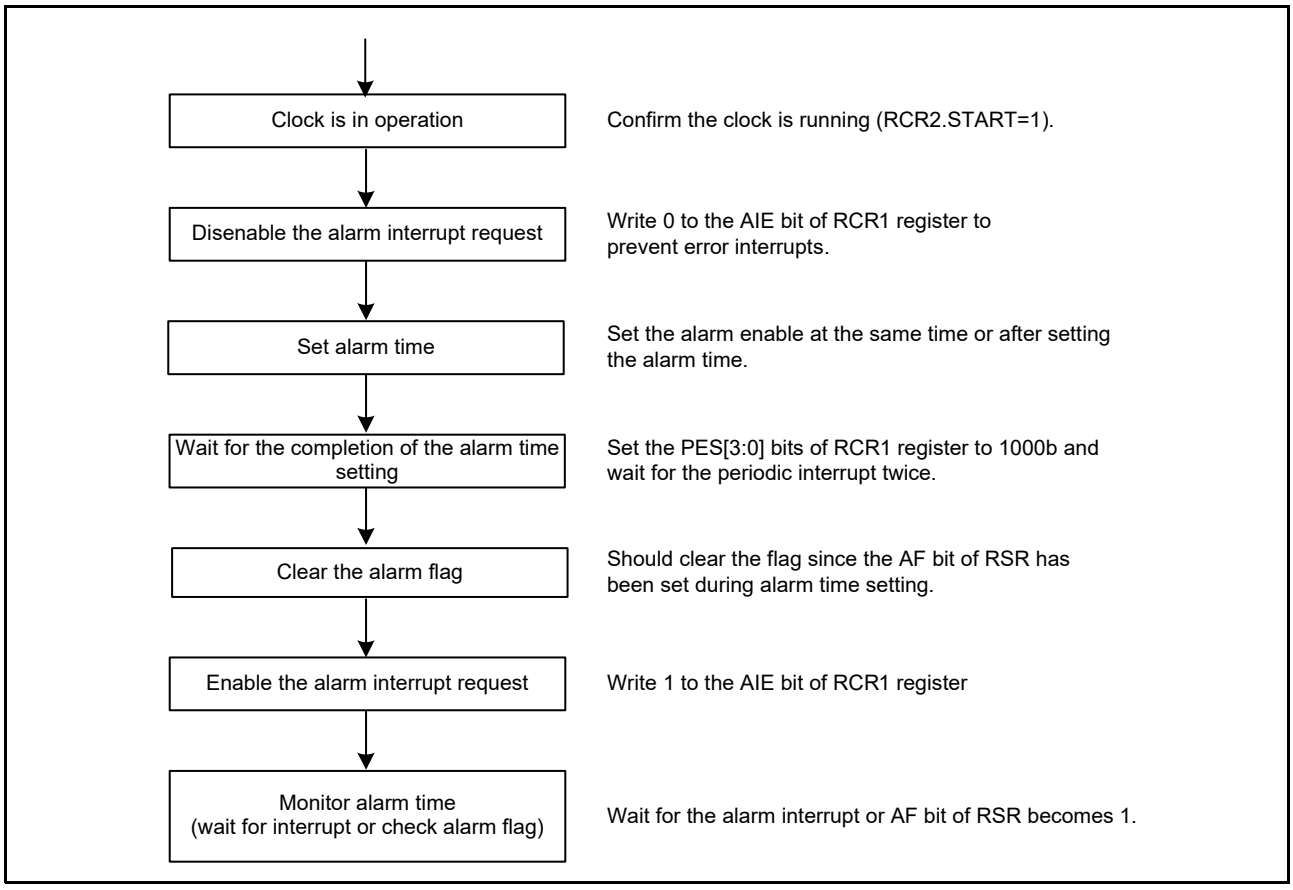


Figure 16.7 Using alarm function

In calendar count mode, an alarm can be generated by any one of year, month, date, day-of-week, hour, minute or second, or any combination of those. Write 1 to the ENB bit in the alarm registers involved in the alarm setting, and set the alarm time in the lower bits. Write 0 to the ENB bit in registers not involved in the alarm setting.

In binary count mode, an alarm can be generated in any bit combination of 32 bits. Write 1 to the ENB bit of the alarm enable register associated with the target bit of the alarm, and set the alarm time in the alarm register. For bits that are not the target of the alarm, write 0 to the ENB bit of the alarm enable register.

When the counter and the alarm time match, the alarm flag of RSR is set to 1. Alarm detection can be confirmed by reading the Interrupt Set-Pending Register corresponding to the RTC_ALM interrupt, but an interrupt should be used in most cases. If 1 is set in the AIE bit of RCR1 register, an alarm interrupt is generated in the event of the alarm, enabling the alarm to be detected.

The alarm flag of RSR is cleared by writing 0.

When the counter and the alarm time match in a low power consumption state, the MCU returns from the low power consumption state.

16.3.7 Procedure for Disabling Alarm Interrupt

Figure 16.8 shows the procedure for disabling the enabled alarm interrupt request.

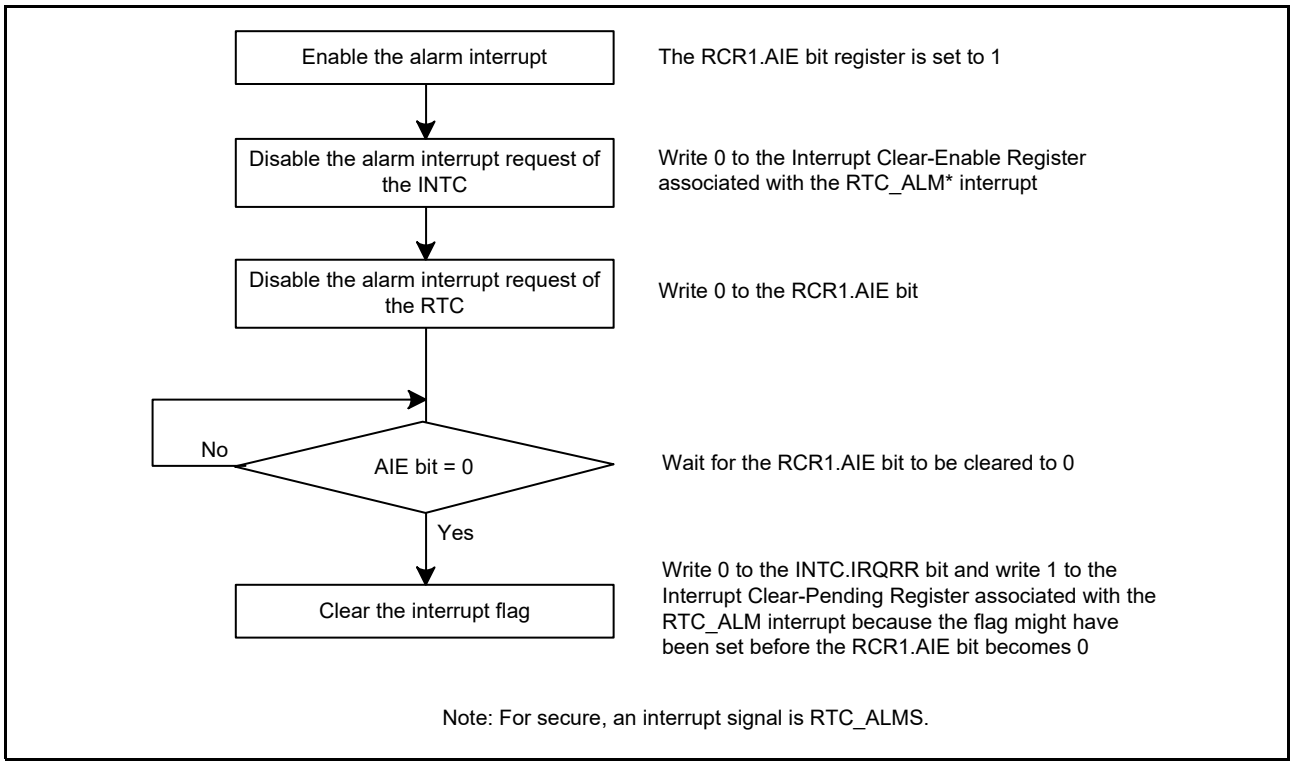


Figure 16.8 Procedure for disabling alarm interrupt request

16.3.8 Time Error Adjustment Function

The time error adjustment function is used to correct errors, running fast or slow, in the time due to variation in the precision of oscillation by the 32-kHz clock. Because 32,768 cycles of the 32-kHz clock constitute 1 second of operation when the 32-kHz clock is selected, the clock runs fast if the 32-kHz clock frequency is high and slow if the 32-kHz clock frequency is low. There are two types of time error adjustment functions:

- Automatic Adjustment
- Adjustment by Software.

Use the RCR2.AADJE bit to select automatic adjustment or adjustment by software.

16.3.8.1 Automatic Adjustment

Enable automatic adjustment by setting the RCR2.AADJE bit to 1. Automatic adjustment is the addition or subtraction of the value counted by the prescaler to or from the value in RADJ every time the adjustment period selected by the RCR2.AADJE bit elapses.

(1) [Example 1] 32-kHz clock running at 32.769 kHz

(a) Adjustment procedure

When the 32-kHz clock is running at 32.769 kHz, 1 second elapses every 32,769 clock cycles. The RTC is meant to run at 32,768 clock cycles, so the clock runs fast by one clock cycle every second. The time on the clock is fast by 60 clock cycles per minute, so adjustment can take the form of setting the clock back by 60 cycles every minute.
Register settings: (when RCR2.CNTMD = 0)

- RCR2.AADJP = 0 (adjustment every minute)
- RADJ.PMADJ[1:0] = 10b (adjustment is performed by the subtraction from the prescaler)
- RADJ.ADJ[5:0] = 60 (3Ch).

(2) [Example 2] 32-kHz clock running at 32.766 kHz

(a) Adjustment procedure

When the 32-kHz clock is running at 32.766 kHz, 1 second elapses every 32,766 clock cycles. The RTC is meant to run at 32,768 clock cycles, so the clock runs slow by two clock cycles every second. The time on the clock is slow by 20 clock cycles every 10 seconds, so adjustment can take the form of setting the clock forward by 20 cycles every 10 seconds.
Register settings: (when RCR2.CNTMD = 0)

- RCR2.AADJP = 1 (adjustment every 10 seconds)
- RADJ.PMADJ[1:0] = 01b (adjustment is performed by the addition to the prescaler.)
- RADJ.ADJ[5:0] = 20 (14h).

(3) [Example 3] 32-kHz clock running at 32.764 kHz

(a) Adjustment procedure

When the 32-kHz clock is running at 32.764 kHz, 1 second elapses every 32,764 clock cycles. The RTC is meant to run at 32,768 clock cycles, so the clock runs slow by four clock cycles every second. The time on the clock is slow by 32 clock cycles every 8 seconds, so adjustment can take the form of setting the clock forward by 32 cycles every 8 seconds.

Register settings when the RCR2.CNTMD bit is 1

- RCR2.AADJP = 1 (adjustment every 8 seconds)
- RADJ.PMADJ[1:0] = 01b (adjustment is performed by the addition to the prescaler)
- RADJ.ADJ[5:0] = 32 (20h).

16.3.8.2 Adjustment by Software

Enable adjustment by software by setting the RCR2.AADJE bit to 0. Adjustment by software is the addition or subtraction of the value counted by the prescaler to or from the value in the RADJ at the time of execution of a write instruction to the RADJ.

(1) [Example 1] 32-kHz clock running at 32.769 kHz

(a) Adjustment procedure

When the 32-kHz clock is running at 32.769 kHz, 1 second elapses every 32,769 clock cycles. The RTC is meant to run at 32,768 clock cycles, so the clock runs fast by one clock cycle every second. The time on the clock is fast by one clock cycle per second, so adjustment can take the form of setting the clock back by one cycle every second.

(b) Register settings

- RADJ.PMADJ[1:0] = 10b (adjustment is performed by the subtraction from the prescaler)
- RADJ.ADJ[5:0] = 1 (01h)
  This is written to the RADJ register once per 1-second interrupt.

16.3.8.3 Procedure for Changing the Mode of Adjustment

When changing the mode of adjustment, change the value of the AADJE bit in RCR2 after setting the RADJ.PMADJ[1:0] bits to 00b (adjustment is not performed).

To change adjustment by software to automatic adjustment:

1. Set the RADJ.PMADJ[1:0] bits to 00b (adjustment is not performed).
2. Set the RCR2.AADJE bit to 1 (automatic adjustment is enabled).
3. Use the RCR2.AADJP bit to select the period of adjustment.
4. In RADJ, set the PMADJ[1:0] bits for addition or subtraction and the ADJ[5:0] bits to the value for use in time error adjustment.

To change automatic adjustment to adjustment by software:

1. Set the RADJ.PMADJ[1:0] bits to 00b (adjustment is not performed).
2. Set the RCR2.AADJE bit to 0 (adjustment by software is enabled).
3. Proceed with the adjustment by setting the RADJ.PMADJ[1:0] bits for addition or subtraction and the RADJ.ADJ[5:0] bits to the value for use in time error adjustment at the desired time. After that, the time is adjusted every time a value is written to the RADJ register.

16.3.8.4 Procedure for Stopping Adjustment

Stop the adjustment by setting the RADJ.PMADJ[1:0] bits to 00b (adjustment is not performed).

16.4 Interrupt Sources

The RTC has three interrupt sources and are listed in Table 16.6.

Table 16.6 RTC Interrupt sources

Name	Interrupt sources
RTC_ALM	Alarm interrupt
RTC_PRD	Periodic interrupt
RTC_CUP	Carry interrupt

Note: For secure, interrupt sources are RTC_ALMS, RTC_PRDS and RTC_CUPS, respectively.

(1) Alarm interrupt (RTC_ALM, RTC_ALMS)

This interrupt is generated according to the result of comparison between the alarm registers and realtime clock counters. For details, see section 16.3.6, Alarm Function.

Because there is a possibility that the interrupt flag might be set to 1 when the settings of the alarm registers match the clock counters, wait for the alarm time settings to be confirmed and clear the INTC.IRQRR and the interrupt Set-Pending Register corresponding to the RTC_ALM interrupt to 0 again after modifying values of the alarm registers. After the interrupt flag for the alarm interrupt is set to 1 and the state is returned to non-matching of the alarm registers and clock counters, the flag does not set again until there is a further match or the values of the alarm registers are modified again.

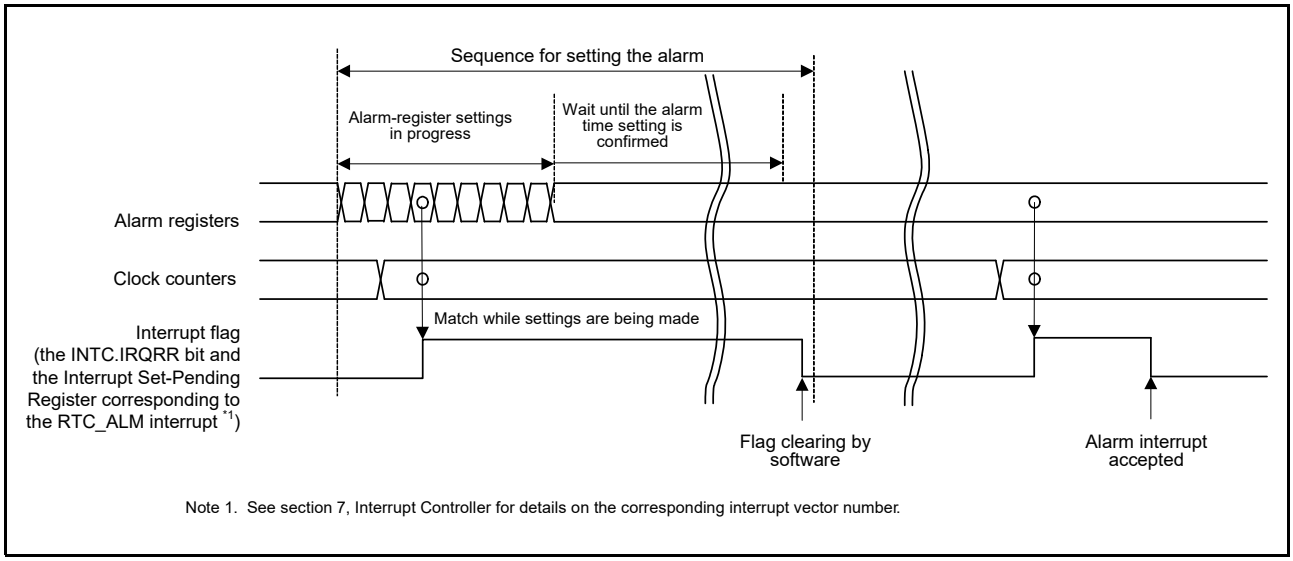


Figure 16.9 Timing chart for the alarm interrupt (RTC_ALM, RTC_ALMS)

(2) Periodic interrupt (RTC_PRD, RTC_PRDS)

This interrupt is generated at intervals of 2 seconds, 1 second, 1/2 second, 1/4 second, 1/8 second, 1/16 second, 1/32 second, 1/64 second, 1/128 second, or 1/256 second. The interrupt interval can be selected through the RCR1.PES[3:0] bits.

(3) Carry interrupt (RTC_CUP, RTC_CUPS)

This interrupt is generated when a carry to the second counter/binary counter 0 occurred or a carry to the R64CNT counter occurred during read access to the 64-Hz counter.

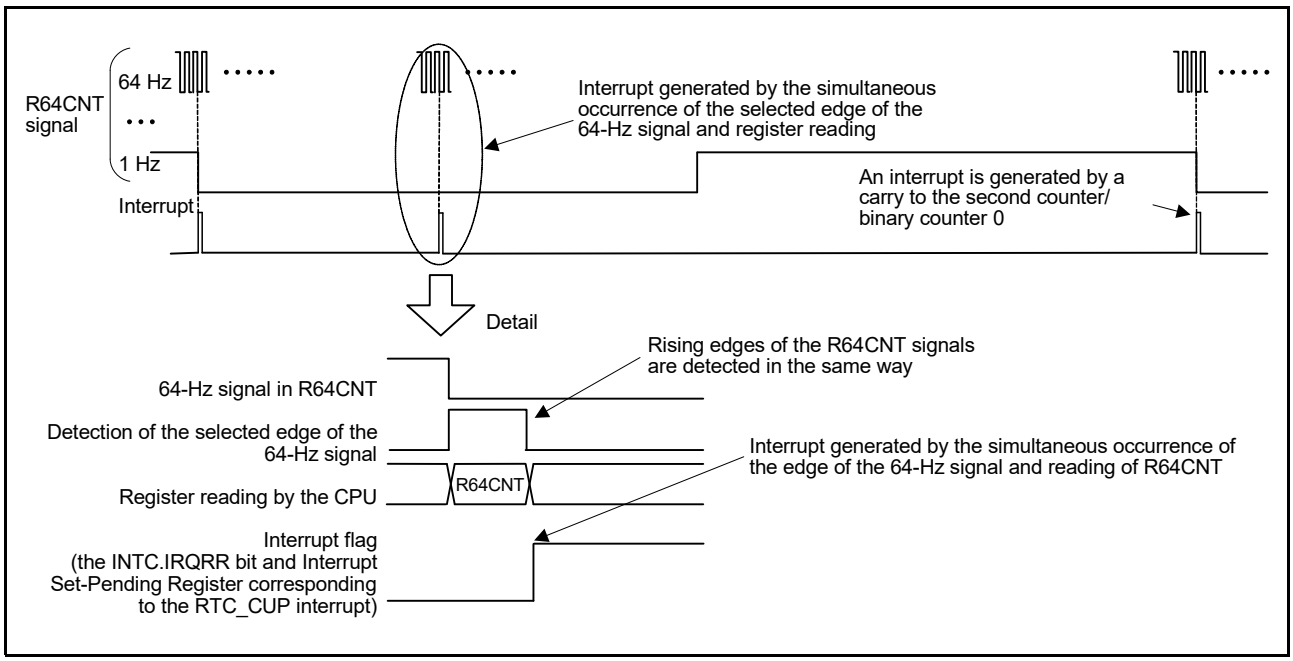


Figure 16.10 Timing chart for the Carry interrupt (RTC_CUP, RTC_CUPS)

16.5 Usage Notes

16.5.1 Register Writing during Counting

The following registers must not be written to during counting, that is, while the RCR2.START bit = 1.

- RSECCNT/BCNT0
- RMINCNT/BCNT1
- RHRCNT/BCNT2
- RDAYCNT
- RWKCNT/BCNT3
- RMONCNT
- RYRCNT
- RCR2.HR24
- RFRH, RFRL

The counter must be stopped before writing to any of the these registers.

16.5.2 Use of Periodic Interrupts

The procedure for using periodic interrupts is shown in Figure 16.11.
The generation and period of the periodic interrupt can be changed by setting the RCR1.PES[3:0] bits. However, because the prescaler R64CNT and RSECCNT/BCNT0 are used to generate interrupts, the interrupt period is not guaranteed immediately after setting the RCR1.PES[3:0] bits. In addition, any of the following can affect the interrupt period:

- Stopping/restarting or resetting counter operation
- Reset by RTC software
- 30-second adjustment by changing the RCR2 value.

When the time error adjustment function is used, the interrupt generation period after adjustment is added or subtracted according to the adjustment value.

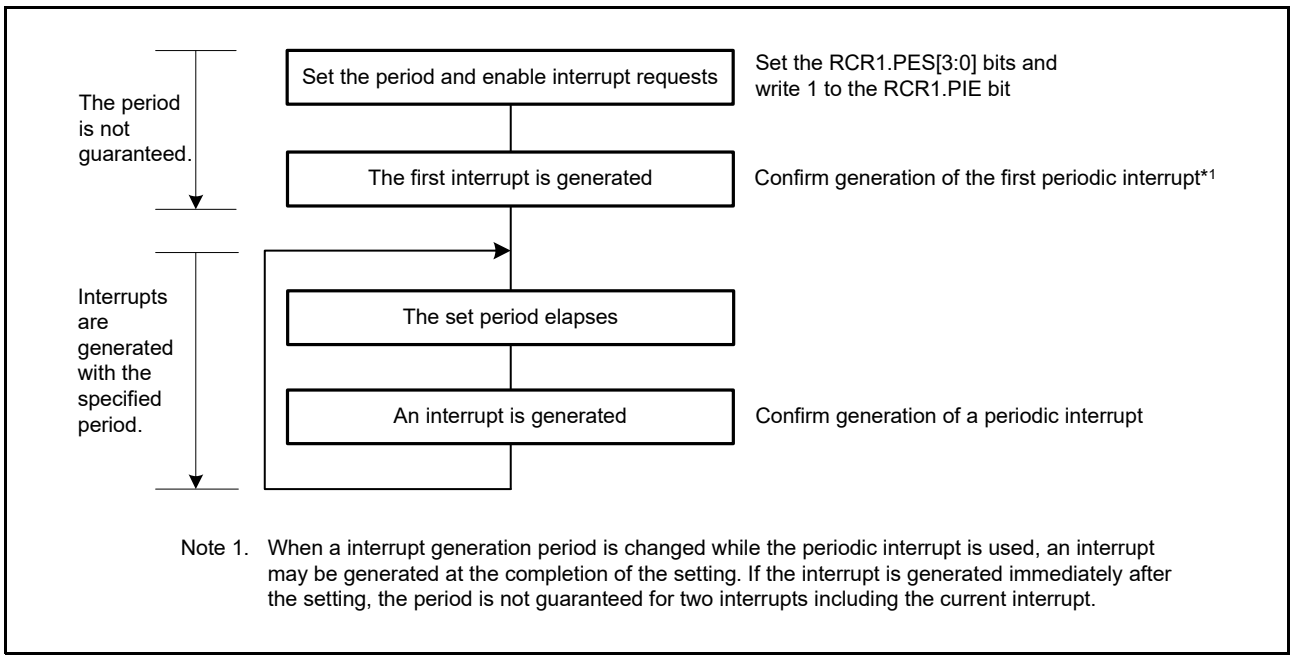


Figure 16.11 Using periodic interrupt function

16.5.3 Transitions to Low Power Consumption Modes after Setting Registers

A transition to a low power consumption state (Software Standby mode or Deep Standby mode) during writing to an RTC register might destroy the value of the register. After setting the register, confirm that the setting is in place before initiating a transition to a low power consumption state.

16.5.4 Notes When Writing to and Reading from Registers

- When reading a counter register such as the second counter after having written to the counter register, follow the procedure in section 16.3.5, Reading 64-Hz Counter and Time
- The value written to the count registers, alarm registers, year alarm enable register, bits RCR2.AADJE, AADJP, and HR24, RCR4 register, or frequency register is reflected when four read operations are performed after writing
- The values written to the RCR1.CIE bits can be read immediately after writing
- To read the value from the timer counter after return from a reset, period in Software Standby mode, or the battery backup state, wait for 1/128 second while the clock is operating (RCR2.START bit = 1)
- After a reset is generated, write to the RTC register after six cycles of the count source clock have elapsed.

16.5.5 Changing the Count Mode

When changing the count mode (calendar/binary), set the RCR2.START bit to 0, stop the counting operation, then restart it from the initial setting. For details on the initial setting, see section 16.3.1, Outline of Initial Settings of Registers after Power On.

16.5.6 Procedure when the Realtime Clock is not to be Used

When the realtime clock is not in use, stop the clock supply to the RTC as described in section 52.4.2, Usage Notes when the Realtime Clocks are Not to be Used.

17. Serial Communications Interface with FIFO (SCIFA)

This LSI has five channels of serial communication interface (SCIFA) with FIFO that support both asynchronous and clock synchronous serial communication. The SCIFA has 16-stage FIFO buffers for transmission and reception, respectively, for each channel that enable this LSI to perform efficient high-speed continuous communication.

17.1 Overview

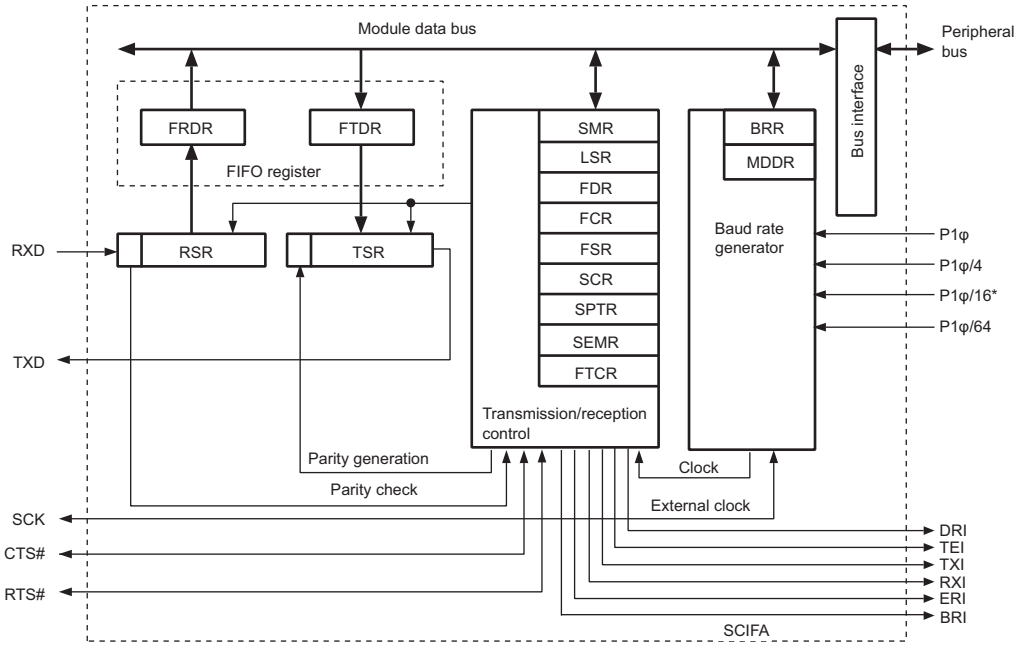
Table 17.1 lists the specifications of the SCIFA.

Table 17.1 Specifications of SCIFA

Item		Description
Channel		5 channels
Serial communication method		Asynchronous communication mode and clock synchronous communication mode
Transfer speed		Selectable bit rate with an on-chip baud rate generator
Full duplex communication		Transmitting section: realizes continuous data transmission using 16-stage FIFO buffer Receiving section: realizes continuous data reception using 16-stage FIFO buffer
Data transmission		Selectable either LSB-first or MSB-first transfer
Interrupt source		The following six sources: • Transmit-end (TEIF) • Transmit-FIFO-data-empty (TXIF) • Receive-FIFO-data-full (RXIF) • Receive-data-ready (DRIF)*1 • Receive-error (ERIF) • Break detection or overrun (BRIF)
Asynchronous communication mode	Character length	7 or 8 bits
	Transmission stop bit length	1 or 2 bits
	Parity	Even, odd, or none
	Receive error detection	Parity, overrun, and framing errors
	Hardware flow control	Controls data transmission and reception using the CTS# and RTS# pins. Not applicable to channel 3 (SCIF3) and channel 4 (SCIF4)
	Break detection	Break signal detection function by hardware.
	Clock source	Selectable from internal or external clock
	Noise cancellation	Incorporates a digital noise filter in the RXD pin input path.
Clock synchronous communication mode	Character length	8 bits
	Receive error detection	Detects an overrun error as a receive error.
	Clock source	Selectable either internal or external clock
Bit rate modulation		Enables errors to be decreased by correcting the output of the on-chip baud rate generator.

Note 1. Effective only for asynchronous communication mode

Figure 17.1 shows a block diagram of the SCIFA.



RSR:	Receive shift register	BRR:	Bit rate register
FRDR:	Receive FIFO data register	SPTR:	Serial port register
TSR:	Transmit shift register	FCR:	FIFO control register
FTDR:	Transmit FIFO data register	FDR:	FIFO data count register
SMR:	Serial mode register	LSR:	Line status register
SCR:	Serial control register	SEMR:	Serial extended mode register
FSR:	Serial status register	FTCR:	FIFO trigger control register
MDDR:	Modulation duty register		

Note: * The P1φ/16 clock is not input in the case of the channel 0 interface.

Figure 17.1 Block Diagram of SCIFA

Table 17.2 lists the input/output pins of the SCIFA.

Table 17.2 Pin Configuration of the SCIFA

Channel	Item	Pin Name	I/O	Function
0 to 4	Serial clock pin	SCK0 to SCK4	I/O	Transmission/reception clock input/output, general output
	Receive data pin	RxD0 to RxD4	Input	Receive data input
	Transmit data pin	TxD0 to TxD4	Output	Transmit data output
0 to 2	Transmission/reception start control pin	CTS#0 to CTS#2	I/O	Input for hardware flow control (transmission enable signal) / general output
		RTS#0 to RTS#2	Output	Output for hardware flow control (transmission request signal) / general output

Note: Channels of each pin is omitted.

17.2 Register Descriptions

Table 17.3 List of Registers

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	Serial mode register	SMR	R/W	0000h	H'E8007000	16
	Bit rate register	BRR	R/W	FFh	H'E8007002	8
	Modulation duty register	MDDR	R/W	FFh	H'E8007002	8
	Serial control register	SCR	R/W	0000h	H'E8007004	16
	Transmit FIFO data register	FTDR	W	Undefined	H'E8007006	8
	Serial status register	FSR	R/W	0020h	H'E8007008	16
	Receive FIFO data register	FRDR	R	Undefined	H'E800700A	8
	FIFO control register	FCR	R/W	0000h	H'E800700C	16
	FIFO data count register	FDR	R	0000h	H'E800700E	16
	Serial port register	SPTR	R/W	00xxh	H'E8007010	16
	Line status register	LSR	R/W	0000h	H'E8007012	16
	Serial extended mode register	SEMR	R/W	00h	H'E8007014	8
	FIFO trigger control register	FTCR	R/W	1F1Fh	H'E8007016	16
1	Serial mode register	SMR	R/W	0000h	H'E8007800	16
	Bit rate register	BRR	R/W	FFh	H'E8007802	8
	Modulation duty register	MDDR	R/W	FFh	H'E8007802	8
	Serial control register	SCR	R/W	0000h	H'E8007804	16
	Transmit FIFO data register	FTDR	W	Undefined	H'E8007806	8
	Serial status register	FSR	R/W	0020h	H'E8007808	16
	Receive FIFO data register	FRDR	R	Undefined	H'E800780A	8
	FIFO control register	FCR	R/W	0000h	H'E800780C	16
	FIFO data count register	FDR	R	0000h	H'E800780E	16
	Serial port register	SPTR	R/W	00xxh	H'E8007810	16
	Line status register	LSR	R/W	0000h	H'E8007812	16
	Serial extended mode register	SEMR	R/W	00h	H'E8007814	8
	FIFO trigger control register	FTCR	R/W	1F1Fh	H'E8007816	16
2	Serial mode register	SMR	R/W	0000h	H'E8008000	16
	Bit rate register	BRR	R/W	FFh	H'E8008002	8
	Modulation duty register	MDDR	R/W	FFh	H'E8008002	8
	Serial control register	SCR	R/W	0000h	H'E8008004	16
	Transmit FIFO data register	FTDR	W	Undefined	H'E8008006	8
	Serial status register	FSR	R/W	0020h	H'E8008008	16
	Receive FIFO data register	FRDR	R	Undefined	H'E800800A	8
	FIFO control register	FCR	R/W	0000h	H'E800800C	16
	FIFO data count register	FDR	R	0000h	H'E800800E	16
	Serial port register	SPTR	R/W	00xxh	H'E8008010	16
	Line status register	LSR	R/W	0000h	H'E8008012	16
	Serial extended mode register	SEMR	R/W	00h	H'E8008014	8
	FIFO trigger control register	FTCR	R/W	1F1Fh	H'E8008016	16

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
3	Serial mode register	SMR	R/W	0000h	H'E8008800	16
	Bit rate register	BRR	R/W	FFh	H'E8008802	8
	Modulation duty register	MDDR	R/W	FFh	H'E8008802	8
	Serial control register	SCR	R/W	0000h	H'E8008804	16
	Transmit FIFO data register	FTDR	W	Undefined	H'E8008806	8
	Serial status register	FSR	R/W	0020h	H'E8008808	16
	Receive FIFO data register	FRDR	R	Undefined	H'E800880A	8
	FIFO control register	FCR	R/W	0000h	H'E800880C	16
	FIFO data count register	FDR	R	0000h	H'E800880E	16
	Serial port register	SPTR	R/W	00xxh	H'E8008810	16
	Line status register	LSR	R/W	0000h	H'E8008812	16
	Serial extended mode register	SEMR	R/W	00h	H'E8008814	8
	FIFO trigger control register	FTCR	R/W	1F1Fh	H'E8008816	16
4	Serial mode register	SMR	R/W	0000h	H'E8009000	16
	Bit rate register	BRR	R/W	FFh	H'E8009002	8
	Modulation duty register	MDDR	R/W	FFh	H'E8009002	8
	Serial control register	SCR	R/W	0000h	H'E8009004	16
	Transmit FIFO data register	FTDR	W	Undefined	H'E8009006	8
	Serial status register	FSR	R/W	0020h	H'E8009008	16
	Receive FIFO data register	FRDR	R	Undefined	H'E800900A	8
	FIFO control register	FCR	R/W	0000h	H'E800900C	16
	FIFO data count register	FDR	R	0000h	H'E800900E	16
	Serial port register	SPTR	R/W	00xxh	H'E8009010	16
	Line status register	LSR	R/W	0000h	H'E8009012	16
	Serial extended mode register	SEMR	R/W	00h	H'E8009014	8
	FIFO trigger control register	FTCR	R/W	1F1Fh	H'E8009016	16

Note: BRR and MDDR are located in the same address. Setting the MDDRS bit of the SEMR register switches these registers.

17.2.1 Receive Shift Register (RSR)

The RSR register receives serial data and temporally stores the data. The SCIFA stores the serial data input via the RXD pin into the RSR register and converts the data to the parallel form. When one byte of data has been received, it is automatically transferred to the receive FIFO data register (FRDR).
The CPU cannot read from or write to the RSR register directly.

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0

17.2.2 Receive FIFO Data Register (FRDR)

The FRDR register is a 8-bit, 16-stage FIFO register that stores the received serial data. When the SCIFA receives one byte of serial data, it transfers the received data from the receive shift register (RSR) to the FRDR register and completes the receive operation. Continuous reception is possible until the received 16 bytes of data are stored. If the FRDR register is read when there is no received data in the FRDR register, an undefined value is read.
When the FRDR register is full of received data, subsequently received serial data is lost.

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0

17.2.3 Transmit Shift Register (TSR)

The SCIFA transfers the transmit data from the transmit FIFO data register (FTDR) to the TSR register, and then transmits the data serially to the TXD pin. After transmitting one byte of data, the SCIFA automatically transfers the next transmit data from the FTDR register into the TSR register and starts transmission.
The CPU cannot read from or write to the TSR register directly.

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

17.2.4 Transmit FIFO Data Register (FTDR)

The FTDR register is a 8-bit, 16-stage FIFO register that stores serial transmission data. When the SCIFA detects that the transmit shift register (TSR) is empty, it transmits data written in the FTDR register to the TSR register and starts serial transmission. Continuous serial transmission is executed until there is no transmit data left in the FTDR register. Writing the transmit data to the FTDR register should be done when a transmit data empty interrupt (TXI) request is generated. When the FTDR register becomes full of transmit data (16 bytes), no more data can be written. Even if new data is written, the data is ignored.
CPU can read from the FTDR register but cannot write to it.

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

17.2.5 Serial Mode Register (SMR)

The SMR register specifies the SCIFA serial communication format and selects the clock source for the baud rate generator.
The CPU can always read and write to the SMR register.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CM	CHR	PE	PM	STOP	—	CKS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	b1 b0 0 0: 1 × P1φ*1 0 1: 1/4 × P1φ*1 1 0: 1/16 × P1φ*1 *2 1 1: 1/64 × P1φ*1	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	STOP	Stop Bit Length	0: One stop bit 1: Two stop bits	R/W
b4	PM	Parity Mode	0: Even parity 1: Odd parity	R/W
b5	PE	Parity Enable	0: Parity bit addition or check is disabled. 1: Parity bit addition or check is enabled.	R/W
b6	CHR	Character Length	0: 8-bit data 1: 7-bit data*3	R/W
b7	CM	Communication Mode	0: Asynchronous mode 1: Clock synchronous mode	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. P1φ: Peripheral clock
Note 2. This setting is not applicable to channel 0. It is only applicable to channels 1 to 4.
Note 3. When 7-bit data is selected, the MSB (bit 7) of the transmit FIFO data register is not transmitted.

CKS[1:0] Bits (Clock Select)

Select an internal clock source for the on-chip baud rate generator. For further information on the clock source, bit rate register settings, and baud rates, see section 17.2.8, Bit Rate Register (BRR).

STOP Bit (Stop Bit Length)

Selects one bit or two bits as the stop bit length in asynchronous mode. This setting is used only in asynchronous mode. It is ignored in clock synchronous mode because no stop bits are added. When receiving, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit, but if the second stop bit is 0, it is treated as the start bit of the next incoming character.

Note: When transmitting with one stop bit, a single 1 bit (stop bit) is added at the end of each transmission character.
Note: When transmitting with two stop bits, two 1 bits (stop bits) are added at the end of each transmission character.

PM Bit (Parity Mode)

Selects either the even or odd parity check. The setting of this bit is effective only when the parity enable (PE) bit of this register is set to 1 in asynchronous mode. The setting of this bit is ignored in clock synchronous mode, or when parity addition/check is disabled in asynchronous mode.

Note: If even parity is selected, the parity bit is added to data to be transmitted to make the total number of 1s even in the transmission character and parity bit combined. When receiving, the SCIFA verifies that the total number of 1s in the received character and parity bit combined is even.
Note: If odd parity is selected, the parity bit is added to data to be transmitted to make the total number of 1s odd in the transmission character and parity bit combined. When receiving, the SCIFA verifies that the total number of 1s in the received character and parity bit combined is odd.

PE Bit (Parity Enable)

Selects whether to add a parity bit on data transmission and whether to enable/disable the parity check on data reception in asynchronous mode. In clock synchronous mode, a parity bit is neither added nor checked, regardless of the setting of this bit.

Note: When this bit is set to 1, an even or odd parity bit specified in the PM bit is added to data to be transmitted. The SCIFA verifies whether the parity bit of the received data is even or odd as specified in the PM bit when receiving.

CHR Bit (Character Length)

Selects 7- or 8-bit data length in asynchronous mode. In clock synchronous mode, the data length is always 8 bits, regardless of the CHR setting.

CM Bit (Communication Mode)

Selects whether the SCIFA operates in asynchronous or clock synchronous mode.

17.2.6 Serial Control Register (SCR)

The SCR register enables or disables the SCIFA transmission/reception and interrupt requests, and selects the transmit/receive clock source. The CPU can always read from and write to the SCR register.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	TIE	RIE	TE	RE	REIE	TEIE	CKE[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKE[1:0]	Clock Enable	In asynchronous mode: b1 b0 0 0: Internal clock or SCK pin is used for input pin (input signal is ignored). The SCK pin state depends on the SCKIO and SCKDT bits in SPTR. 0 1: Internal clock or SCK pin is used for clock output (The output clock frequency is 16 or 8 times of the bit rate). 1 0: External clock or SCK pin is used for clock input (The input clock frequency is 16 or 8 times of the bit rate). 1 1: Setting prohibited In clock synchronous mode: b1 b0 0 0: Internal clock or SCK pin is used for synchronous clock output. 0 1: Internal clock or SCK pin is used for synchronous clock output. 1 0: External clock or SCK pin is used for synchronous clock input. 1 1: Setting prohibited	R/W
b2	TEIE*1	Transmit End Interrupt Enable	0: Transmit end interrupt (TEI) request is disabled. 1: Transmit end interrupt (TEI) request is enabled.	R/W
b3	REIE	Receive Error Interrupt Enable	0: Receive-error interrupt (ERI) and break interrupt (BRI) requests are disabled. 1: Receive-error interrupt (ERI) and break interrupt (BRI) requests are enabled.	R/W
b4	RE	Receive Enable	0: Data reception is disabled. 1: Data reception is enabled.	R/W
b5	TE	Transmit Enable	0: Data transmission is disabled. 1: Data transmission is enabled.	R/W
b6	RIE	Receive Interrupt Enable	0: Receive-FIFO-data-full interrupt (RXI), receive-data ready interrupt (DRI), receive-error interrupt (ERI), and break interrupt (BRI) requests are disabled. 1: Receive-FIFO-data-full interrupt (RXI), receive-data ready interrupt (DRI), receive-error interrupt (ERI), and break interrupt (BRI) requests are enabled.	R/W
b7	TIE	Transmit Interrupt Enable	0: Transmit-FIFO-data-empty interrupt request (TXI) is disabled. 1: Transmit-FIFO-data-empty interrupt request (TXI) is enabled.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. TEI interrupt requests can be cleared by reading 1 from the TEND flag, and then clearing the setting to 0, or by setting the TEIE bit to 0.

CKE[1:0] Bits (Clock Enable)

Select the SCIFA clock source and enable or disable clock output from the SCK pin. Depending on the settings of these bits, the SCK pin can be used for serial clock output or serial clock input. If the SCK pin is set for the synchronous clock output in the clock synchronous mode, set the CM bit in the SMR register to 1, and then set the CKE[1:0] bits. The settings of the CKE[1:0] bits are listed in Table 17.16.

REIE Bit (Receive Error Interrupt Enable)

Specifies whether to enable or disable a receive-error interrupt (ERI) request and a break interrupt (BRI) request. The setting of this bit is only valid when the RIE bit is set to 0.

Note: ERI interrupt requests can be cleared by reading 1 from the ER bit in the FSR register, and then clearing the setting to 0, or by clearing both the RIE and REIE bits in this register to 0. BRI interrupt requests can be cleared by reading 1 from the BRK bit in the FSR register, or from the ORER flag in the LSR register, and then clearing the setting to 0, or by clearing both the RIE and REIE bits in this register to 0.

RE Bit (Receive Enable)

Specifies whether to enable or disable the serial data reception.

Note: Setting this bit to 0 does not affect the receive flags (DR, ER, BRK, RDF, FER, and PER in the FSR register, and ORER in the LSR register). These flags retain their previous values.

Note: Serial reception starts when a start bit is detected in asynchronous mode, or a synchronous clock input is detected in clock synchronous mode. Before setting this bit to 1, be sure to set the serial mode register (SMR) and the FIFO control register (FCR) to select the receive format and reset the receive FIFO.

TE Bit (Transmit Enable)

Specifies whether to enable or disable the serial data transmission.

Note: Serial transmission starts after writing of data to be transmitted into the FTDR register under this condition. Before setting this bit to 1, be sure to set the serial mode register (SMR) and the FIFO control register (FCR) to select the transmit format and reset the transmit FIFO.

RIE Bit (Receive Interrupt Enable)

Specifies whether to enable or disable a receive-FIFO-data-full (RXI) interrupt request when the RDF flag in the serial status register (FSR) is set to 1, a receive-data ready (DRI) interrupt request when the DR flag in the FSR register is set to 1, a receive-error (ERI) interrupt request when the ER flag in the FSR register is set to 1, and a break (BRI) interrupt request when the BRK flag in the FSR register or the ORER flag in the line status register (LSR) is set to 1.

Note: RXI interrupt requests can be cleared by reading 1 from the DR or RDF flag in the FSR register, then clearing the flag to 0, or by clearing the RIE bit to 0. DRI interrupt requests can be cleared by reading 1 from the DR flag in the FSR register, and then clearing the setting to 0, or by clearing the RIE bit in this register to 0. Receive error interrupt (ERI) requests and break interrupt (BRI) requests can be cleared by clearing both the RIE and REIE bits in this register to 0.

TIE Bit (Transmit Interrupt Enable)

Specifies whether to enable or disable a transmit-FIFO-data-empty interrupt (TXI) request when the serial transmit data is transferred from the transmit FIFO data register (FTDR) into the transmit shift register (TSR), the quantity of data in the transmit FIFO data register falls below the specified trigger number for transmission, and the TDFE flag in the serial status register (FSR) is set to 1.

Note: TXI interrupt requests can be cleared either by writing a greater quantity of transmit data than the specified transmission trigger number into the FTDR register, reading 1 from the TDFE flag, and then clearing the TDFE flag to 0, or by clearing this bit to 0.

17.2.7 Serial Status Register (FSR)

The FSR register is a 16-bit register. The 8 lower-order bits indicate the status flag representing the SCIFA operating state.

The CPU can always read and write to the FSR register, but cannot write 1 to the status flags (ER, TEND, TDFE, BRK, RDF, and DR bits) in this register. These flags can be only cleared to 0 when they have first been read (after being set to 1). b3 (FER) and b2 (PER) are read-only bits that cannot be written.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ER	TEND	TDFE	BRK	FER	PER	RDF	DR
Value after reset:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DR	Receive Data Ready Flag	0: Reception is in progress, or no received data has remained in the FRDR register after normally completed receiving. 1: Next receive data has not been received.	R/(W)*1
b1	RDF	Receive FIFO Data Full Flag	0: The quantity of receive data in the FRDR register falls below the specified reception trigger number. 1: The quantity of receive data written in the FRDR register is equal to or greater than the specified reception trigger number.	R/(W)*1
b2	PER	Parity Error Flag*4	0: No receive parity error occurred in the next receive data read from the FRDR register. 1: A receive parity error occurred in the next receive data read from the FRDR register.	R
b3	FER	Framing Error Flag*4	0: No receive framing error occurred in the next data read from the FRDR register. 1: A receive framing error occurred in the next data read from the FRDR register.	R
b4	BRK	Break Detect Flag	0: No break signal is received. 1: A break signal is received.*2	R/(W)*1
b5	TDFE	Transmit FIFO Data Empty Flag	0: The quantity of transmit data written in the FTDR register exceeds the specified transmission trigger number. 1: The quantity of transmit data written in the FTDR register is equal to or less than the specified transmission trigger number.*3	R/(W)*1
b6	TEND	Transmit End Flag	0: Transmission is in the waiting state or in progress. 1: Transmission is completed.	R/(W)*1
b7	ER	Receive Error Flag	0: Reception is in progress or has normally completed. 1: A framing error or parity error has occurred during reception.	R/(W)*1
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. 0 can be only written to clear the flag after 1 is read.

Note 2. When a break signal is detected, transfer of the receive data (00h) to the FRDR register stops after the detection. When the break ends and the receive signal becomes mark state (high level), the transfer of receive data resumes.

Note 3. Since the FTDR register is a 16-byte FIFO register, the maximum quantity of data that can be written when TDFE is 1 is “16 minus the number of non-transmitted data units”. If additional data is written, the data is ignored. The quantity of data in the FTDR register is indicated by the 8 higher-order bits of the FDR register.

Note 4. When the DMAC is used to read data, the generation of errors cannot be checked by reading this flag.

DR Bit (Receive Data Ready Flag)

Indicates that the quantity of data stored in the receive FIFO data register (FRDR) falls below the specified reception trigger number, and that no next data has been received yet after the elapse of 15 ETUs from the last stop bit in asynchronous mode. In clock synchronous mode, this bit is not set to 1.

[Setting condition]

- DR is set to 1 when the FRDR register contains less data than the specified reception trigger number, and no next data has been received yet after the elapse of 15 ETUs*1 from the last stop bit.

[Clearing conditions]

When either of the following is satisfied:

- DR is cleared to 0 when DR = 1 is read and then 0 is written to the DR flag.
- DR is cleared to 0 when all received data in the FRDR register are read.

Note 1. This is equivalent to one and a half (1.5) frames in the 8-bit format with one stop bit (ETU: elementary time unit).
Note 2. When the RE bit in SCR is cleared to 0, the DR bit is not affected and retains its previous value.

RDF Bit (Receive FIFO Data Full Flag)

Indicates that receive data has been transferred to the receive FIFO data register (FRDR), and the quantity of data in FRDR becomes equal to or greater than the specified reception trigger number.

[Setting condition]

- RDF is set to 1 when the quantity of receive data which is equal to or greater than the specified reception trigger number are stored in the FRDR register*1.

[Clearing conditions]

- RDF is cleared to 0 when RDF = 1 is read and then 0 is written to this bit.
- RDF is cleared to 0 when the FRDR register is read.

Note 1. Since the FRDR register is a 16-byte FIFO register, the maximum quantity of data that can be read when this bit is 1 is equivalent to the specified reception trigger number. If an attempt is made to read after all the data in the FRDR register has been read, the read data is undefined. The quantity of receive data in the FRDR register is indicated by the 8 lower-order bits of the FDR register.

PER Bit (Parity Error Flag)

Indicates whether there is a parity error in the data read from the receive FIFO data register (FRDR) in asynchronous mode.

[Setting condition]

- PER is set to 1 when a parity error is present in the next data read from the FRDR register.

[Clearing condition]

- PER is cleared to 0 when no parity error is present in the next data read from the FRDR register.

FER Bit (Framing Error Flag)

Indicates whether there is a framing error in the data read from the receive FIFO data register (FRDR) in asynchronous mode.

[Setting condition]

- FER is set to 1 when a framing error is present in the next data read from the FRDR register.

[Clearing condition]

- FER is cleared to 0 when no framing error is present in the next data read from the FRDR register.

BRK Bit (Break Detect Flag)

Indicates that a break signal has been detected in receive data.

[Setting condition]

- BRK is set to 1 when data including a framing error is received, and the framing error is followed by at least one frame of data received at the space 0 level (low level).

[Clearing condition]

- BRK is cleared to 0 when software reads BRK after it has been set to 1 and then writes 0 to BRK.

TDFE Bit (Transmit FIFO Data Empty Flag)

Indicates that data has been transferred from the transmit FIFO data register (FTDR) into the transmit shift register (TSR), the quantity of data in the FRDR register becomes equal to or less than the specified transmission trigger number, and writing of transmit data to the FRDR register is enabled.

[Setting conditions]

When either of the following is satisfied:

- TDFE is set to 1 when the TE bit in SCR is 0.
- TDFE is set to 1 when the quantity of transmit data written in the FRDR register is equal to or less than the specified transmission trigger number.

[Clearing conditions]

- TDFE is cleared to 0 when 0 is written in the TDFE bit after reading TDFE = 1.
- When transmit data is written to the FTDR register

TEND Bit (Transmit End Flag)

Indicates that the FRDR register contains no more valid data and transmission is completed when transmitting the last bit of the transmit data.

[Setting condition]

When the following is satisfied:

- TEND is set to 1 when the FTDR register does not contain transmit data when the last bit of the serial transmission data is transmitted.

[Clearing conditions]

- When transmit data is written to the FTDR register
- When 0 is written to TEND after it has been read as 1

ER Bit (Receive Error Flag)

Indicates the occurrence of a framing error, or of a parity error when receiving the parity-added data*1.

[Setting conditions]

When either of the following is satisfied:

- ER is set to 1 when the stop bit is found to be 0 after checking whether the stop bit of the received data is 1 at the end of one data receive operation*1.
- ER is set to 1 when the total number of 1s in the received data and parity bit combined does not match the even or odd parity setting specified by the PM bit in the SMR register.

[Clearing condition]

- When 0 is written to ER after it has been read as 1

Note 1. Clearing the RE bit to 0 in the SCR register does not affect this bit, which retains its previous value. Even if a receive error occurs, the receive data is transferred to the FRDR register and the receive operation is continued. Whether the data read from the FRDR register includes a receive error can be detected by the FER and PER bits in the FSR register.

Note 2. In two stop bits mode, only the first stop bit is checked; the second stop bit is not checked.

17.2.8 Bit Rate Register (BRR)

The BRR register is an 8-bit register that, together with the baud rate generator clock source selected by the CKS[1:0] bits in the serial mode register (SMR), determines the serial transmit/receive bit rate.
This register is located in the same address as that of the MDDR register and selected when the MDDRS bit in SEMR is 0. The CPU can read and write to BRR. Writing to BRR should be executed when TE = RE = 0 in the SCR register.

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

The BRR setting is calculated using the following formulae.
[Asynchronous mode]

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$N = \frac{P1\varphi}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0))

$$N = \frac{P1\varphi}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$N = \frac{P1\varphi}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0))

$$N = \frac{P1\varphi}{16 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1))

[Clock synchronous mode]

$$N = \frac{P1\varphi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bit/s)
N: Setting of the BRR register ($0 \leq N \leq 255$) (The setting must satisfy the electrical characteristics).
P1φ: Operating frequency for peripheral modules (MHz)
n: Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see Table 17.4).

Note: The MDDR register is used to adjust the bit rate. For details, see section 17.2.9, Modulation Duty Register (MDDR).

Table 17.4 SMR Register Setting

		SMR Register Settings	
n	Clock Source	CKS1	CKS0
0	P1φ	0	0
1	P1φ/4	0	1
2	P1φ/16	1	0
3	P1φ/64	1	1

The bit rate error in asynchronous mode is calculated using the following formulae.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$\text{Error (\%)} = \left\{ \frac{P1\varphi \times 10^6}{(N + 1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0))

$$\text{Error (\%)} = \left\{ \frac{P1\varphi \times 10^6}{(N + 1) \times B \times 32 \times 2^{2n-1}} - 1 \right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$\text{Error (\%)} = \left\{ \frac{P1\varphi \times 10^6}{(N + 1) \times B \times 32 \times 2^{2n-1}} - 1 \right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0))

$$\text{Error (\%)} = \left\{ \frac{P1\varphi \times 10^6}{(N + 1) \times B \times 16 \times 2^{2n-1}} - 1 \right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1))

Table 17.5 list the examples of the BRR register setting in asynchronous mode, and Table 17.6 list the examples of the BRR register setting in clock synchronous mode.

Table 17.5 Bit Rates and BRR Register Settings in Asynchronous Mode

Bit Rate (bps)	P1φ (MHz)					
	27.5			66		
	n	N	Error (%)	n	N	Error (%)
150	3	89	-0.54	3	214	-0.53
300	3	44	-0.54	3	106	0.39
600	2	89	-0.54	2	214	-0.07
1200	2	44	-0.54	2	106	0.39
2400	1	89	-0.54	1	214	-0.07
4800	1	44	-0.54	1	106	0.39
9600	(*1)	(*1)	(*1)	0	214	-0.07
14400	1	14	-0.54	0	142	0.16
19200	(*1)	(*1)	(*1)	0	106	0.39
28800	(*1)	(*1)	(*1)	0	71	0.54
31250	(*1)	(*1)	(*1)	0	65	0.00
38400	(*1)	(*1)	(*1)	0	53	-0.53
115200	(*1)	(*1)	(*1)	0	17	-0.53
500000	(*1)	(*1)	(*1)	0	3	(*1)

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 0. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is doubled. When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is quadrupled. Configure settings so the range of error is no greater than 1%.

Note 1. Values for the blank cells in the table can be set using the MDDR register. For details, see section 17.2.9, Modulation Duty Register (MDDR) and the Table 17.11.

Table 17.6 Bit Rates and BRR Register Settings in Clock Synchronous Mode

Bit Rate (bps)	P1φ(MHz)			
	27.5		66	
	n	N	n	N
250	—	—	—	—
500	3	214	—	—
1000	3	106	—	—
2500	3	42	3	102
5000	3	20	2	205
10000	3	10	2	102
25000	3	3	1	164
50000	3	1	1	82
100000	2	3	0	164
250000	2	1	0	65
500000	1	2	0	32
1000000	1	1	0	16
2500000	0	2	0	6
5000000	0	0*	0	2

Note: Continuous transmission or reception is not possible.

—: Setting is prohibited.

Note: Set the BRR register so that the range of error can fall within 1% or less.

Table 17.7 lists the maximum bit rates for various frequencies in asynchronous mode when the baud rate generator is used. Table 17.8 lists the maximum bit rates for various frequencies in clock synchronous mode when the baud rate generator is used. Table 17.9 and Table 17.10 list the maximum rates for external clock inputs in asynchronous mode and clock synchronous mode, respectively.

Table 17.7 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (in Asynchronous Mode)

P1φ (MHz)	Maximum Bit Rate (bit/s)	Settings	
		n	N
27.5	3437500	0	0
66	8250000	0	0

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 1. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is 1/2. When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is 1/4.

Table 17.8 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (in Clock Synchronous Mode)

	Discontinuous Transmission/Reception			Continuous Transmission/Reception		
	Maximum Bit Rate	Settings		Maximum Bit Rate	Settings	
P1φ (MHz)	(bit/s)	n	N	(bit/s)	n	N
27.5	6875000	0	0	3437500	0	1
66	16500000	0	0	8250000	0	1

Table 17.9 Maximum Bit Rates with External Clock Input (in Asynchronous Mode)

P1φ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)
27.5	6.875	859375
66	16.50	2062500

Note: This is an example when the SEMR.ABCS0 bit is 1. When the ABCS0 bit is set to 0, the bit rate is 1/2 .

Table 17.10 Maximum Bit Rates with External Clock Input (in Clock Synchronous Mode)

P1φ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)
27.5	2.29	2291667
66	5.50	5500000

17.2.9 Modulation Duty Register (MDDR)

The MDDR register corrects the bit rate adjusted by the BRR register. The value after reset of this register is FFh. When the BRME bit in SEMR is set to 1, the bit rate generated by the on-chip baud rate generator is evenly corrected according to the settings of MDDR (MDDR/256). The relationship between the MDDR register setting and the bit rate (B) is given by the following formula. The MDDR register is located in the same address as that of the BRR register and is selected when the MDDRS bit in SEMR is 1. This register is only writable when TE = RE = 0 in the SCR register. b7 in this register is fixed to 1.

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

The formulae below show the relationships between the MDDR setting and the bit rate (B) when the bit rate modulation function is used.

[Asynchronous mode]

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$B = \frac{P1\varphi \times 10^6}{64 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$B = \frac{P1\varphi \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$B = \frac{P1\varphi \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$B = \frac{P1\varphi \times 10^6}{64 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

[Clock synchronous mode]

$$B = \frac{P1\varphi \times 10^6}{8 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

When the bit rate modulation is used, the bit rate average error is given by the following formulae.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$Error\ (\%) = \left\{ \frac{P1\varphi \times 10^6}{B \times 64 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)} - 1 \right\} 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$Error\ (\%) = \left\{ \frac{P1\varphi \times 10^6}{B \times 32 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)} - 1 \right\} 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$\text{Error (\%)} = \left\{ \frac{P1\varphi \times 10^6}{B \times 32 \times 2^{2n-1} \times (256/MDDR) \times (N+1)} - 1 \right\} 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\text{Error (\%)} = \left\{ \frac{P1\varphi \times 10^6}{B \times 16 \times 2^{2n-1} \times (256/MDDR) \times (N+1)} - 1 \right\} 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

B: Bit rate (bits/s)
N: BRR register setting ($0 \leq N \leq 255$)
The setting must satisfy the electrical characteristics).
P1φ:Operating frequency for peripheral modules (MHz)
MDDR: MDDR setting ($128 \leq MDDR \leq 256$)
n: Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see Table 17.4).

Table 17.11 Bit Rates and BRR and MDDR Registers Settings in Asynchronous Mode

Bit Rate (bps)	P1φ (MHz)							
	27.5				66			
	n	N	MDDR	Error (%)	n	N	MDDR	Error (%)
150	—	—	—	—	3	213	255	0.00
300	—	—	—	—	3	106	255	0.00
600	—	—	—	—	2	213	255	0.00
1200	—	—	—	—	2	106	255	0.00
2400	—	—	—	—	1	213	255	0.00
4800	—	—	—	—	1	106	255	0.00
9600	1	21	252	0.14	0	213	255	0.00
14400	1	14	254	-0.93	—	—	—	—
19200	1	10	253	0.53	0	106	255	0.00
28800	1	6	242	0.74	0	70	254	0.08
31250	1	5	225	0.71	—	—	—	—
38400	1	4	230	0.53	0	52	253	0.15
115200	0	6	240	-0.09	—	—	—	—
500000	0	0	150	0.71	0	3	249	0.31

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 0. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is doubled. When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is quadrupled. Configure settings so the range of error is no greater than 1%.

17.2.10 FIFO Control Register (FCR)

The FCR register resets the quantity of data in the transmit FIFO data register (FTDR) and the receive FIFO data register (FRDR) and specifies the number of triggers. This register also specifies whether to enable the loop-back test.
The CPU can always read and write to the FCR register.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	RSTRG[2:0]			RTRG[1:0]		TTRG[1:0]		MCE	TFRST	RFRST	LOOP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	LOOP	Loop-Back Test	0: Loop back test is disabled. 1: Loop back test is enabled	R/W
b1	RFRST	Receive FIFO Data Register Reset	0: Normal operation 1: Resets the FRDR register.	R/W
b2	TFRST	Transmit FIFO Data Register Reset	0: Normal operation 1: Resets the FTDR register.	R/W
b3	MCE	Modem Control Enable	0: Model signal is disabled.*1 1: Model signal is enabled.	R/W
b5, b4	TTRG[1:0]	Transmit FIFO Data Trigger Number Select	b5 b4 0 0: 8 (8)*2 0 1: 4 (12)*2 1 0: 2 (14)*2 1 1: 0 (16)*2	R/W
b7, b6	RTRG[1:0]	Receive FIFO Data Trigger Number Select	In asynchronous mode: b7 b6 0 0: 1 0 1: 4 1 0: 8 1 1: 14 In clock synchronous mode: b7 b6 0 0: 1 0 1: 2 1 0: 8 1 1: 14	R/W
b10 to b8	RSTRG[2:0]	RTS# Output Active Trigger Number Select	b10 b8 0 0 0: 15 0 0 1: 1 0 1 0: 4 0 1 1: 6 1 0 0: 8 1 0 1: 10 1 1 0: 12 1 1 1: 14	R/W
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The CTS# input level does not affect the transmit operation. Similarly, the RTS# input level does not affect the receive operation.
Note 2. Values in parentheses mean the number of empty bytes in the FTDR register when the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXI) request is generated.

LOOP Bit (Loop-Back Test)

Internally connects between the transmit output pin (TXD) and the receive input pin (RXD) and between the RTS# pin and the CTS# pin, to perform loop-back testing.

RFRST Bit (Receive FIFO Data Register Reset)

Disables the receive data in the receive FIFO data register (FRDR) and makes the data to the empty state. If you set this bit to 1, be sure to clear it to 0 afterward.

TFRST Bit (Transmit FIFO Data Register Reset)

Disables the transmit data in the transmit FIFO data register (FTDR) and makes the data to the empty state. If you set this bit to 1, be sure to clear it to 0 afterward.

MCE Bit (Modem Control Enable)

Specifies whether to enable or disable the modem control signals, CTS# and RTS#.
In clock synchronous mode, this bit should always be set to 0.

TTRG[1:0] Bits (Transmit FIFO Data Trigger Number Select)

Specify the reference quantity of data for transmission (i.e., the threshold number of entries to trigger the writing of further data for transmission) for setting of the TDFE flag in the serial status register (FSR). When the number of entries for transmission in the transmission FIFO, i.e. the number of entries written to the transmit FIFO data register (FTDR) that are yet to be transmitted, falls to or below the specified trigger number for transmission, the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXI) request is generated.
The setting in these bits is valid when the TTRGS bit in the FTCR register is 0. When the TTRGS bit in the FTCR register is 1, the setting of the TFTC[4:0] bits in the FTCR register is valid.

RTRG[1:0] Bits (Receive FIFO Data Trigger Number Select)

Specify the reference quantity of receive data (i.e., the threshold number of entries to trigger the reading of received data) for setting of the RDF flag in the serial status register (FSR). When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number for reception, the RDF flag is set to 1 and a receive FIFO data full interrupt (RXI) request is generated.
The setting in these bits is valid when the RTRGS bit in the FTCR register is 0. When the RTRGS bit in the FTCR register is 1, the setting of the RFTC[4:0] bits in the FTCR register is valid.

RSTRG[2:0] Bits (RTS# Output Active Trigger Number Select)

When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number, the RTS# signal is in the high state.
The setting in these bits is only valid when a modem signal is enabled by the MCE bit in this register in asynchronous mode.

17.2.11 FIFO Data Count Register (FDR)

The FDR register indicates the quantity of data stored in the transmit FIFO data register (FTDR) and the receive FIFO data register (FRDR).

This register indicates the quantity of transmit data in the FTDR register with the 8 higher-order bits, and the quantity of receive data in the FRDR register with the 8 lower-order bits. The CPU can always read the FDR register.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	T[4:0]					—	—	—	R[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	R[4:0]	Receive Data Quantity in FRDR	Indicate the quantity of receive data stored in the FRDR register.	R
b7 to b5	—	Reserved	These bits are read as 0.	R
b12 to b8	T[4:0]	Non-Transmitted Data Quantity in FTDR	Indicate the quantity of non-transmitted data stored in the FTDR register.	R
b15 to b13	—	Reserved	These bits are read as 0.	R

R[4:0] Bits

Indicate the quantity of receive data stored in the FRDR register.

00h means no received data, and 10h means that all of the received data is stored in the FRDR register.

T[4:0] Bits

Indicate the quantity of non-transmitted data stored in the FTDR register.

00h means no transmit data, and 10h means that all of the data for transmission is stored in the FTDR register.

17.2.12 Serial Port Register (SPTR)

The SPTR register controls input/output and data of the pins multiplexed to SCIFA function.
The CPU can always read and write to the SPTR register.

Note: b6, b4, b2, and b0 of this register respectively indicate the input status of their corresponding pins. See the descriptions for each bit for details. Writings to these bits in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	RTS2IO	RTS2DT	CTS2IO	CTS2DT	SCKIO	SCKDT	SPB2IO	SPB2DT
Value after reset:	0	0	0	0	0	0	0	0	0	x	0	x	0	x	0	x

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	SPB2DT	Serial Port Break Data Select	Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2IO bit. See Table 17.14.	R/W
b1	SPB2IO	Serial Port Break Input/Output	Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2DT bit.	R/W
b2	SCKDT	SCK Port Data Select	Controls the SCK pin in combination with the CM bit in the SMR register, the SCKIO bit, and the CKE1 and CKE0 bits in the SCR register. See Table 17.16.	R/W
b3	SCKIO	SCK Port Input/Output	Controls the SCK pin in combination with the CM bit in the SMR register, the SCKDT bit, and the CKE1 and CKE0 bits in the SCR register. See Table 17.16.	R/W
b4	CTS2DT	CTS# Port Data Select	Controls the CTS# pin in combination with MCE bit in FCR and CTS2IO bit. See Table 17.13. The SCIF 3 and SCIF 4 channels are not supported.	R/W
b5	CTS2IO	CTS# Port Output Specify		R/W
b6	RTS2DT	RTS# Port Data Select	Controls the RTS# pin in combination with MCE bit in FCR and RTS2IO bit. See Table 17.12. The SCIF 3 and SCIF 4 channels are not supported.	R/W
b7	RTS2IO	RTS# Port Output Specify		R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPB2DT Bit (Serial Port Break Data Select)

This bit specifies the output level of the TXD pin when the setting of the SCR.TE bit is 0. The RXD pin input status can be read from this bit regardless of the SPB2IO bit setting. However, the RXD pin function must have been selected with the general I/O port.

SPB2IO Bit (Serial Port Break Input/Output)

Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2DT bit.

SCKDT Bit (SCK Port Data Select)

The SCK pin status can be read from this bit regardless of the SCKIO bit setting. (When the SCK pin is used for input, the input signal is invalid (has no means) but the pin status can be read.) However, the SCK pin function must have been selected with the general I/O port.

SCKIO Bit (SCK Port Input/Output)

Specifies input or output status of the SCK pin. This bit controls the SCK pin in combination with the SCKDT bit, the CM bit in the SMR register, and the CKE1 and CKE0 bits in the SCR register.

CTS2DT Bit (CTS# Port Data Select)

The status of the CTS# pin can be read from this bit regardless of the CTS2IO bit setting. However, the CTS# pin function must have been selected with the general I/O port.

RTS2DT Bit (RTS# Port Data Select)

The status of the RTS# pin can be read from this bit regardless of the RTS2IO bit setting. However, the RTS# pin function must have been selected with the general I/O port

Table 17.12 RTS# Pin Status

FCR.MCE Bit Setting	RTS2IO Bit Setting	RTS2DT Bit Setting	RTS# Pin Status
0	0	×	Setting prohibited*1
0	1	0	Low output
0	1	1	High output
1	×	×	Modem control output

×: Don't care

Note 1. There is no problem with the initial setting if the RTS# pin is not used.

Table 17.13 CTS# Pin Status

FCR.MCE Bit Setting	CTS2IO Bit Setting	CTS2DT Bit Setting	CTS# Pin Status
0	0	×	Setting prohibited*1
0	1	0	Low output
0	1	1	High output
1	×	×	Modem control input

×: Don't care

Note 1. There is no problem with the initial setting if the CTS# pin is not used.

Table 17.14 TXD Pin Status

SCR.TE Bit Setting	SPB2IO Bit Setting	SPB2DT Bit Setting	TXD Pin Status
0	0	×	Setting prohibited
0	1	0	Low output
0	1	1	High output
1	×	×	Transmit data output

×: Don't care

17.2.13 Line Status Register (LSR)

The LSR register is a 16-bit register. The PER and FER bits indicate the number of receive errors in the receive FIFO data register. 1 cannot be written to the OREER status flag. The flag should be read as 1 prior to clearing it to 0.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PER[3:0]				—	—	FER[3:0]				—	ORER
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ORER	Overrun Error Flag	0: Reception is in progress or has normally completed. 1: An overrun error has occurred during reception.	R/(W)*1
b1	—	Reserved	This bit is read as 0.	R
b5 to b2	FER[3:0]	Framing Error Count	Indicates the quantity of data with a framing error among the receive data stored in the receive FIFO data register (FRDR).	R
b7, b6	—	Reserved	These bits are read as 0.	R
b11 to b8	PER[3:0]	Parity Error Count	Indicates the quantity of data with a parity error among the receive data stored in the receive FIFO data register (FRDR).	R
b15 to b12	—	Reserved	These bits are read as 0.	R

Note 1. To clear the flag, 0 can be only written after 1 is read.

ORER Bit (Overrun Error Flag)

Indicates that receive operation abnormally stops due to occurrence of an overrun error. This flag is not affected and retains its previous state if the RE bit in the serial control register (SCR) is cleared to 0. The receive FIFO data register (FRDR) retains the data before an overrun error occurred, and newly received data is lost. When the ORER bit is set to 1, the SCIFA cannot continue subsequent serial reception.

[Setting condition]

- When the next serial reception is completed with the receive FIFO in full state (16-byte data is received)

[Clearing condition]

- When 0 is written to ORER after being read as 1.

Note: When the internal clock is selected while the SCIFA is in clock synchronous mode, the amount of receive data can be controlled, so no overrun occurs.

FER[3:0] Bits (Framing Error Count)

The values of bits 5 to 2 indicate the quantity of data with a framing error after the ER bit in the FSR register is set. Reading 0000 from the FER[3:0] bits means all 16-byte receive data in the FRDR register have a framing error.

PER[3:0] Bits (Parity Error Count)

The values of bits 11 to 8 indicate the quantity of data with a parity error after the ER bit in the FSR register is set. Reading 0000 from the PER[3:0] bits means all 16-byte receive data in the FRDR register have a parity error.

17.2.14 Serial Extended Mode Register (SEMR)

The SEMR register specifies either LSB or MSB first, enables the noise cancellation, operation in normal or double-speed mode of the baud rate generator, and bit rate modulation, and selects the modulation register and the sampling count (either 8 or 16 times).

	b7	b6	b5	b4	b3	b2	b1	b0
	BGDM	—	BRME	MDDRS	DIR	NFEN	—	ABCS0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ABCS0	Asynchronous Base Clock Select	0: Operates on a frequency 16 times the transfer rate as the base clock. 1: Operates on a frequency 8 times the transfer rate as the base clock.	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	NFEN	Noise Cancellation Enable	0: Noise cancellation for the RxD pin is disabled. 1: Noise cancellation for the RxD pin is enabled.	R/W
b3	DIR	Data Transfer Direction Select	0: Transmits the data in the FTDR register by the LSB-first method. The received data is stored in the FRDR register by the LSB-first method. 1: Transmits the data in the FTDR register by the MSB-first method. The received data is stored in the FRDR register by the MSB-first method.	R/W
b4	MDDRS	Modulation Duty Register Select	0: BRR register is accessible. 1: MDDR register is accessible.	R/W
b5	BRME	Bit Rate Modulation Enable	0: Bit rate modulation is disabled. 1: Bit rate modulation is enabled.	R/W
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b7	BGDM	Baud Rate Generator Double-Speed Mode Select	0: Baud rate generator normal mode: Baud rate generator operates on the clock signal produced by dividing the clock source by two. 1: Baud rate generator double-speed mode: Baud rate generator operates on the clock signal produced by the clock source (no frequency division).	R/W

ABCS0 Bit (Asynchronous Base Clock Select)

Selects the base clock for 1-bit period in asynchronous mode.

This bit setting is valid only in asynchronous mode (i.e., when the CM bit in the SMR register is 0).

NFEN Bit (Noise Cancellation Enable)

Reduces noise of the input to the RxD pin. This function is only valid in asynchronous mode. For details, see section 17.7, Noise Cancellation.

In clock synchronous mode, this bit should always be set to 0.

DIR Bit (Data Transfer Direction Select)

Selects the serial communication format. This bit is valid only when the transmit/receive data is in 8-bit formats.*1

Note 1. Asynchronous mode or clock synchronous mode with the 8-bit data length

MDDRS Bit (Modulation Duty Register Select)

Selects the register to be enabled access to it.

BRME Bit (Bit Rate Modulation Enable)

Specifies whether to enable or disable the bit rate modulation.

BGDM Bit (Baud Rate Generator Double-Speed Mode Select)

Selects operating mode of the baud rate generator. When setting 1 in this bit, the baud rate generator included in the SCIFA operates in double-speed mode. The setting of this bit is only effective in asynchronous mode (SMR.CM bit = 0) when the internal clock is selected as the clock source (SCR.CKE[1:0] = 00b). Use normal mode under any other settings.

17.2.15 FIFO Trigger Control Register (FTCR)

The FTCR register is a 16-bit register that specifies FIFO trigger conditions. The CPU can always read from and write to the FTCR register.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RTRGS	—	—	RFTC[4:0]					TTRGS	—	—	TFTC[4:0]				
Value after reset:	0	0	0	1	1	1	1	1	0	0	0	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	TFTC[4:0]	Transmit FIFO Data Trigger Number	00h: Transmit data trigger number is 0. 0Fh: Transmit data trigger number is 15. Do not set 10h to 1Fh in these bits.	R/W
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	TTRGS	Transmit Trigger Select	0: TTRG[1:0] bits in FCR are valid. 1: TFTC[4:0] bits in FTCR are valid.	R/W
b12 to b8	RFTC[4:0]	Receive FIFO Data Trigger Number	01h: Receive data trigger number is 1. 10h: Receive data trigger number is 16. Do not set 00h and 11h to 1Fh in these bits.	R/W
b14, b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	RTRGS	Receive Trigger Select	0: RTRG[1:0] bits in FCR are valid. 1: RFTC[4:0] bits in FTCR are valid.	R/W

TFTC[4:0] Bits (Transmit FIFO Data Trigger Number)

Specify the reference quantity of data for transmission (i.e., the threshold number of entries to trigger the writing of further data for transmission) for setting of the TDFE flag in the serial status register (FSR).

When the number of entries for transmission in the transmission FIFO, i.e. the number of entries written to the transmit FIFO data register (FTDR) that are yet to be transmitted, falls to or below the specified trigger number for transmission, the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXI) request is generated.

RFTC[4:0] Bits (Receive FIFO Data Trigger Number)

Specify the reference quantity of receive data (i.e., the threshold number of entries to trigger the reading of received data) for setting of the RDF flag in the serial status register (FSR).

When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number for reception, the RDF flag is set to 1 and a receive FIFO data full interrupt (RXI) request is generated.

17.3 Operation

17.3.1 Overview

For serial communication, the SCIFA can select either asynchronous mode in which characters are synchronized individually or a clock synchronous mode in which communication is synchronized with clock pulses.
The SCIFA has a 16-stage FIFO buffer for both transmission and receptions, reducing the overhead of the CPU and enabling continuous high-speed communication. The RTS# and CTS# signals are provided as modem control signals.
Selection of a transmission/reception format is enabled with the serial mode register (SMR). Table 17.15 shows the transmission format which can be selected in the serial mode register (SMR). As shown in Table 17.16, the SCIFA clock source can be selected by the combination of the CKE1 and CKE0 bits in the serial control register (SCR).

(1) Asynchronous Mode

- Data length is selectable either 7 or 8 bits
- Parity addition and 1- or 2-bit stop bit addition are selectable.
  (The combination of the preceding selections determines the transmission/reception format and character length).
- In reception, it is possible to detect framing errors, parity errors, receive FIFO data full, overrun errors, receive data ready, and breaks.
- The stored data quantities are indicated in the FIFO data count register (FDR), respectively for transmit and receive FIFO data.
- An internal or external clock can be selected as the SCIFA clock source.
  When an internal clock is selected, the SCIFA operates using the clock of on-chip baud rate generator and can output the clock with a frequency 16 (or 8) times the bit rate.
  When an external clock is selected, the external clock input must have a frequency 16 (or 8) times the bit rate. (The on-chip baud rate generator is not used.)

(2) Clock Synchronous Mode

- The transmission/reception format is fixed to the 8-bit data length.
- In reception, it is possible to detect overrun errors (ORER).
- An internal or external clock can be selected as the SCIFA clock source.
  When an internal clock is selected, the SCIFA operates using the clock of the on-chip baud rate generator, and outputs this clock to external devices as the synchronous clock.
  When an external clock is selected, the SCIFA operates on the input synchronous clock not using the on-chip baud rate generator.

Table 17.15 SMR Register Settings and SCIFA Communication Formats

SMR Register					SCIFA Transmission/Reception Format		
b7	b6	b5	b3				
CM	CHR	PE	STOP	Mode	Data Length	Parity Bit	Stop Bit Length
0	0	0	0	Asynchronous mode	8 bits	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
	1	0	0		7 bits	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
1	×	×	×	Clock synchronous mode	8 bits	Not set	None

×: Don't care

Table 17.16 SMR, SCR, and SPTR Register Settings and SCIFA Clock Source Selection

SMR Register	SCR Register		SPTR Register				
b7	b1	b0	b3	b2			
CM	CKE1	CKE0	SCKIO	SCKDT	Mode	Clock Source	SCK Pin Function
0	0	0	0	×	Asynchronous mode	Internal	Input pin (input signal invalid) (Initial state)
			1	0			SCK pin state: Low
			1	1			SCK pin state: High
		1	×	×			Outputs a clock with frequency 16/8 times the bit rate*1
	1	0	×	×		External	Inputs a clock with frequency 16/8 times the bit rate*2
		1	×	×		Setting prohibited	
1	0	×	×	×	Clock synchronous mode	Internal	Outputs the synchronous clock
	1	0	×	×		External	Inputs the synchronous clock
		1	×	×		Setting prohibited	

×: Don't care

Note 1. SEMR.ABCS0 = 0: Output a clock that has a frequency 16 times the bit rate.
SEMR.ABCS0 = 1: Output a clock that has a frequency 8 times the bit rate.

Note 2. SEMR.ABCS0 = 0: Input a clock that has a frequency 16 times the bit rate.
SEMR.ABCS0 = 1: Input a clock that has a frequency 8 times the bit rate.

17.3.2 Operation in Asynchronous Mode

In asynchronous mode, each transmitted or received character begins with a start bit and ends with a stop bit. Serial communication is synchronized one character at a time.

The transmitting and receiving sections of the SCIFA are independent, so full duplex communication is possible. The transmitter and receiver are 16-stage FIFO buffered, so data can be written and read while transmitting and receiving are in progress, enabling continuous transmission and reception.

Figure 17.2 shows the general format of asynchronous serial communication.

In asynchronous serial communication, the communication line is normally held in the mark (high) state. The SCIFA monitors the line and starts serial communication when the line goes to the space (low) state, considered as a start bit. One serial character consists of a start bit (low), data (LSB first when LSB-first transfer is selected), parity bit (high or low), and stop bit (high), in that order.

When receiving in asynchronous mode, the SCIFA synchronizes at the falling edge of the start bit. The SCIFA samples each data bit on the eighth pulse of a clock with a frequency 16 or 8 times the bit rate*1. Receive data is latched at the center of each bit.

Note 1. When the SEMR.ABCS0 bit = 0, data is sampled on the eighth pulse of a clock with a frequency 16 times the bit rate.
When the SEMR.ABCS0 bit = 1, data is sampled on the fourth pulse of a clock with a frequency 8 times the bit rate.

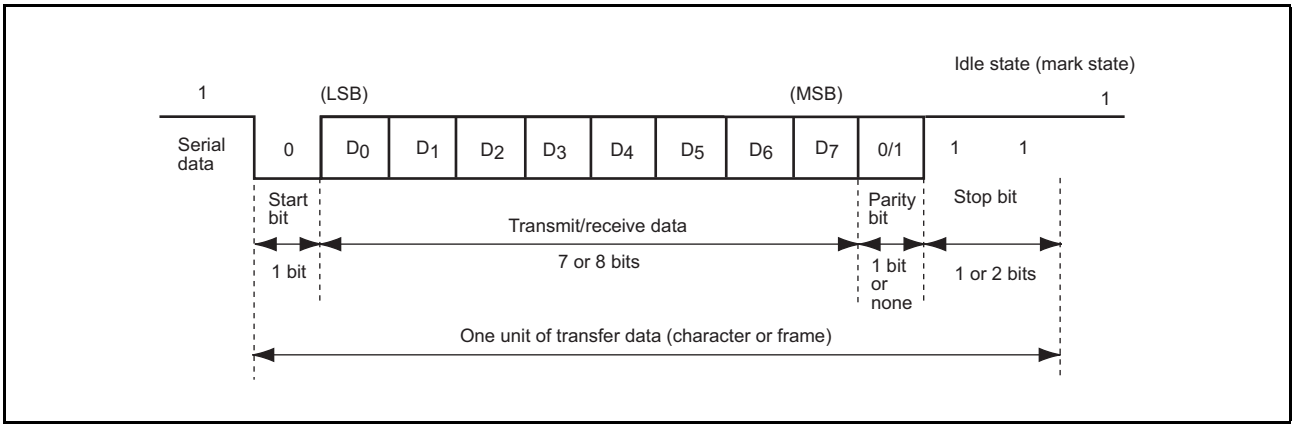


Figure 17.2 Data Format in Asynchronous Communication
(8-Bit Data with Parity and Two Stop Bits when LSB-First Transfer is Selected)

(1) Transmit/Receive Formats

Table 17.17 lists the eight communications formats that can be selected in asynchronous mode. The format is selected by setting in the serial mode register (SMR).

Table 17.17 Serial Communications Formats (in Asynchronous Mode)

SMR Setting			Serial Transmit/Receive Format and Frame Length											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	START	8-bit data								STOP		
		1	START	8-bit data								STOP	STOP	
	1	0	START	8-bit data								P	STOP	
		1	START	8-bit data								P	STOP	STOP
1	0	0	START	7-bit data							STOP			
		1	START	7-bit data							STOP	STOP		
	1	0	START	7-bit data							P	STOP		
		1	START	7-bit data							P	STOP	STOP	

START: Start bit
STOP: Stop bit
P: Parity bit

(2) Clock

An SCIFA transmit/receive clock can be selected from two types of clock sources: the internal clock generated by the on-chip baud rate generator, the external clock input from the SCK pin. The clock source is selected by the settings of the CM bit in the serial mode register (SMR), the CKE[1:0] bits in the serial control register (SCR), and the ACS0 bit in the serial extended mode register (SEMR). For clock source selection, refer to Table 17.16.

When an external clock is input at the SCK pin, it must have a frequency equal to 16/8 times the desired bit rate.

When the SCIFA operates on an internal clock, it can output a clock signal on the SCK pin. The frequency of this output clock is 16/8 times the desired bit rate.

(3) Transmitting and Receiving Data

- SCIFA Initialization (in Asynchronous Mode)

Before transmitting or receiving data, clear the TE and RE bits to 0 in the serial control register (SCR), and then initialize the SCIFA as follows.

When changing operating mode or communication format, always clear the TE and RE bits in the SCR register to 0 before following the procedure given below. Clearing TE to 0 initializes the transmit shift register (TSR). Clearing TE and RE to 0, however, does not initialize the serial status register (FSR), transmit FIFO data register (FTDR), or receive FIFO data register (FRDR), which retain their previous contents. Clear TE to 0 after all transmit data has been transmitted and the TEND flag in the FSR register is set. The TE bit can be cleared to 0 during transmission, but the transmit data (the TXD pin output level) after the TE bit is cleared to 0 depends on the settings of the SPB2IO and SPB2DT bits in the SPTR register. Set the TFRST bit in the FCR register to 1 and reset the FTDR register before TE is set to 1 again to start transmission.

When an external clock is used, the clock should not be stopped during initialization or subsequent operation. SCIFA operation becomes unreliable if the clock is stopped. Figure 17.3 shows a sample flowchart for initializing the SCIFA in asynchronous mode.

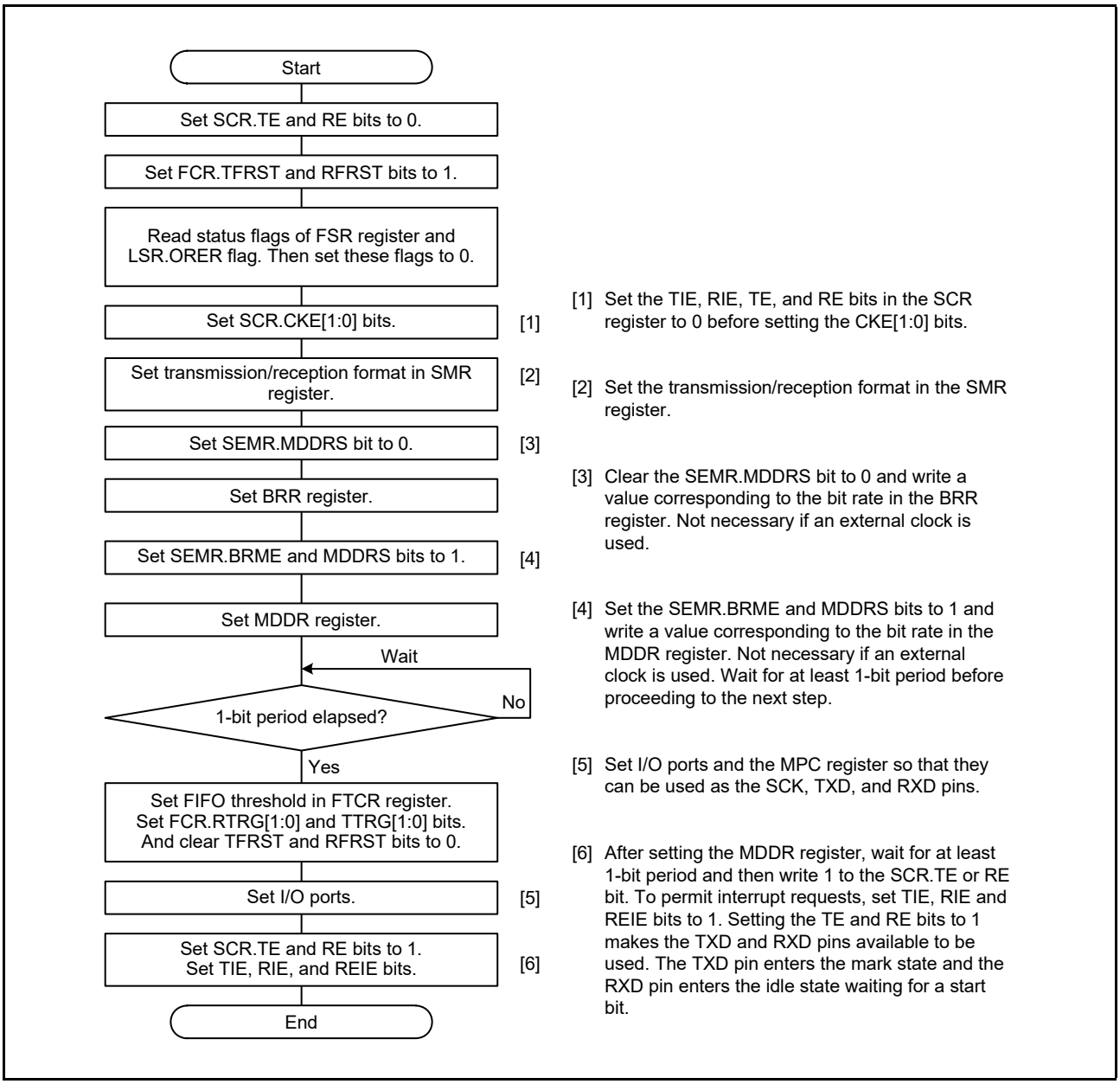
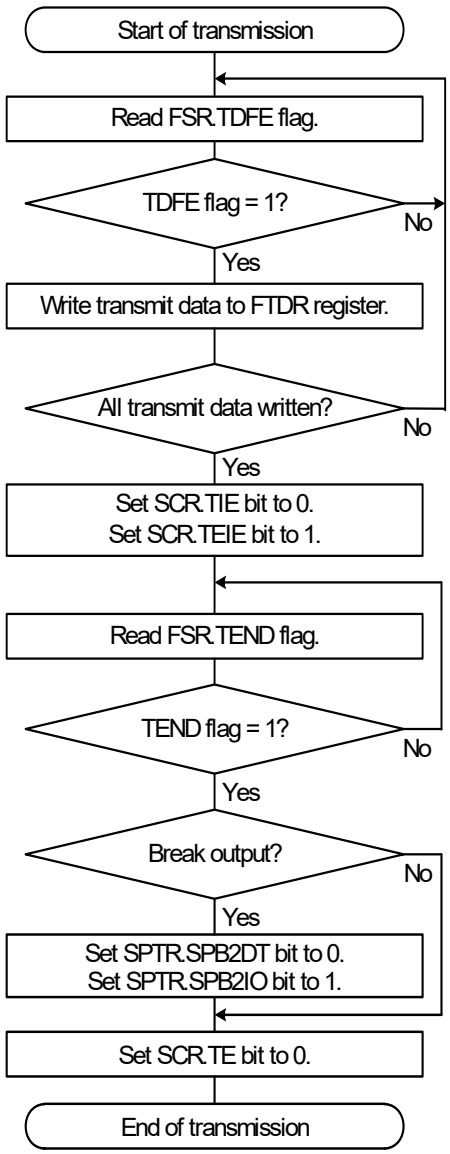


Figure 17.3 Sample Flowchart for SCIFA Initialization in Asynchronous Mode

- Transmitting Serial Data (in Asynchronous Mode)

Figure 17.4 shows a sample flowchart for serial transmission in asynchronous mode.
Follow the procedure given below for serial data transmission after enabling the SCIFA for transmission.



[1] Check the SCIFA status and write the transmit data:
Read the FSR register, confirm that the TDFE flag is set to 1, and then write the transmit data to the FTDR register. The transmit data quantity that can be written is "16 - FDR.T[4:0]".
A change of the TDFE flag from 0 to 1 can be known by a TXIF interrupt request.

[2] Serial transmission continuation procedure:
When transmit data is written to the FTDR register, the TDFE flag is automatically cleared to 0.

[3] Serial transmission end procedure (when TXIF or TEIF interrupt is used):
After the last transmission data is written to the FTDR register, set the TIE bit to 0 and the TEIE bit to 1 in the SCR register. If the DMAC is activated to write the transmit data to the FTDR register, enable the transmit end interrupt. This will generate a CPU interrupt request when the last transmit data is written in the FTDR register. The TIE and TEIE bit settings will be changed by the CPU interrupt process.
This procedure is unnecessary if the TXIF or TEIF interrupt is not used.

[4] Serial transmission end confirmation:
Read the FSR register and confirm that the TEND flag has changed to 1.
A change of the TEND flag from 0 to 1 can be known by a TEIF interrupt request.

[5] Break output during serial transmission:
To cause a break to be output during serial transmission, set the SPB2DT bit to 0 and the SPB2IO bit to 1 in the SPTR register. Then set the SCR.TE bit to 0.*1

In Step [1], the number of data bytes that can be written can be checked by the FDR.T[4:0] flag (non-transmitted data quantity in the FTDR register).

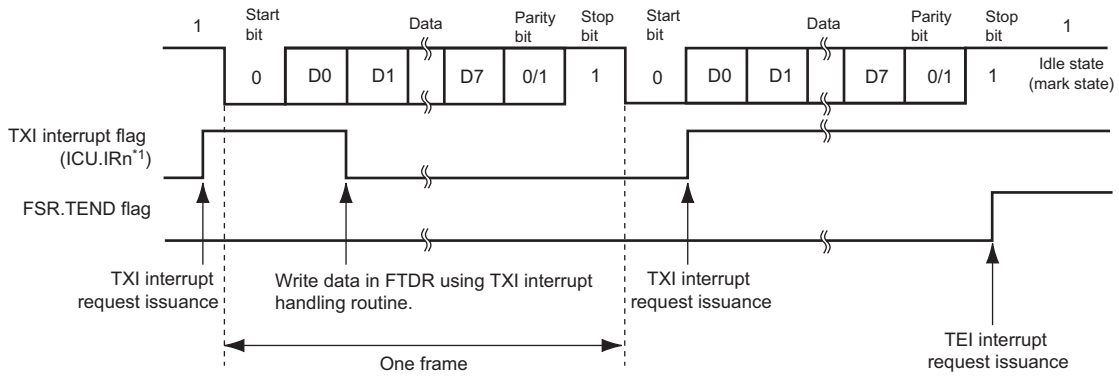
Note 1. The SPB2DT bit of the SPTR register indicates the input status of the RXD pin. Writings to the SPB2DT and SPB2IO bits of thsi register in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written to the SPB2DT bit. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

Figure 17.4 Sample Flowchart for Transmitting Serial Data in Asynchronous Mode

In asynchronous mode, the SCIFA performs serial transmission as described below.

1. When data is written into the transmit FIFO data register (FTDR) by the TXI interrupt processing routine, the SCIFA transfers the data from the FTDR register to the transmit shift register (TSR) and starts transmission. Confirm that the TDFE flag in the serial status register (FSR) is set to 1 before writing transmit data to the FTDR register. The number of data bytes that can be written is “16 minus the number of non-transmitted data units”.
2. When data is transferred from the FTDR register to the TSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the FTDR register. When the number of transmit data bytes in the FTDR register becomes equal to or less than the transmission trigger number specified in the FIFO control register (FCR) or FIFO trigger control register (FTCR), the TDFE flag is set. If the TIE bit in the serial control register (SR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated. The serial transmit data is output from the TXD pin in the following order.
   (a) Start bit: One-bit 0 is output.
   (b) Transmit data: 8- or 7-bit data is output in LSB-first order (when LSB-first transfer is selected).
   (c) Parity bit: One parity bit (even or odd parity) is output. (A format in which a parity bit is not output can also be selected.)
   (d) Stop bit(s): One or two 1 bits (stop bits) are output.
   (e) Mark state: 1 is output continuously until the start bit that starts the next transmission is output.
3. The SCIFA checks the transmit data of the FTDR register at the timing for sending the stop bit. If data is present, the data is transferred from the FTDR register to the TSR register, the stop bit is output, and then serial transmission of the next frame is started. If there is no data to be transmitted, the TEND flag in the FSR register is set to 1, the stop bit is output, and then the SCIFA enters the mark state (high level) in which 1 is output continuously.

Figure 17.5 shows an example of the operation for transmission in asynchronous mode.



Note 1. For the corresponding interrupt number, see section 7, Interrupt Controller.

Figure 17.5 Example of Transmit Operation in Asynchronous Mode
(8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

4. When modem control is enabled, transmission can be stopped/resumed by the input level to the CTS# pin. When a high level is input to the CTS# pin during transmission, the SCIFA enters the mark state (high level) after completion of one-frame data transmission. When a low level is input to the CTS# pin, output of the next data to be transmitted begins with a start bit. Figure 17.6 shows an example of the operation for transmission when using the modem control function.

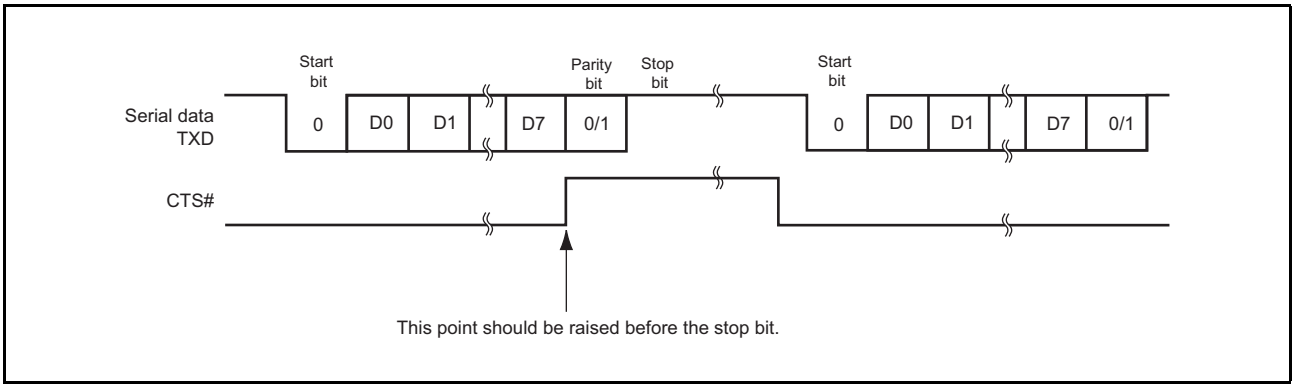


Figure 17.6 Example of Transmit Operation in Asynchronous Mode Using Modem Control Function (CTS#)

- Receiving Serial Data (in Asynchronous Mode)

Figure 17.7and Figure 17.8 show sample flowcharts for serial reception in asynchronous mode.
Follow the procedure given below for serial data reception after enabling the SCIFA for reception.

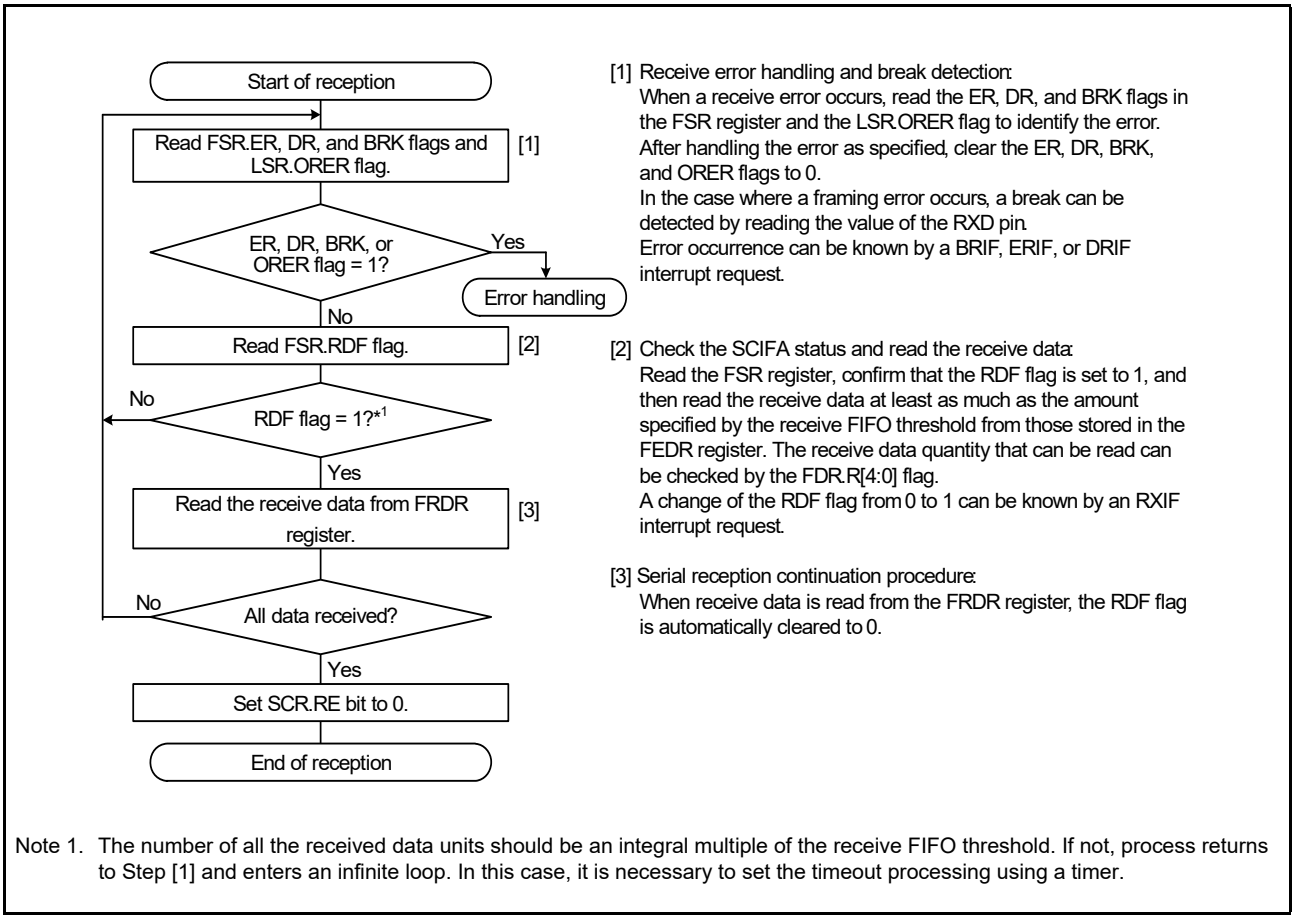
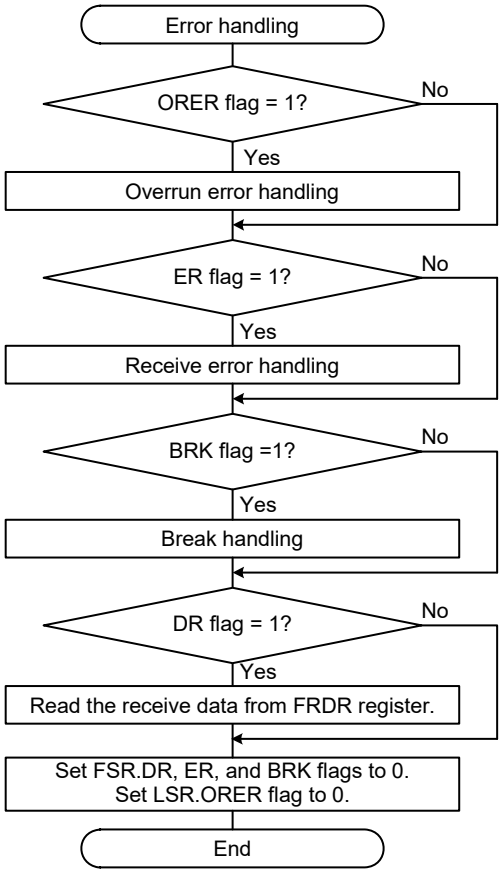


Figure 17.7 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (1)



- Whether a framing error or parity error has occurred in the received data that is read from the FRDR register can be checked with the FER and PER flags in the FSR register.
- When a break signal is received, the received data is not transferred to the FRDR register while the BRK flag is set to 1. However, note that the last data in the FRDR register is 00h and the break data in which a framing error occurred is stored.

Figure 17.8 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (2)

In asynchronous mode, the SCIFA performs serial reception as described below.

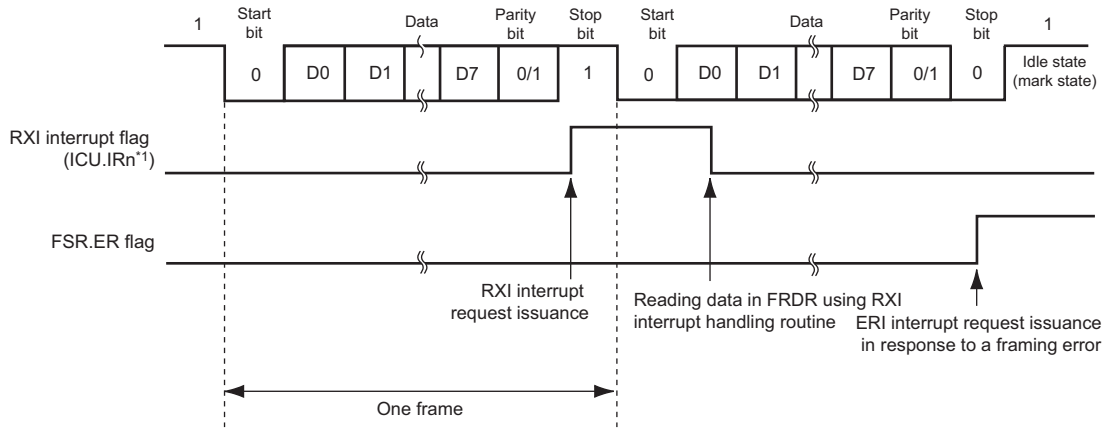
1. The SCIFA monitors the communication line, and if a 0 start bit is detected, it performs internal synchronization to start reception.
2. The received data is stored into the RSR register in LSB-to-MSB order (when LSB-first transfer is selected).
3. The parity bit and stop bit are received.
   After receiving these bits, the SCIFA carries out the following checks.
   (a) Stop bit check: The SCIFA checks whether the stop bit is 1. If there are two stop bits, only the first is checked.
   (b) The SCIFA checks whether receive data can be transferred from the receive shift register (RSR) to the receive FIFO data register (FRDR).
   (c) Parity bit check: The SCIFA checks whether the parity bit is an expected value.
   (d) Overrun error check: The SCIFA checks whether the ORER flag is 0, indicating that the overrun error has not occurred.
   (e) Break check: The SCIFA checks whether the BRK flag is 0, indicating that the break state is not set.

If all the above checks are passed, the receive data is stored in the FRDR register.
Note: When a parity error or a framing error occurs, reception is not suspended.

4. When receive data units equaling or exceeding the specified reception trigger number are stored in the receive FIFO data register (FRDR) and the RDF flag is changed to 1, a receive FIFO data full interrupt (RXI) request is generated while the RIE bit in the SCR register is set to 1. When the quantity of data in the FRDR register falls below the specified reception trigger number and the RIE bit in the SCR register is set to 1, a receive data ready interrupt (DRI) request is generated if no next data is received after the elapse of 15 ETUs*1 from the last stop bit (the DR flag in the FSR register is 1). When the ER flag in the FSR register is changed to 1, a receive error interrupt (ERI) request is generated while the RIE or REIE bit in the SCR register is set to 1. When the BRK or ORER flag is changed to 1 in the FSR register, a break reception interrupt (BRI) request is generated while the RIE or REIE bit in the SCR register is set to 1.

Note 1. It is equivalent to 1 and half frames of 8-bit format with one stop bit (ETU: Element Time Unit).

Figure 17.9 shows an example of the operation for reception in asynchronous mode.



Note 1. For the corresponding interrupt number, see section 7, Interrupt Controller.

Figure 17.9 Example of SCIFA Receive Operation in Asynchronous Mode
(8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

5. When modem control is enabled, the RTS# signal that indicates the FRDR register has space is output. When the RTS# pin is at low level, reception is possible. The RTS# pin being at the high level indicates that the number of entries in the FRDR register is equal to or greater than the threshold for output of the active level of the RTS# signal and that the transmission of further data needs to be suspended until the FRDR register has enough space. Figure 17.10 shows an example of the operation for reception in asynchronous mode when using the modem control function.

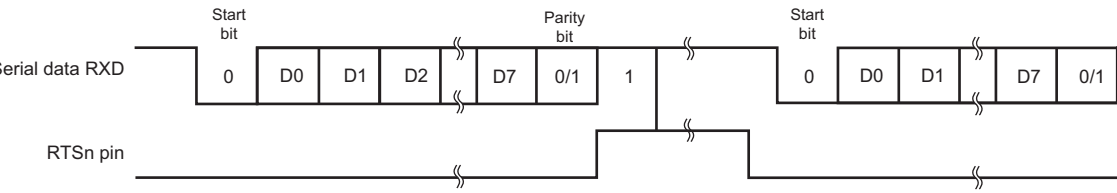
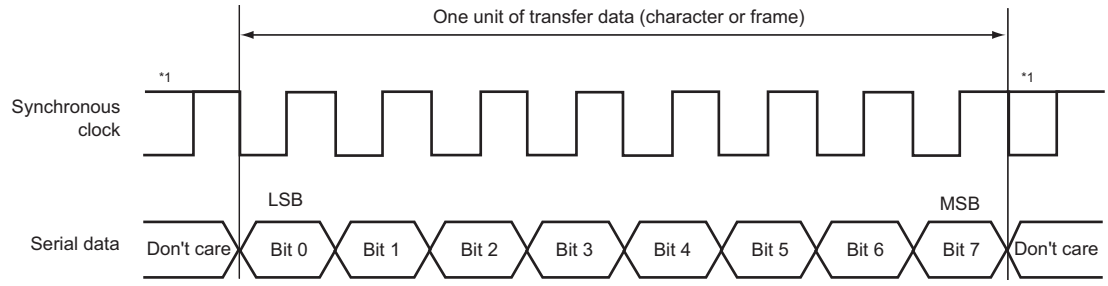


Figure 17.10 Example of SCIFA Receive Operation in Asynchronous Mode Using Model Control Function (RTS#)

17.3.3 Operation in Clock Synchronous Mode

In clock synchronous mode, the SCIFA transmits and receives data in synchronization with clock pulses. This mode is suitable for high-speed serial communication.
Full-duplex communication is possible because the SCIFA transmitter and receiver are independent and share the same clock. Since the transmitter and the receiver have 16-stage FIFO buffers, respectively, continuous transmission or reception is possible by reading or writing data while transmission or reception is in progress.
Figure 17.11 shows the general format in clock synchronous serial communication.



Note 1. High level except in continuous transmission/reception

Figure 17.11 Data Format in Clock Synchronous Communication (when LSB-First Transfer is Selected)

In clock synchronous serial communication, each data bit is output on the communication line from one falling edge of the synchronous clock to the next. Data is guaranteed valid at the rising edge of the synchronous clock.
In each character, the serial data bits are transmitted in order from the LSB (first) to the MSB (last). After output of the MSB, the communication line remains in the state of the MSB (when LSB-first transfer is selected).
In clock synchronous mode, the SCIFA receives data by synchronizing with the rising edge of the synchronous clock.

(1) Transmit/Receive Formats

The data length is fixed at eight bits.
No parity bit can be added.

(2) Clock

An internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the SCIFA transmit/receive clock according to the settings of the CM bit in the serial mode register (SMR) and the CKE[1:0] bits in the serial control register (SCR).
When the SCIFA operates on an internal clock, it outputs the synchronous clock signal at the SCK pin. Eight synchronous clock pulses are output per transmitted or received character. Unless the SCIFA is transmitting or receiving, the synchronous clock signal remains in the high state. When the SCIFA only receives data on an internal clock, the internal clock signal outputs while the RE bit in the SCR register is 1 until the number of data units in the receive FIFO reaches the specified reception trigger number.

(3) Transmitting and Receiving Data

- SCIFA Initialization (in Clock Synchronous Mode)

Before transmitting or receiving, clear the TE and RE bits to 0 in the serial control register (SCR), and then initialize the SCIFA by performing the following procedure.
Similarly, before changing the mode or communication format, clear the TE and RE bits to 0, and then change it by performing the following procedure. Clearing TE to 0 initializes the transmit shift register (TSR). Clearing RE to 0, however, does not initialize the RDF, PER, FER, and ORER flags and the receive FIFO data register (FRDR), which retain their previous contents.

Figure 17.12 shows a sample flowchart for initializing the SCIFA in clock synchronous mode.

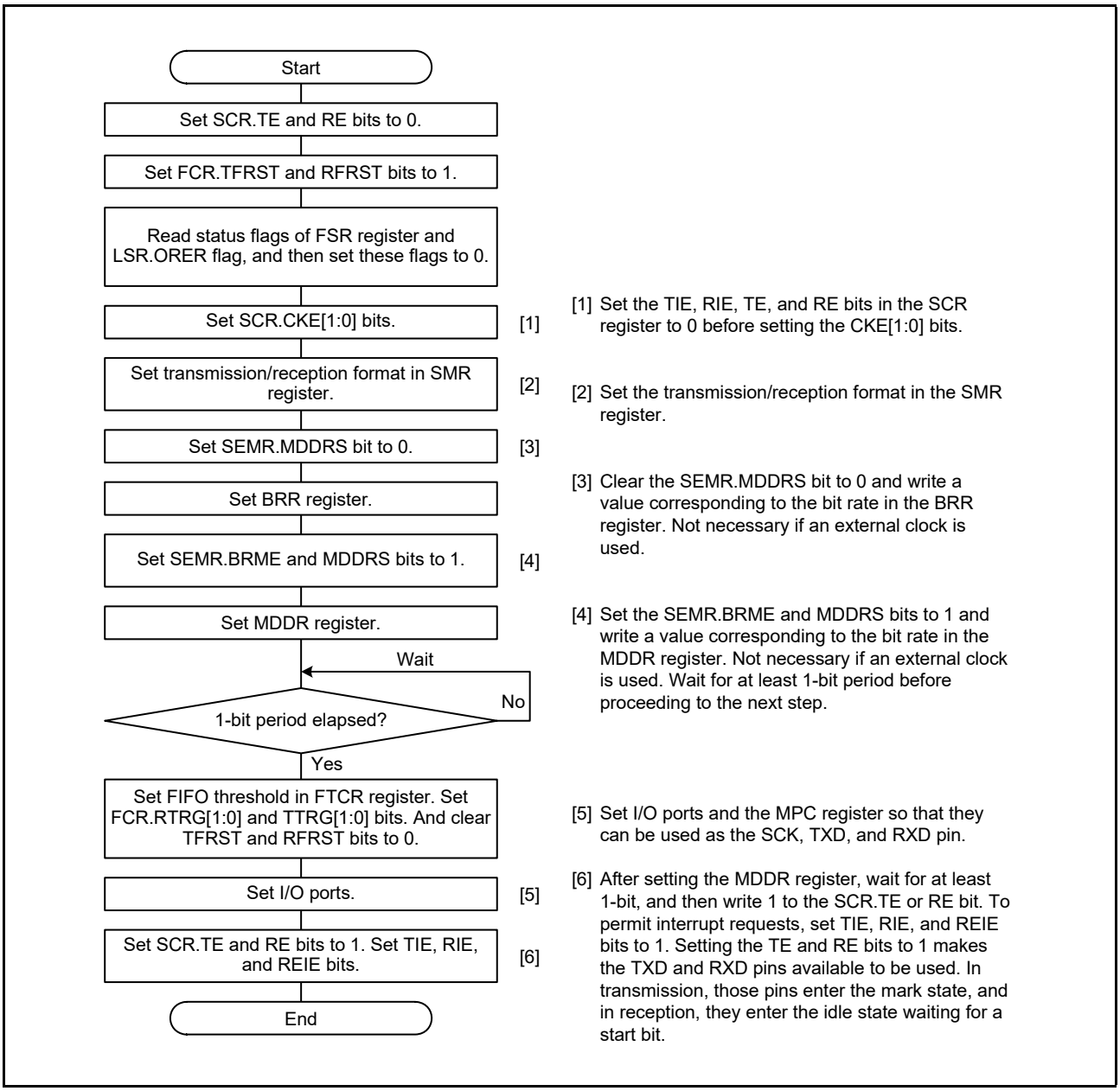


Figure 17.12 Sample Flowchart for SCIFA Initialization in Clock Synchronous Mode

- Transmitting Serial Data (in Clock Synchronous Mode)

Figure 17.13 shows a sample flowchart for transmitting serial data in clock synchronous mode.
Follow the procedure given below for serial data transmission after enabling the SCIFA for transmission.

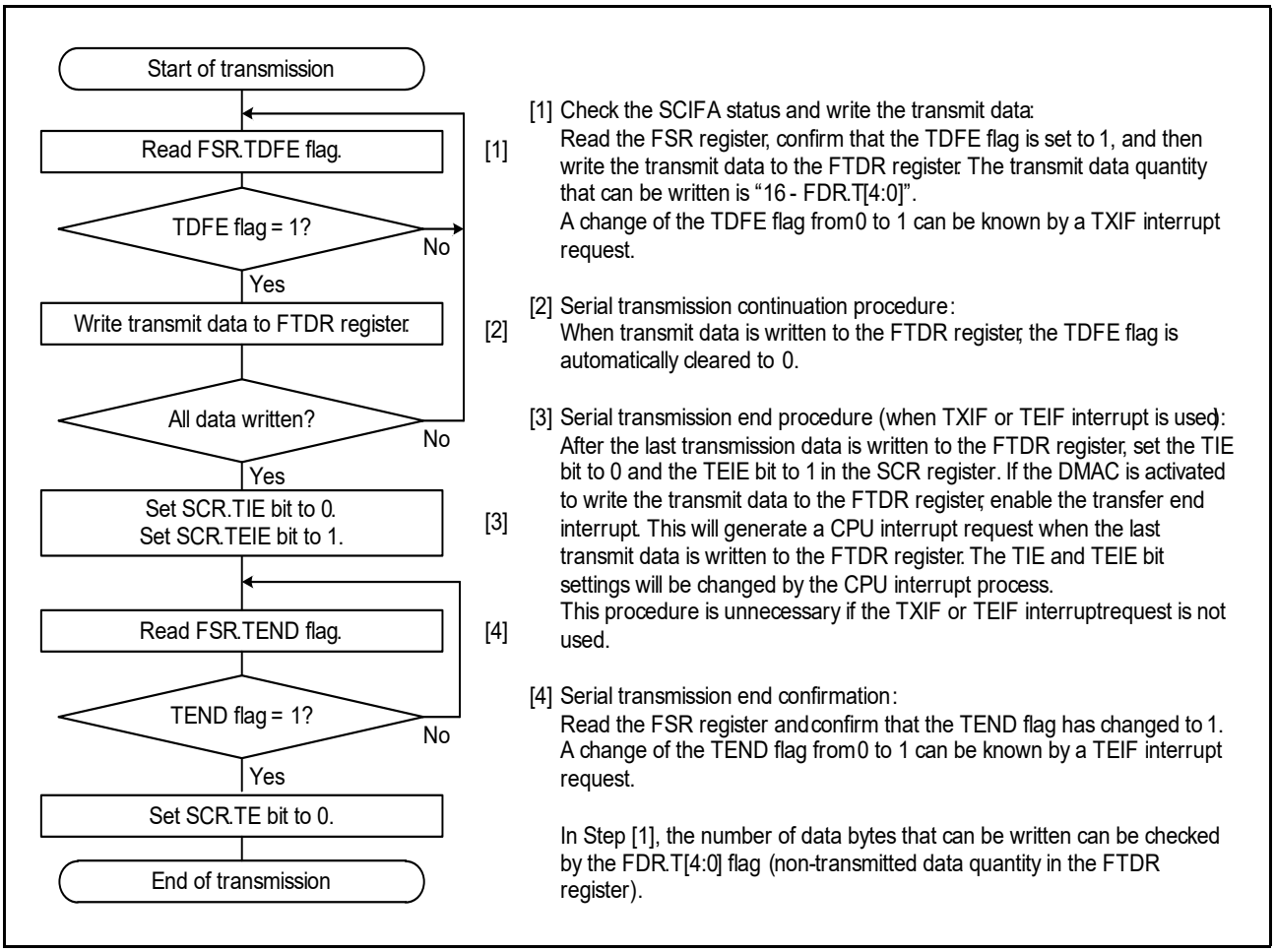
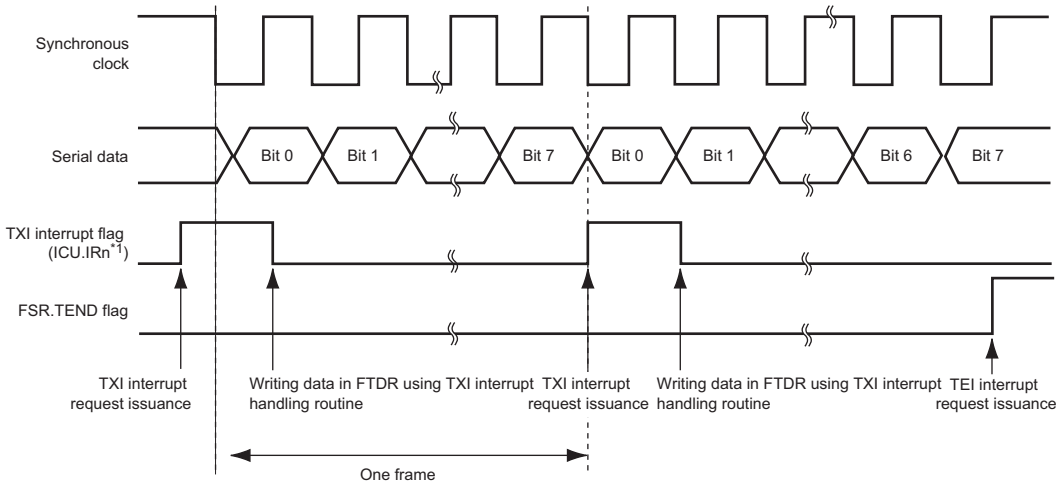


Figure 17.13 Sample Flowchart for Transmitting Serial Data in Clock Synchronous Mode

In clock synchronous mode, the SCIFA performs serial transmission as described below.

1. When data is written into the transmit FIFO data register (FTDR) by the TXI interrupt processing routine, the SCIFA transfers the data from the FTDR register to the transmit shift register (TSR) and starts transmission. Confirm that the TDFE flag in the serial status register (FSR) is set to 1 before writing transmit data to the FTDR register. The number of data bytes that can be written is “16 minus the specified number of non-transmitted data units”.
2. When data is transferred from the FTDR register to the TSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the FTDR register. When the number of transmit data bytes in the FTDR register becomes equal to or less than the transmission trigger number set in the FIFO control register (FCR) or FIFO trigger control register (FTCR), the TDFE flag in the FSR register is set. If the TIE bit in the serial control register (SCR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated.
   If clock output mode is selected, the SCIFA outputs eight synchronous clock pulses. If an external clock source is selected, the SCIFA outputs data in synchronization with the input clock. Data is output from the TXD pin in order from the LSB (b0) to the MSB (b7) (when LSB-first transfer is selected).
3. The SCIFA checks the transmit data of the FTDR register at the timing for sending the MSB (bit 7). If data is present, the data is transferred from the FTDR register to the TSR register, and then serial transmission of the next frame is started. If there is no data, the TXD pin holds the output level of the last data after the TEND flag in the FSR register is set to 1 and the MSB (bit 7) is output.
4. After the end of serial transmission, the SCK pin is held in the high state.

Figure 17.14 shows an example of SCIFA transmit operation in clock synchronous mode.

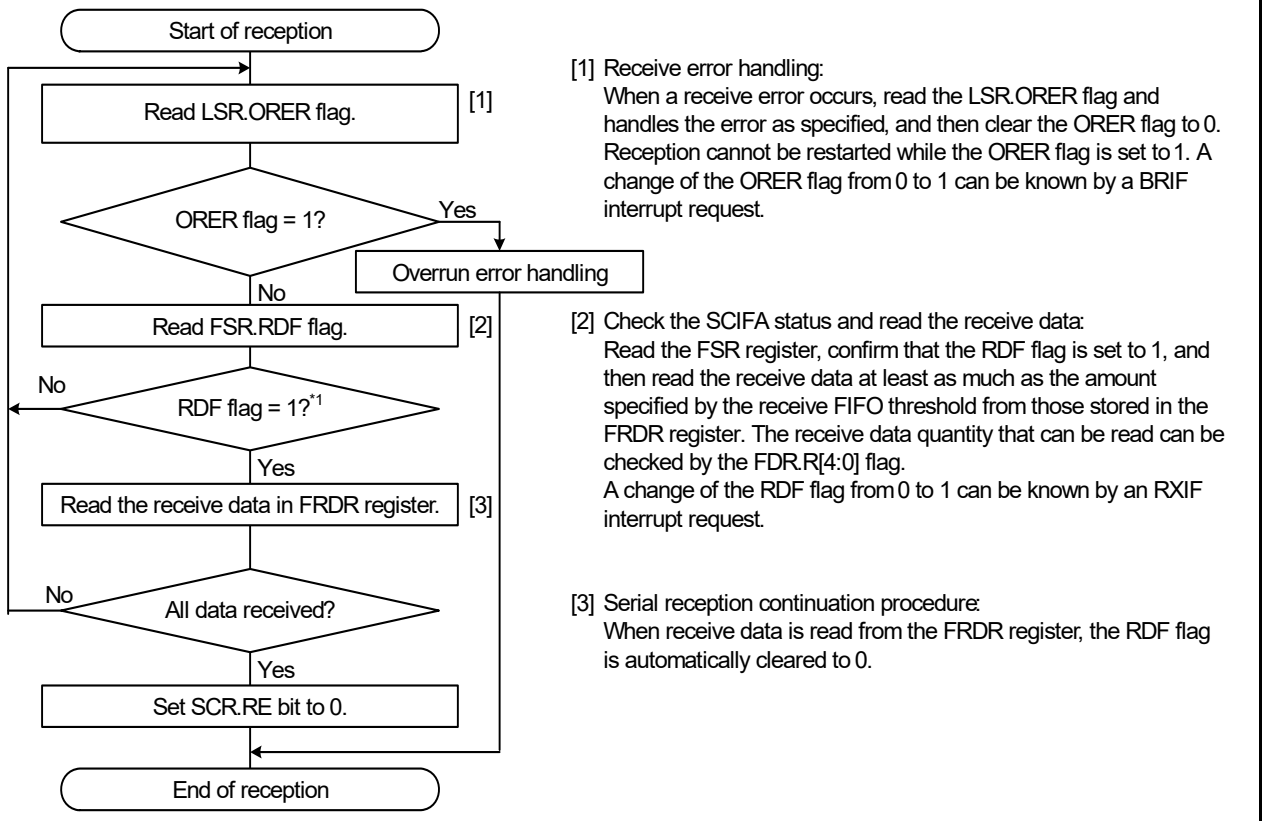


Note: For the corresponding interrupt number, see section 12, Interrupt Controller (ICUA).

Figure 17.14 Example of SCIFA Transmit Operation in Clock Synchronous Mode (when LSB-First Transfer is Selected)

- Receiving Serial Data (in Clock Synchronous Mode)

Figure 17.15 and Figure 17.16 show sample flowcharts for receiving serial data in clock synchronous mode. Follow the procedure given below for serial data reception after enabling the SCIFA for reception. When switching from asynchronous mode to clock synchronous mode without SCIFA initialization, make sure that the ORER, PER, and FER flags in the line status register (LSR) are cleared to 0.



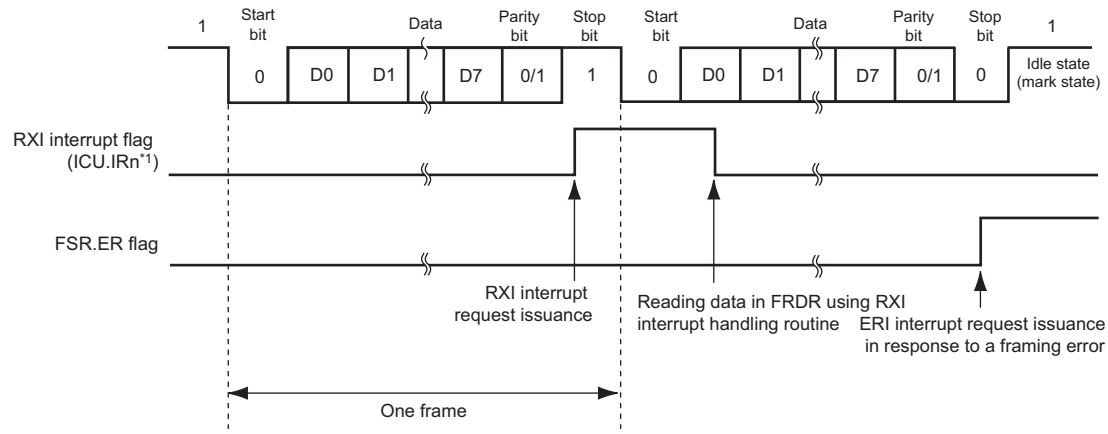
Note 1. The number of all the received data units should be an integral multiple of the receive FIFO threshold. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 17.15 Sample Flowchart for Receiving Serial Data in Clock Synchronous Mode

In clock synchronous mode, the SCIFA performs serial reception as described below.

1. The SCIFA synchronizes with the synchronous clock input or output and starts reception.
2. Receive data is stored into the receive shift register (RSR) in order from the LSB to the MSB (when LSB-first transfer is selected). After receiving the data, the SCIFA checks whether the receive data can be transferred from the RSR register to the FRDR register. If data can be transferred, the SCIFA stores the received data in the FRDR register. If an overrun error is detected during the error check, further reception is not performed.
3. After the received data units equaling or exceeding the specified reception trigger number are stored in the FRDR register and the RDF flag is set to 1, a receive-data-full interrupt (RXI) request is generated when the RIE bit in the serial control register (SCR) is set to 1. When the ORER flag in the line status register (LSR) is set to 1 and the RIE or REIE bit in the SCR register is also set to 1, a break interrupt (BRI) request is generated.

Figure 17.16 shows an example of SCIFA receive operation in clock synchronous mode.



Note 1. For the corresponding interrupt number, see section 7, Interrupt Controller.

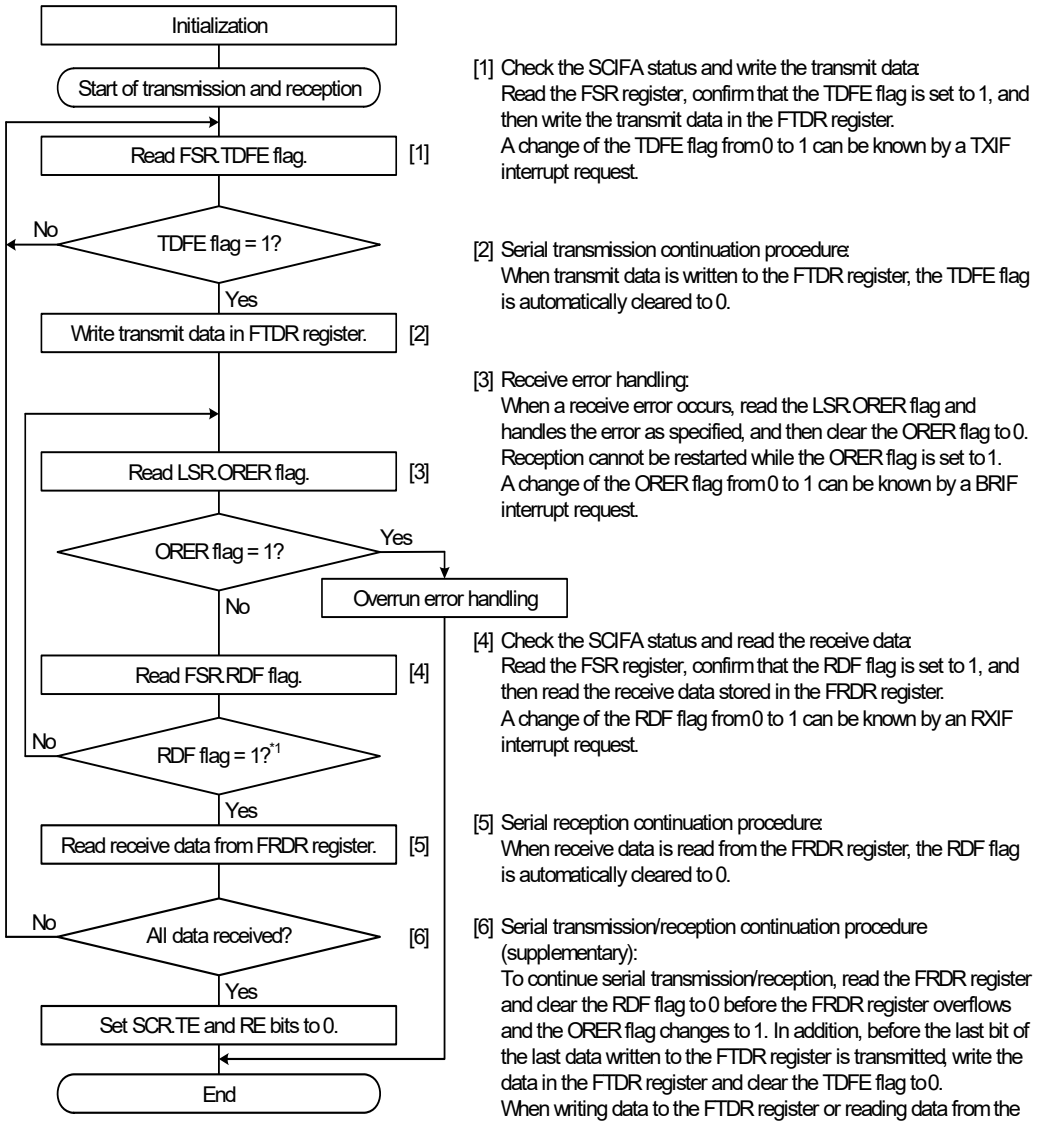
Figure 17.16 Example of SCIFA Receive Operation (when LSB-First Transfer is Selected)

- Transmitting and Receiving Serial Data Simultaneously (in Clock Synchronous Mode)

Figure 17.17 shows a sample flowchart for transmitting and receiving serial data simultaneously in clock synchronous mode.

In simultaneous transmission/reception of serial data, number of receive data = number of transmit data = number of transmit data to be written to the FTDR register.

Follow the procedure given below for the simultaneous transmission/reception of serial data, after enabling the SCIFA for transmission/reception.



Note: When switching the operation from transmission or reception to the simultaneous transmission and reception, clear the TE and RE bits to 0 and then simultaneously set these bits to 1.

Note 1. The number of all the received data units should be an integral multiple of the receive FIFO threshold. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 17.17 Sample Flowchart for Simultaneous Transmitting/Receiving Serial Data in Clock Synchronous Mode

17.4 Bit Rate Modulation

Using the bit rate modulation, the bit rate can be corrected by skipping the specified number of clock pulses input to the baud rate generator. To correct the bit rate, only the number of clock pulses specified in the MDDR register are enabled among 256 internal clock pulses specified by the CKS1 and CKS0 bits in the SMR register in a way that forms average intervals.

Figure 17.18 shows an example where P1φ is selected by the CKS[1:0] bits in SMR and BRR and MDDR are set to 0 and 160, respectively, in asynchronous mode. In this example, the cycle of the base clock is evenly corrected (256/160) and the bit rate is also corrected (160/256). Note that skipping an internal clock causes bias and expansion or contraction is generated in the pulse width of the base clock.

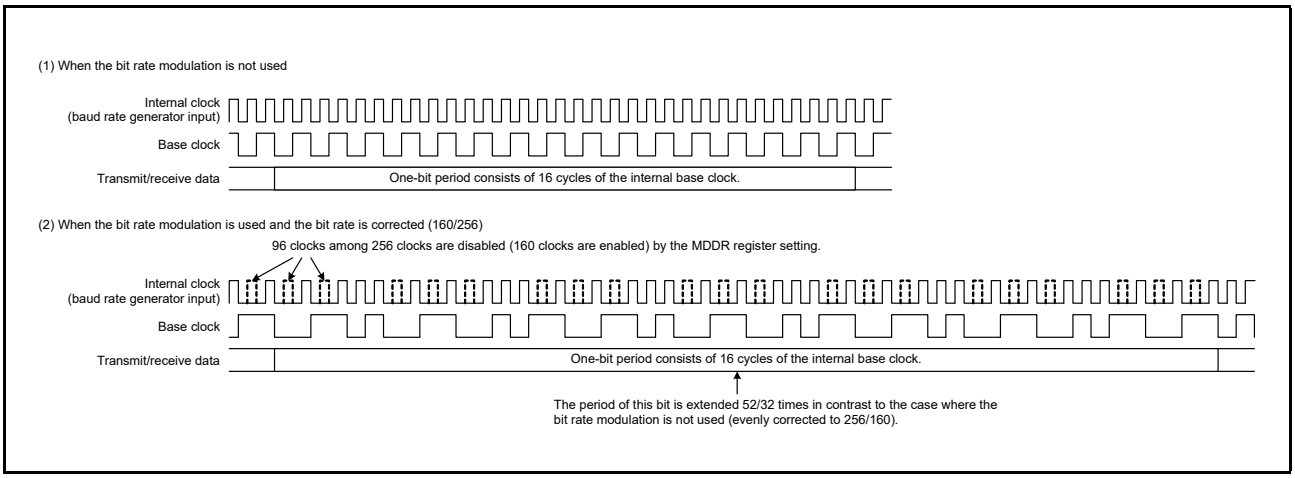


Figure 17.18 Example of Internal Base Clock when Bit Modulation is Used

17.5 Interrupt Sources

The SCIFA has six interrupt sources: transmit-FIFO-data-empty (TXI), receive-error (ERI), receive-FIFO-data-full (RXI), break (BRI), transmit-end (TEI), receive-data-ready (DRI). The TEI, DRI, ERI, and BRI interrupts share the same vector number.

Table 17.18 shows the interrupt sources and priority. The interrupt sources can be enabled or disabled using the TIE, RIE, REIE, TEIE bits in the SCR register and are separately input to the interrupt controller.

When the quantity of transmit data written in the FTDR register as a result of transmission is equal to or less than the specified transmission trigger number, the TDFE flag in the serial status register (FSR) is set to 1 and a TXI interrupt request is generated.

When the data units equaling or exceeding the specified transmission trigger number are stored in the receive FIFO register (FRDR) and the RDF flag in the FSR register is set to 1, a receive data full interrupt (RXI) request is generated.

When the quantity of received data in the FRDR register is below the specified reception trigger number and no next data has been received yet even after the period of 15 ETUs elapsed*1 from the last stop bit, the DR flag in the FSR register is set to 1 and a receive data ready interrupt (DRI) request is generated. In clock synchronous mode, a DRI interrupt request is not generated.

When the BRK flag in the FSR register or the ORER flag in the LSR register is set to 1, a BRI interrupt request is issued.

When the ER flag in the FSR register is set to 1, an ERI interrupt request is issued.

When the TEND flag in the FSR register is set to 1, a TEI interrupt request is issued.

When the RIE bit is cleared to 0 and the REIE bit in the SCR register is set to 1, an ERI and a BRI interrupt requests are issued but an RXI interrupt request is not.

An TXI interrupt indicates that transmit data can be written and an RXI interrupt indicates that receive data is stored in the FRDR register.

Note 1. It is equivalent to 1 and half frames of 8-bit format with one stop bit (ETU: Element Time Unit).

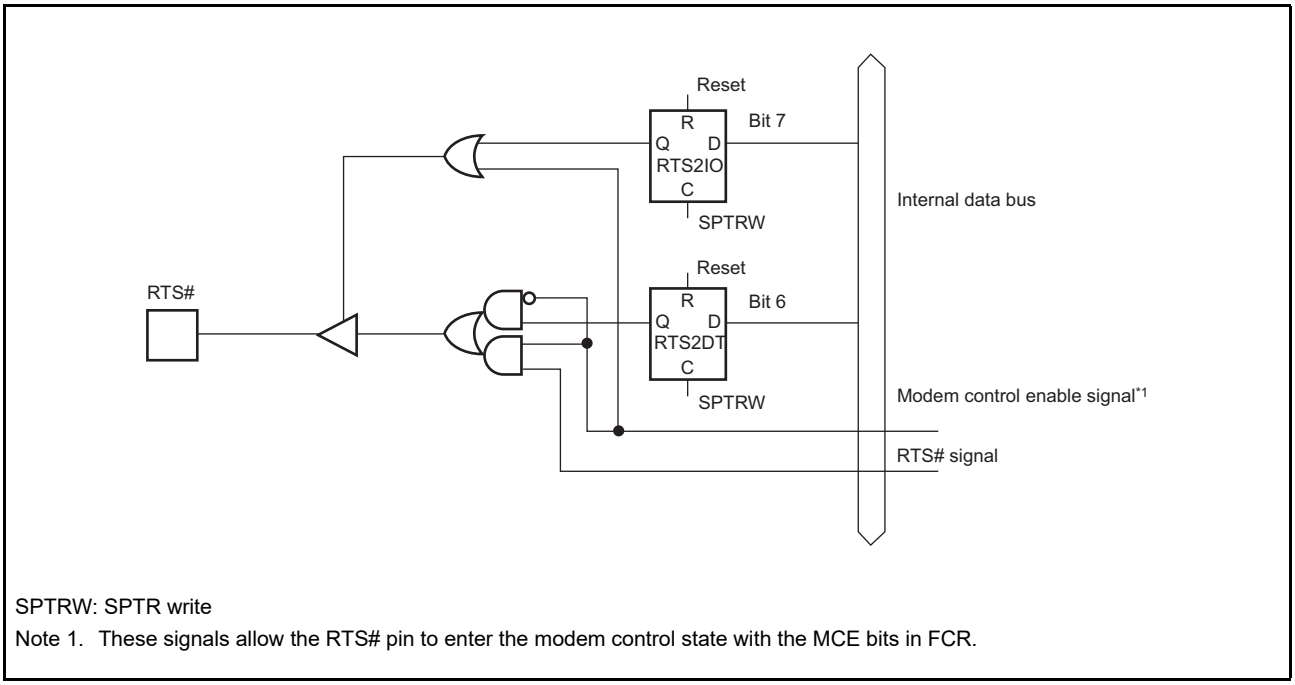
Table 17.18 SCIFA Interrupt Sources

Name	Level/ Edge	Interrupt Source	Interrupt Enable BIt	DMAC Activation	Priority
BRI	Level	Interrupt caused by break (BRK) or overrun (ORER).	RIE or REIE	Impossible	High
ERI	Level	Interrupt caused by framing or parity (ER).	RIE or REIE	Impossible	↕
RXI	Level	Interrupt caused by receive FIFO data full (RDF).	RIE	Possible	
TXI	Level	Interrupt caused by transmit FIFO data empty (TDFE).	TIE	Possible	
TEI	Level	Interrupt caused by transmit end (TEND).	TEIE	Impossible	
DRI	Level	Interrupt caused by receive data ready (DR).	RIE	Impossible	Low

Note: The TEI and DRI interrupts share the same vector number.
The ERI and BRI interrupts share the same vector number.
If CPU processing is used, clear the flag after the block transfer. If the DMAC is activated, access to the flags is prohibited.

17.6 Serial Port Register (SPTR) and SCIFA-Related Pins

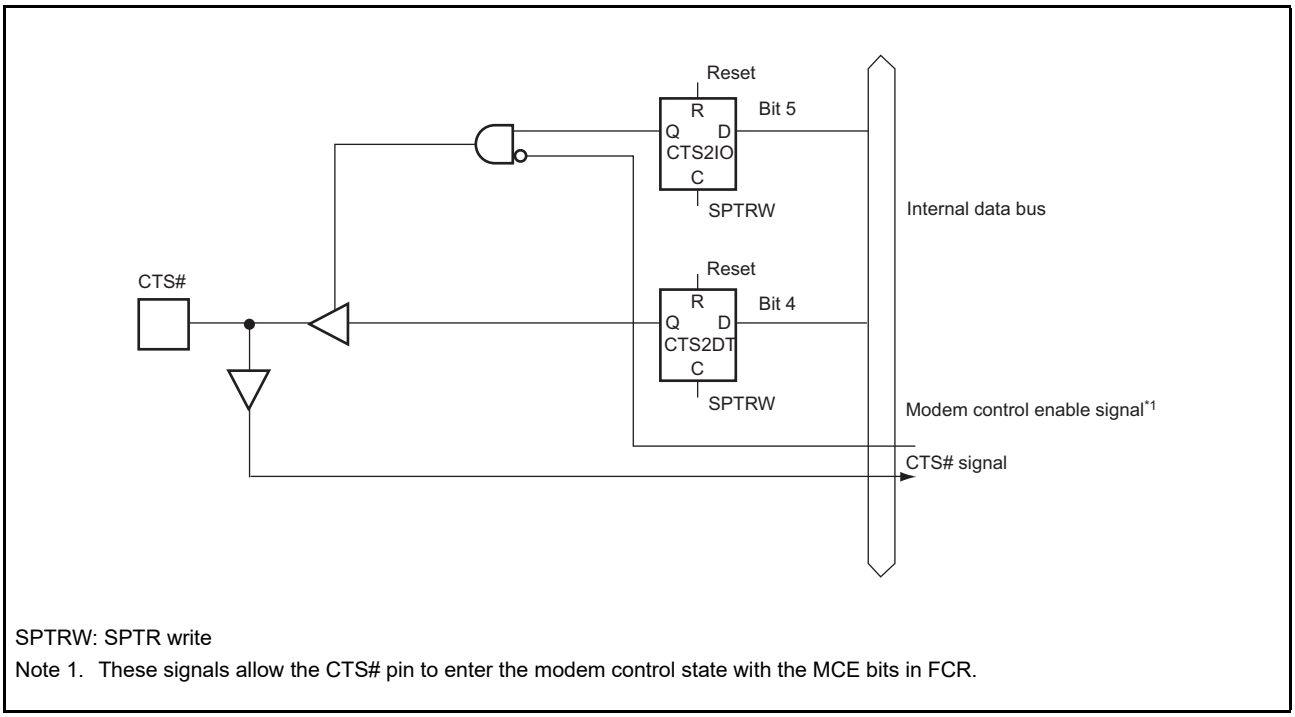
Figure 17.19 to Figure 17.22 show the relationships between the SPTR register and the SCIFA-related pins.



SPTRW: SPTR write

Note 1. These signals allow the RTS# pin to enter the modem control state with the MCE bits in FCR.

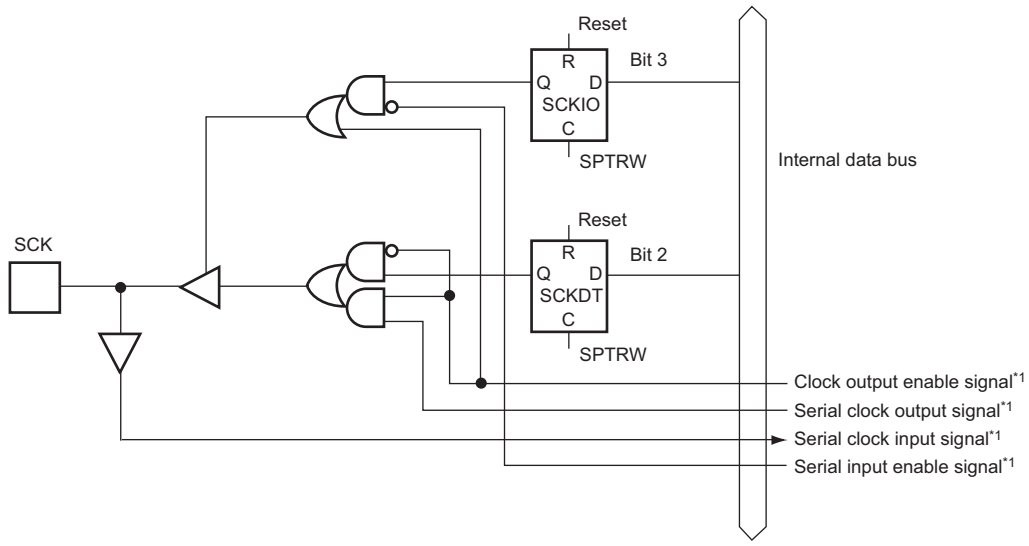
Figure 17.19 RTS2IO Bit and RTS2DT Bit in the SPTR Register, and RTS# Pin



SPTRW: SPTR write

Note 1. These signals allow the CTS# pin to enter the modem control state with the MCE bits in FCR.

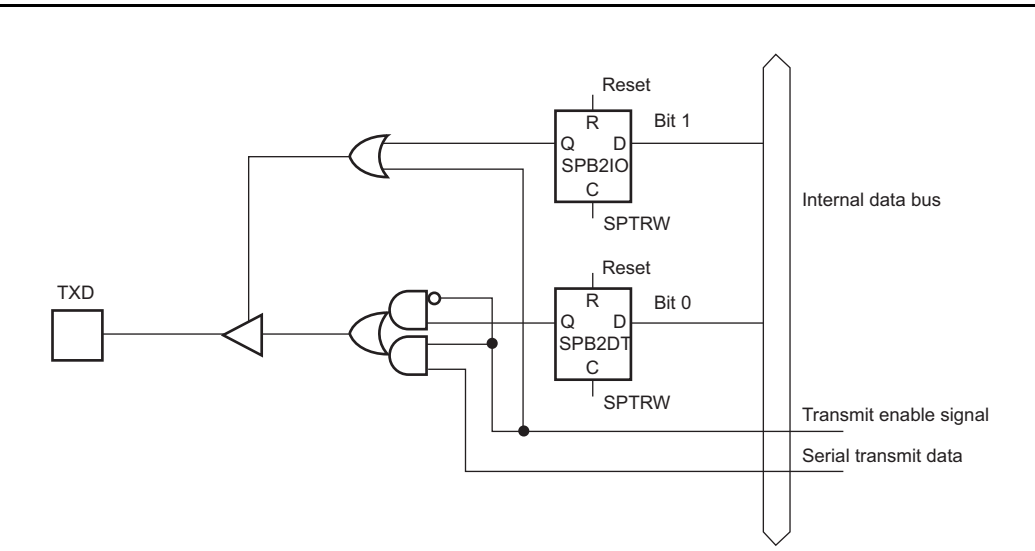
Figure 17.20 CTS2IO Bit and CTS2DT Bit in the SPTR Register, and CTS# Pin



SPTRW: SPTR write

Note 1. These signals control the SCK pin according to the settings of the CM bit in SMR and CKE[1:0] bits in SCR.

Figure 17.21 SCKIO Bit and SCKDT Bit in the SPTR Register, and SCK Pin



SPTRW: SPTR write

Figure 17.22 SPB2IO Bit and SPB2DT Bit in the SPTR Register, and TXD Pin

17.7 Noise Cancellation

Figure 17.23 shows the configuration of the noise filter used for noise cancellation. The noise filter consists of a two-stage flip-flop circuits and a match detection circuit. When the input signals of the noise filter and the output signals of the two-stage flip-flop circuits completely match, the matched level is conveyed as an internal signal. Unless otherwise matched, the previous value is retained. (When the levels sampled on three consecutive cycles of the sampling clock of the noise filter match, the signal is considered valid. If three consecutive sampled values do not match, the signal is considered to be noise rather than a received signal).

In asynchronous mode, the noise cancellation can be applied to the receive signal input to the RXDn pin. The receive level of the RXDn pin is taken in the flip-flop circuit of the noise filter on the base clock (the clock with a frequency 16 or 8 times the transfer rate*1).

If the base clock is stopped once with the noise filter enabled and then the base clock input is restarted again, the noise filter operation resumes from the state where the clock was stopped. When SCR.RE is set to 0 during input of the base clock, the noise filter outputs 0 as the internal RxDn signal. The internal match detector continues operating even while operations for reception are stopped, and the result from the last time previous consecutive samples matched is output at the same time as operations for reception are resumed.

Note 1. A frequency 16 times bit rate when the SEMR.ABCS0 bit and the SEMR.BGDM bit are both 0, and a frequency 8 times bit rate when either the SEMR.ABCS0 bit or the SEMR.BGDM bit is 1.

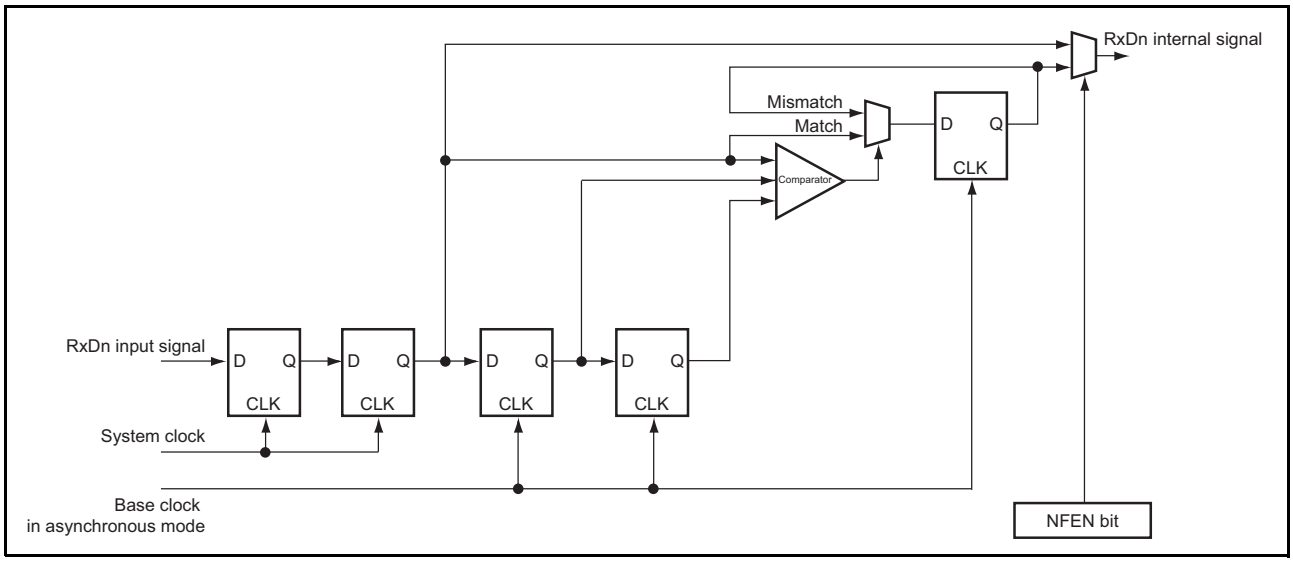


Figure 17.23 Block Diagram of Digital Noise Filter Circuit

17.8 Usage Notes

The following is the notes on using the SCIFA.

17.8.1 FTDR Register Writing and TDFE Flag

The TDFE flag in the serial status register (FSR) is set when the number of transmit data bytes written in the transmit FIFO data register (FTDR) has fallen below the transmission trigger number set by bits TTRG[1:0] in the FIFO control register (FCR) or bits TFTC[4:0] in the FIFO trigger control register (FTCR). After the TDFE flag is set, transmit data up to the number of empty bytes in the FTDR register can be written, allowing efficient continuous transmission.
However, if the number of data bytes written in the FTDR register is equal to or less than the specified transmission trigger number, the TDFE flag will be set to 1 again even after being read as 1 and cleared to 0.
The number of transmit data bytes in the FTDR register can be checked by the 8 higher-order bits of the FIFO data count register (FDR).

17.8.2 FRDR Register Reading and RDF Flag

The RDF flag in the serial status register (FSR) is set when the number of receive data bytes in the receive FIFO data register (FRDR) has become equal to or greater than the reception trigger number set by bits RTRG[1:0] in the FIFO control register (FCR) or bits RFTC[4:0] in the FIFO trigger control register (FTCR). After the RDF flag is set, receive data equivalent to the trigger number can be read from the FRDR register, allowing efficient continuous reception.
However, if the number of data bytes in the FRDR register exceeds the reception trigger number, the RDF flag will be set to 1 again even after being read as 1 and cleared to 0.
The number of receive data bytes in the FRDR register can be checked by the 8 lower-order bits of the FIFO data count register (FDR).

17.8.3 Break Detection and Processing

When a framing error (FER) is detected, a break signal can be detected by reading the RXD pin value directly. In a break, the input from the RXD pin becomes all low. Therefore, the FER flag in the serial status register (FSR) is set to 1 and the parity error flag (PER) may also be set to 1.
Upon detection of a break signal, the SCIFA stops the received data transfer to the FRDR register but continues the receive operation.

17.8.4 Writing to the SPTR Register

b6, b4, b2, and b0 of the SPTR register respectively indicate the input status of their corresponding pins. (See the description of each bit of section 17.2.12, Serial Port Register (SPTR) for details.)
Writings to these bits in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

17.8.5 Break Signal Transmission

The output signal from the TXD pin is determined by the SPB2IO bit and the SPB2DT bit in the serial port register (SPTR). The break signal can be sent by using these bits.

The TXD pin does not function as a transmit data output pin during the period from when the SCIFA is initialized to when the TE bit in the SCR register is set to 1 (transmission possible). The TXD pin status during this period is replaced by the SPB2DT bit value. Therefore, the SPB2IO and SPB2DT bits in the SPTR register must have been set to 1 (high output) at first (mark (high) status).

To transmit the break signal during serial transmission, set the SPB2IO bit in the SPTR register to 1, clear the SPB2DT bit to 0 (specify a low level), and then clear the TE bit in the SCR register to 0 (transmission stop). Clearing the TE bit to 0 initializes the transmitter regardless of the current transmission status, and outputs a low level from the TXD pin.

17.8.6 Receive Data Sampling Timing and Receive Margin in Asynchronous Mode

The SCIFA operates on a base clock with a frequency 16 times the transfer rate*1. In reception, the SCIFA internally latches the received data at the rising edge of the eighth base clock pulse*1. The timing is shown in Figure 17.24.

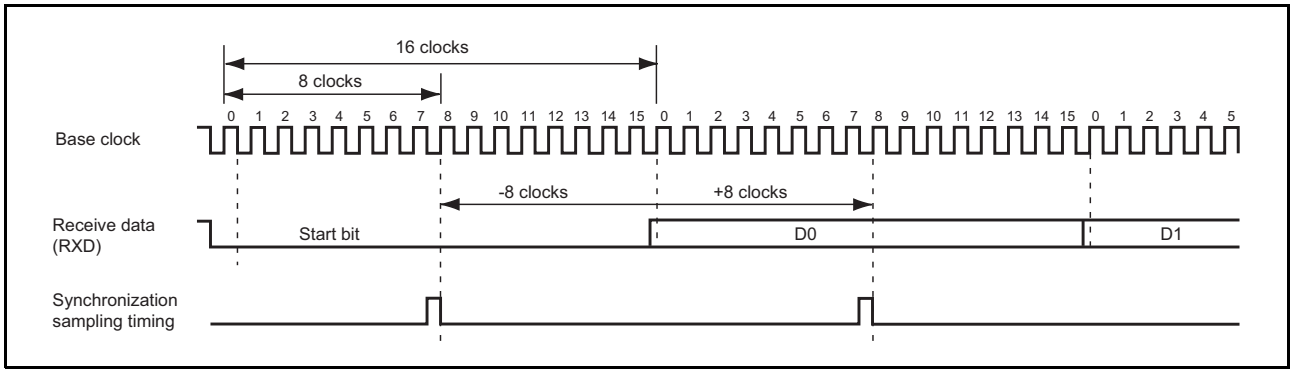


Figure 17.24 Receive Data Sampling Timing in Asynchronous Mode

Note 1. This is an example when the SEMR.ABCS0 bit is 0. When the ABCS0 bit is 1, a frequency of 8 times the bit rate is used as a base clock and receive data is sampled at the rising edge of the 4th pulse of the base clock. The receive margin in asynchronous mode can therefore be expressed as shown in equation 1.

The receive margin in asynchronous mode can therefore be expressed as shown in equation 1.

Equation 1:

$$M = \{(0.5 - \frac{1}{2N}) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F)\} \times 100[\%]$$

Where: M: Receive margin (%)
N: Ratio of clock frequency to bit rate (N = 16)
D: Clock duty (D = 0 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute deviation of clock frequency

From equation 1, if F = 0 and D = 0.5, the receive margin is 46.875%, as given by equation 2.

Equation 2:

When D = 0.5 and F = 0:
$M = (0.5 - 1/(2 \times 16)) \times 100\% = 46.875\%$

This is a theoretical value. A reasonable margin to allow in system designs is 20% to 30%.

17.8.7 Note on FER Flag and PER Flag in Serial Status Register (FSR)

The FER flag and PER flag in the serial status register (FSR) are status flags that apply to next entry to be read from the receive FIFO data register (FRDR). After the CPU or DMAC reads the receive FIFO data register, the flags of framing errors and parity errors in the receive data will be cleared. To check the received data for the states of framing errors and parity errors, only read the receive FIFO data register after reading the serial status register.

17.8.8 Notes on External Clock Input in Clock Synchronous Mode

Before setting the TE and RE bits in the serial control register (SCR) to 1, wait for four or more cycles of the peripheral operating clock after the external clock (SCK) is changed from 0 (low) to 1 (high). To input the external clock (SCK) (to start communication), wait for one or more cycles of the external clock after the TE and RE bits in the SCR register are set to 1.

17.8.9 Module Standby Mode Setting

SCIFA operation can be disabled or enabled using the standby control register. As the initial setting, the SCIFA operation is halted. Register access is enabled by clearing module standby mode. For details, refer to section 52, Power-Down Modes.

17.8.10 Notes on Operation for Reception when an Internal Clock is Selected in Clock Synchronous Mode

When an internal clock is selected as the clock for reception in clock-synchronous mode, if the number of data stored through the receive FIFO data register (FRDR) becomes equal to or greater than the specified reception trigger number, the RDF flag is set, the RXI interrupt request is generated and, at the same time, output of the synchronizing clock and reception of serial data are stopped. Once the number of data are again less than the specified reception trigger number, output of the synchronizing clock and the reception of serial data are restarted. In addition, if an internal clock is selected for reception in clock synchronous mode, the ORER flag is not set to 1 since no overrun occurs. Accordingly, overruns (indicated by the ORER flag) cannot be used as a BRI interrupt source.

18. Serial Communications Interface (SCIg)

This MCU has two independent serial communications interface (SCI) channels. The SCIs consist of the SCI0 and SCI1 modules, referred to in common as “SCIg modules”.
The SCIg modules (SCI0 and SCI1) can handle both asynchronous and clock synchronous serial communications. Asynchronous serial data communications can be carried out with standard asynchronous communications chips such as a Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communications Interface Adapter (ACIA). As an extended function in asynchronous communications mode, the SCI also supports smart card (IC card) interfaces conforming to ISO/IEC 7816-3 (standard for Identification Cards).
In this section, “P1φ” is used to refer to the peripheral clock 1C.

18.1 Overview

Table 18.1 lists the specifications of the SCIg modules. Figure 18.1 shows the block diagram of the SCI0 and SCI1 modules.

Table 18.1 SCIg Specifications

Item		Description
Serial communications modes		• Asynchronous • Clock synchronous • Smart card interface • IrDA (only for channel 0)
Transfer speed		Bit rate specifiable with the on-chip baud rate generator.
Full-duplex communications		Transmitter: Continuous transmission possible using double-buffer structure. Receiver: Continuous reception possible using double-buffer structure.
I/O pins		See Table 18.2.
Data transfer		Selectable as LSB first or MSB first transfer
Interrupt sources		Transmit end, transmit data empty, receive data full, and receive error
Low power consumption function		Module stop state can be set for each channel.
Asynchronous mode	Data length	7, 8, or 9 bits
	Transmission stop bit	1 or 2 bits
	Parity	Even parity, odd parity, or no parity
	Receive error detection	Parity, overrun, and framing errors
	Hardware flow control	CTSn# and RTSn# pins can be used in controlling transmission/reception.
	Start-bit detection	Low level or falling edge is selectable.
	Break detection	When a framing error occurs, a break can be detected by reading the RXDn pin level directly.
	Clock source	An internal or external clock can be selected.
	Double-speed mode	Baud rate generator double-speed mode is selectable.
	Multi-processor communications function	Serial communication among multiple processors
	Noise cancellation	The signal paths from input on the RXDn pins incorporate digital noise filters.
Clock synchronous mode	Data length	8 bits
	Receive error detection	Overrun error
	Hardware flow control	CTSn# and RTSn# pins can be used in controlling transmission/reception.
Smart card interface mode	Error processing	An error signal can be automatically transmitted when detecting a parity error during reception
		Data can be automatically retransmitted when receiving an error signal during transmission
	Data type	Both direct convention and inverse convention are supported.
Bit rate modulation function		Correction of outputs from the on-chip baud rate generator can reduce errors.

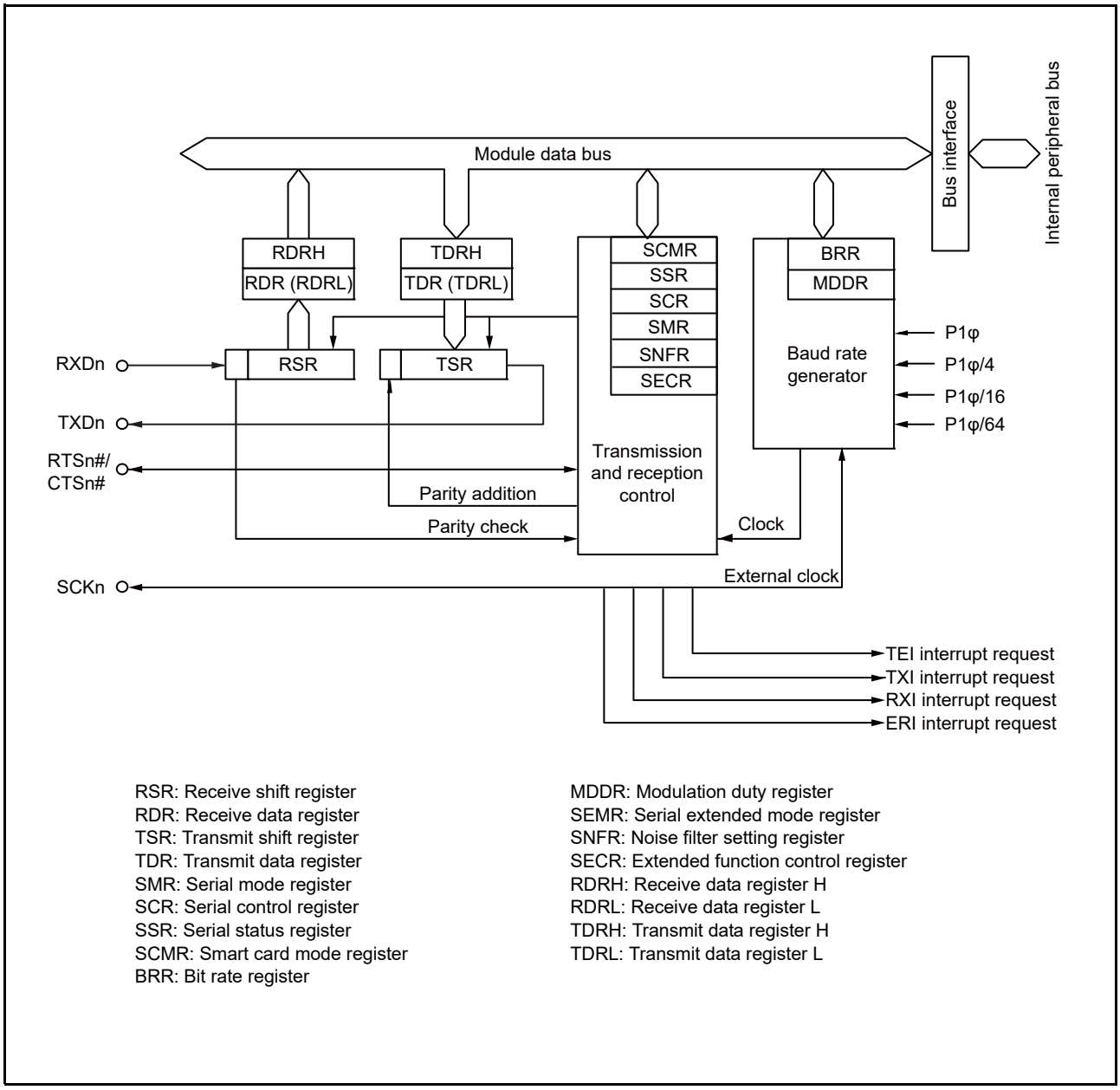


Figure 18.1 Block Diagram of SCIg (SCI0 and SCI1)

Table 18.2 lists the pin configuration of the SCIs for the individual modes.

Table 18.2 SCI Pin Configuration in Asynchronous Mode and Clock Synchronous Mode

Channel	Pin Name	I/O	Function
SCI0	SCI_SCK0	I/O	SCI0 clock input/output
	SCI_RXD0	Input	SCI0 receive data input
	SCI_TXD0	Output	SCI0 transmit data output
	SCI_CTS#0/RTS#0	I/O	SCI0 transfer start control input/output
SCI1	SCI_SCK1	I/O	SCI1 clock input/output
	SCI_RXD1	Input	SCI1 receive data input
	SCI_TXD1	Output	SCI1 transmit data output
	SCI_CTS#1/RTS#1	I/O	SCI1 transfer start control input/output

18.2 Register Descriptions

Table 18.3 Register configuration

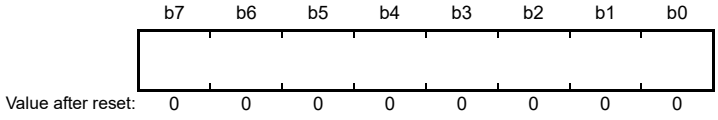
Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	Serial Mode Register	SMR	R/W	00h	H'E800B000	8
	Bit Rate Register	BRR	R/W	FFh	H'E800B001	8
	Serial Control Register	SCR	R/W	00h	H'E800B002	8
	Transmit Data Register	TDR	R/W	FFh	H'E800B003	8
	Serial Status Register	SSR	R/W*	84h	H'E800B004	8
	Receive Data Register	RDR	R	00h	H'E800B005	8
	Smart Card Mode Register	SCMR	R/W*	F2h	H'E800B006	8
	Serial Extended Mode Register	SEMR	R	00h	H'E800B007	8
	Noise Filter Setting Register	SNFR	R	00h	H'E800B008	8
	Extended Function Control Register	SECR	R/W	00h	H'E800B00D	8
	Transmit Data Register H	TDRH	R/W	FFh	H'E800B00E	8
	Transmit Data Register L	TDRL	R/W	FFh	H'E800B00F	8
	Transmit Data Register HL	TDRHL	R/W	FFh	H'E800B00E	16
	Receive Data Register H	RDRH	R	00h	H'E800B010	8
	Receive Data Register L	RDRL	R	00h	H'E800B011	8
	Receive Data Register HL	RDRHL	R	00h	H'E800B010	16
	Modulation Duty Register	MDDR	R/W	FFh	H'E800B012	8
1	Serial Mode Register	SMR	R/W	00h	H'E800B800	8
	Bit Rate Register	BRR	R/W	FFh	H'E800B801	8
	Serial Control Register	SCR	R/W	00h	H'E800B802	8
	Transmit Data Register	TDR	R/W	FFh	H'E800B803	8
	Serial Status Register	SSR	R/W*	84h	H'E800B804	8
	Receive Data Register	RDR	R	00h	H'E800B805	8
	Smart Card Mode Register	SCMR	R/W*	F2h	H'E800B806	8
	Serial Extended Mode Register	SEMR	R	00h	H'E800B807	8
	Noise Filter Setting Register	SNFR	R	00h	H'E800B808	8
	Extended Function Control Register	SECR	R/W	00h	H'E800B80D	8
	Transmit Data Register H	TDRH	R/W	FFh	H'E800B80E	8
	Transmit Data Register L	TDRL	R/W	FFh	H'E800B80F	8
	Transmit Data Register HL	TDRHL	R/W	FFh	H'E800B80E	16
	Receive Data Register H	RDRH	R	00h	H'E800B810	8
	Receive Data Register L	RDRL	R	00h	H'E800B811	8
	Receive Data Register HL	RDRHL	R	00h	H'E800B810	16
	Modulation Duty Register	MDDR	R/W	FFh	H'E800B812	8

Note: * Partly, R only

18.2.1 Receive Shift Register (RSR)

RSR is a shift register which is used to receive serial data input from the RXDn pin and converts it into parallel data. When one frame of data has been received, it is automatically transferred to the RDR register.
The RSR register cannot be directly accessed by the CPU.

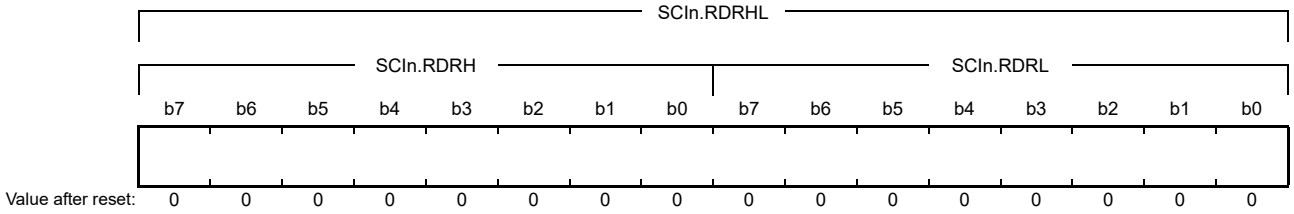
18.2.2 Receive Data Register (RDR)



RDR is an 8-bit register that stores receive data.
When one frame of serial data has been received, the received serial data is transferred from RSR to RDR. Then the RSR register can receive the next data.
Since RSR and RDR function as a double buffer in this way, continuous receive operations can be performed.
Read RDR only once after a receive data full interrupt (RXI) has occurred. Note that if next one frame of data is received before reading receive data from RDR, an overrun error occurs.
RDR cannot be written to by the CPU.

18.2.3 Receive Data Register H, L, HL (RDRH, RDRL, RDRHL)

- Receive Data Register H (RDRH)
- Receive Data Register L (RDRL)
- Receive Data Register HL (RDRHL)



RDRH and RDRL are 8-bit registers that store receive data. Use these registers when asynchronous mode and 9-bit data length are selected.

RDRL is the shadow register of RDR; i.e. access to RDRL is equivalent to access to RDR.

After one frame of data is received, the received data is transferred from the RSR register to these registers, thus allowing the RSR register to receive the next data.

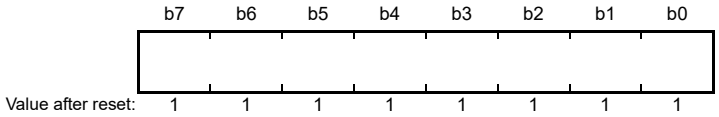
The RSR, RDRH and RDRL registers have a double-buffered construction to enable continuous reception.

Read RDRH and RDRL should be performed only once in the order from RDRH to RDRL when a receive data full interrupt (RXI) request is issued. Note that an overrun error occurs when the next frame of data is received before the received data has been read from RDRL.

The CPU cannot write to the RDRH and RDRL registers. Bits 0 to 7 in RDRH are fixed to 0. These bits are read as 0. The write value should be 0.

The RDRHL register can be accessed in 16-bit units.

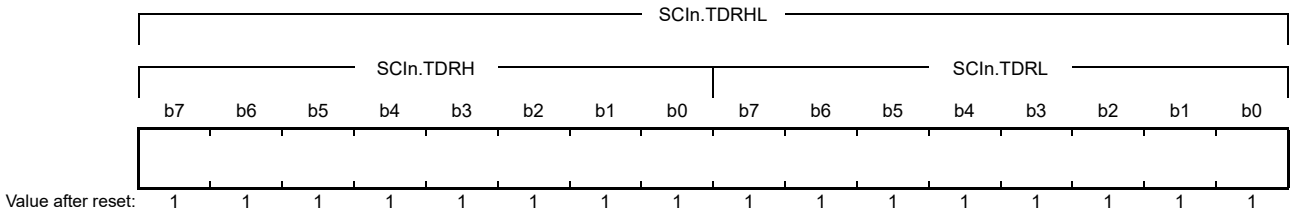
18.2.4 Transmit Data Register (TDR)



TDR is an 8-bit register that stores transmit data.
When the SCI detects that TSR is empty, it transfers the transmit data written in TDR to TSR and starts transmission.
The double-buffered structures of TDR and TSR enable continuous serial transmission. If the next transmit data has already been written to TDR when one frame of data is transmitted, the SCI transfers the written data to TSR to continue transmission.
The CPU is able to read from or write to TDR at any time. Only write transmit data to TDR once after each instance of the transmit data empty interrupt (TXI).

18.2.5 Transmit Data Register H, L, HL (TDRH, TDRL, TDRHL)

- Transmit Data Register H (TDRH)
- Transmit Data Register L (TDRL)
- Transmit Data Register HL (TDRHL)



TDRH and TDRL are 8-bit registers that store transmit data. Use these registers when asynchronous mode and 9-bit data length are selected.
TDRL is the shadow register of TDR; i.e. access to TDRL is equivalent to access to TDR.
When empty space is detected in the TSR register, the transmit data stored in the TDRH and TDRL registers is transferred to TSR; i.e., transmitting is started.
The TSR, TDRH and TDRL registers have a double-buffered construction to realize continuous reception. When the next data to be transmitted is stored in TDRL after one frame of data has been transmitted, the transmitting operation is continued by transfer to the TSR register.
The CPU can read and write to the TDRH and TDRL registers. Bits 0 to 7 in RDRH are fixed to 1. These bits are read as 1. The write value should be 1.
Writing transmit data to the TDRH and TDRL registers should be performed only once in the order from TDRH to TDRL when a transmit data empty interrupt (TXI) request is issued.
The TDRHL register can be accessed in 16-bit units.

18.2.6 Transmit Shift Register (TSR)

TSR is a shift register that transmits serial data.
To perform serial data transmission, the SCI first automatically transfers transmit data from TDR to TSR, and then sends the data to the TXDn pin.
TSR cannot be directly accessed by the CPU.

18.2.7 Serial Mode Register (SMR)

Note: Some bits in SMR have different functions in smart card interface mode and non-smart card interface mode.

(1) Non-Smart Card Interface Mode (SCMR.SMIF = 0)

	b7	b6	b5	b4	b3	b2	b1	b0
	CM	CHR	PE	PM	STOP	MP	CKS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	b1 b0 0 0: P1φ clock (n = 0)*1 0 1: P1φ/4 clock (n = 1)*1 1 0: P1φ/16 clock (n = 2)*1 1 1: P1φ/64 clock (n = 3)*1	R/W*4
b2	MP	Multi-Processor Mode	(Valid only in asynchronous mode) 0: Multi-processor communications function is disabled 1: Multi-processor communications function is enabled	R/W*4
b3	STOP	Stop Bit Length	(Valid only in asynchronous mode) 0: 1 stop bit 1: 2 stop bits	R/W*4
b4	PM	Parity Mode	(Valid only when the PE bit is 1) 0: Selects even parity 1: Selects odd parity	R/W*4
b5	PE	Parity Enable	(Valid only in asynchronous mode) • When transmitting 0: Parity bit addition is not performed 1: The parity bit is added • When receiving 0: Parity bit checking is not performed 1: The parity bit is checked	R/W*4
b6	CHR	Character Length	(Valid only in asynchronous mode*2) Selects in combination with the SCMR.CHR1 bit. CHR1 CHR 0 0: Transmit/receive in 9-bit data length 0 1: Transmit/receive in 9-bit data length 1 0: Transmit/receive in 8-bit data length (initial value) 1 1: Transmit/receive in 7-bit data length*3	R/W*4
b7	CM	Communications Mode	0: Asynchronous mode 1: Clock synchronous mode	R/W*4

Note 1. n is the decimal notation of the value of n in BRR (refer to section 18.2.11, Bit Rate Register (BRR)).
Note 2. In other than asynchronous mode, this bit setting is invalid and a fixed data length of 8 bits is used.
Note 3. LSB first is fixed and the MSB (bit 7) in TDR is not transmitted in transmission.
Note 4. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

CKS[1:0] Bits (Clock Select)

These bits select the clock source for the on-chip baud rate generator.
For the relation between the settings of these bits and the baud rate, refer to section 18.2.11, Bit Rate Register (BRR).

MP Bit (Multi-Processor Mode)

Disables/enables the multi-processor communications function. The settings of the PE bit and PM bit are invalid in multi-processor mode.

STOP Bit (Stop Bit Length)

Selects the stop bit length in transmission.
In reception, only the first stop bit is checked regardless of this bit setting. If the second stop bit is 0, it is treated as the start bit of the next transmit frame.

PM Bit (Parity Mode)

Selects the parity mode (even or odd) for transmission and reception.
The setting of the PM bit is invalid in multi-processor mode.

PE Bit (Parity Enable)

When this bit is set to 1, the parity bit is added to transmit data, and the parity bit is checked in reception.
Irrespective of the setting of the PE bit, the parity bit is not added or checked in multi-processor format.

CHR Bit (Character Length)

Selects the data length for transmission and reception.
Selects in combination with the CHR1 bit in SCMR.
In other than asynchronous mode, a fixed data length of 8 bits is used.

(2) Smart Card Interface Mode (SCMR.SMIF = 1)

	b7	b6	b5	b4	b3	b2	b1	b0
	GM	BLK	PE	PM	BCP[1:0]		CKS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	b1 b0 0 0: P1φ clock (n = 0)*1 0 1: P1φ/4 clock (n = 1)*1 1 0: P1φ/16 clock (n = 2)*1 1 1: P1φ/64 clock (n = 3)*1	R/W*2
b3, b2	BCP[1:0]	Base Clock Pulse	Selects the number of base clock cycles in combination with the SCMR.BCP2 bit. Table 18.4 lists the combinations of the SCMR.BCP2 bit and SMR.BCP[1:0] bits.	R/W*2
b4	PM	Parity Mode	(Valid only when the PE bit is 1) 0: Selects even parity 1: Selects odd parity	R/W*2
b5	PE	Parity Enable	When this bit is set to 1, a parity bit is added to transmit data, and the parity of received data is checked. Set this bit to 1 in smart card interface mode.	R/W*2
b6	BLK	Block Transfer Mode	0: Normal mode operation 1: Block transfer mode operation	R/W*2
b7	GM	GSM Mode	0: Normal mode operation 1: GSM mode operation	R/W*2

Note 1. n is the decimal notation of the value of n in BRR (refer to section 18.2.11, Bit Rate Register (BRR)).
Note 2. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

CKS[1:0] Bits (Clock Select)

These bits select the clock source for the on-chip baud rate generator.
For the relationship between the settings of these bits and the baud rate, refer to section 18.2.11, Bit Rate Register (BRR).

BCP[1:0] Bits (Base Clock Pulse)

These bits select the number of base clock cycles in a 1-bit data transfer time in smart card interface mode.
Set these bits in combination with the SCMR.BCP2 bit.
For details, refer to section 18.6.4, Receive Data Sampling Timing and Reception Margin.

Table 18.4 Combinations of the SCMR.BCP2 Bit and SMR.BCP[1:0] Bits

SCMR.BCP2 Bit	SMR.BCP[1:0] Bits		Number of Base Clock Cycles for 1-Bit Transfer Period
0	0	0	93 clock cycles (S = 93)*1
0	0	1	128 clock cycles (S = 128)*1
0	1	0	186 clock cycles (S = 186)*1
0	1	1	512 clock cycles (S = 512)*1
1	0	0	32 clock cycles (S = 32)*1 (Initial Value)
1	0	1	64 clock cycles (S = 64)*1
1	1	0	372 clock cycles (S = 372)*1
1	1	1	256 clock cycles (S = 256)*1

Note 1. S is the value of S in BRR (refer to section 18.2.11, Bit Rate Register (BRR)).

PM Bit (Parity Mode)

Selects the parity mode for transmission and reception (even or odd).
For details on the usage of this bit in smart card interface mode, refer to section 18.6.2, Data Format (Except in Block Transfer Mode).

PE Bit (Parity Enable)

Set the PE bit to 1.
The parity bit is added to transmit data before transmission, and the parity bit is checked in reception.

BLK Bit (Block Transfer Mode)

Setting this bit to 1 allows block transfer mode operation.
For details, refer to section 18.6.3, Block Transfer Mode.

GM Bit (GSM Mode)

Setting this bit to 1 allows GSM mode operation.
In GSM mode, the SSR.TEND flag set timing is put forward to 11.0 etu (elementary time unit = 1-bit transfer time) from the start and the clock output control function is appended. For details, refer to section 18.6.6, Serial Data Transmission (Except in Block Transfer Mode) and section 18.6.8, Clock Output Control.

18.2.8 Serial Control Register (SCR)

Note: Some bits in SCR have different functions in smart card interface mode and non-smart card interface mode.

(1) Non-Smart Card Interface Mode (SCMR.SMIF = 0)

	b7	b6	b5	b4	b3	b2	b1	b0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKE[1:0]	Clock Enable	• For SCI0 and SCI1 (Asynchronous mode) b1 b0 0 0: On-chip baud rate generator The SCKn pin functions as I/O port. 0 1: On-chip baud rate generator The clock with the same frequency as the bit rate is output from the SCKn pin. 1 x: External clock The clock with a frequency 16 times the bit rate should be input from the SCKn pin. Input a clock signal with a frequency eight times the bit rate when the SEMR.ABCS bit is 1. (Clock synchronous mode) b1 b0 0 x: Internal clock The SCKn pin functions as the clock output pin. 1 x: External clock The SCKn pin functions as the clock input pin.	R/W*1
b2	TEIE	Transmit End Interrupt Enable	0: A TEI interrupt request is disabled 1: A TEI interrupt request is enabled	R/W
b3	MPIE	Multi-Processor Interrupt Enable	(Valid in asynchronous mode when SMR.MP = 1) 0: Normal reception 1: When the data with the multi-processor bit set to 0 is received, the data is not read, and setting the status flags ORER and FER in SSR to 1 is disabled. When the data with the multi-processor bit set to 1 is received, the MPIE bit is automatically cleared to 0, and normal reception is resumed.	R/W
b4	RE	Receive Enable	0: Serial reception is disabled 1: Serial reception is enabled	R/W*2
b5	TE	Transmit Enable	0: Serial transmission is disabled 1: Serial transmission is enabled	R/W*2
b6	RIE	Receive Interrupt Enable	0: RXI and ERI interrupt requests are disabled 1: RXI and ERI interrupt requests are enabled	R/W
b7	TIE	Transmit Interrupt Enable	0: A TXI interrupt request is disabled 1: A TXI interrupt request is enabled	R/W

x: Don't care

Note 1. Writable only when TE = 0 and RE = 0.

Note 2. 1 can be written only when TE = 0 and RE = 0, while the SMR.CM bit is 1. After setting TE or RE to 1, only 0 can be written to TE and RE. While the SMR.CM bit is 0 and the SIMR1.IICM bit is 0, writing is enabled under any condition.

CKE[1:0] Bits (Clock Enable)

These bits select the clock source and SCKn pin function.
The combination of the settings of these bits and of the SEMR.ACS0 bit sets the internal TMR clock.

TEIE Bit (Transmit End Interrupt Enable)

Enables or disables a TEI interrupt request.
A TEI interrupt request is disabled by setting the TEIE bit to 0.

MPIE Bit (Multi-Processor Interrupt Enable)

When this bit is set to 1 and the data with the multi-processor bit set to 0 is received, the data is not read and setting the status flags ORER and FER in SSR to 1 is disabled. When the data with the multi-processor bit set to 1 is received, the MPIE is automatically cleared to 0, and normal reception is resumed. For details, refer to section 18.4, Multi-Processor Communications Function.
When the receive data includes the MPB bit is SSR set to 0, the receive data is not transferred from the RSR to the RDR, a receive error is not detected, and setting the flags ORER and FER to 1 is disabled.
When the receive data includes the MPB bit set to 1, the MPB bit is set to 1, the MPIE bit is automatically cleared to 0, the RXI and ERI interrupt requests are enabled (if the RIE bit in SCR is set to 1), and setting the flags ORER and FER to 1 is enabled.
MPIE should be set to 0 if multi-processor communications function is not to be used.

RE Bit (Receive Enable)

Enables or disables serial reception.
When this bit is set to 1, serial reception is started by detecting the start bit in asynchronous mode or the synchronous clock input in clock synchronous mode. Note that SMR should be set prior to setting the RE bit to 1 in order to designate the reception format.
Even if reception is halted by setting the RE bit to 0, the ORER, FER, and PER flags in SSR are not affected and the previous value is retained.

TE Bit (Transmit Enable)

Enables or disables serial transmission.
When this bit is set to 1, serial transmission is started by writing transmit data to TDR. Note that SMR should be set prior to setting the TE bit to 1 in order to designate the transmission format.

RIE Bit (Receive Interrupt Enable)

Enables or disables RXI and ERI interrupt requests.
An RXI interrupt request is disabled by setting the RIE bit to 0.
An ERI interrupt request can be canceled by reading 1 from the ORER, FER, or PER flag in SSR and then setting the flag to 0, or setting the RIE bit to 0.

(2) Smart Card Interface Mode (SCMR.SMIF = 1)

	b7	b6	b5	b4	b3	b2	b1	b0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKE[1:0]	Clock Enable	• When SMR.GM = 0 b1 b0 0 0: Output disabled (The SCKn pin is available for use as an I/O port according to the I/O port settings.) 0 1: Clock output 1 x: (Setting prohibited) • When SMR.GM = 1 b1 b0 0 0: Output fixed low x 1: Clock output 1 0: Output fixed high	R/W*1
b2	TEIE	Transmit End Interrupt Enable	This bit should be 0 in smart card interface mode.	R/W
b3	MPIE	Multi-Processor Interrupt Enable	This bit should be 0 in smart card interface mode.	R/W
b4	RE	Receive Enable	0: Serial reception is disabled 1: Serial reception is enabled	R/W*2
b5	TE	Transmit Enable	0: Serial transmission is disabled 1: Serial transmission is enabled	R/W*2
b6	RIE	Receive Interrupt Enable	0: RXI and ERI interrupt requests are disabled 1: RXI and ERI interrupt requests are enabled	R/W
b7	TIE	Transmit Interrupt Enable	0: A TXI interrupt request is disabled 1: A TXI interrupt request is enabled	R/W

x: Don't care

Note 1. Writable only when TE = 0 and RE = 0.

Note 2. 1 can be written only when TE = 0 and RE = 0. After setting TE or RE to 1, only 0 can be written in TE and RE.

For details on interrupt requests, refer to section 18.8, Interrupt Sources.

CKE[1:0] Bits (Clock Enable)

These bits control the clock output from the SCKn pin.

In GSM mode, clock output can be dynamically switched. For details, refer to section 18.6.8, Clock Output Control.

TEIE Bit (Transmit End Interrupt Enable)

This bit should be 0 in smart card interface mode.

MPIE Bit (Multi-Processor Interrupt Enable)

This bit should be 0 in smart card interface mode.

RE Bit (Receive Enable)

Enables or disables serial reception.

When this bit is set to 1, serial reception is started by detecting the start bit. Note that SMR should be set prior to setting the RE bit to 1 in order to designate the reception format.

Even if reception is halted by setting the RE bit to 0, the ORER, FER, and PER flags in SSR are not affected and the previous value is retained.

TE Bit (Transmit Enable)

Enables or disables serial transmission.

When this bit is set to 1, serial transmission is started by writing transmit data to TDR. Note that SMR should be set prior to setting the TE bit to 1 in order to designate the transmission format.

RIE Bit (Receive Interrupt Enable)

Enables or disables RXI and ERI interrupt requests.

An RXI interrupt request is disabled by setting the RIE bit to 0.

An ERI interrupt request can be canceled by reading 1 from the ORER, FER, or PER flag in SSR and then setting the flag to 0, or setting the RIE bit to 0.

18.2.9 Serial Status Register (SSR)

Some bits in SSR have different functions in smart card interface mode and non-smart card interface mode.

(1) Non-Smart Card Interface Mode (SCMR.SMIF = 0)

	b7	b6	b5	b4	b3	b2	b1	b0
	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
Value after reset:	1	0	0	0	0	1	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MPBT	Multi-Processor Bit Transfer	Sets the multi-processor bit for adding to the transmission frame 0: Data transmission cycles 1: ID transmission cycles	R/W
b1	MPB	Multi-Processor	Value of the multi-processor bit in the reception frame 0: Data transmission cycles 1: ID transmission cycles	R
b2	TEND	Transmit End Flag	0: A character is being transmitted. 1: Character transfer has been completed.	R
b3	PER	Parity Error Flag	0: No parity error occurred 1: A parity error has occurred	R/(W)*1
b4	FER	Framing Error Flag	0: No framing error occurred 1: A framing error has occurred	R/(W)*1
b5	ORER	Overrun Error Flag	0: No overrun error occurred 1: An overrun error has occurred	R/(W)*1
b6	RDRF	Receive Data Full Flag	0: When data is transferred from RDR 1: When data has been received normally, and transferred from RSR to RDR	R/(W)*2
b7	TDRE	Transmit Data Empty Flag	0: When data is transferred to TDR 1: When data is transferred from TDR to TSR	R/(W)*2

Note 1. Only 0 can be written to this bit, to clear the flag.
Note 2. Write 1 when writing is necessary.

MPB Bit (Multi-Processor)

Holds the value of the multi-processor bit in the reception frame. This bit does not change when the SCR.RE bit is 0.

TEND Flag (Transmit End Flag)

Indicates completion of transmission.

[Setting conditions]

- When the SCR.TE bit is set to 0 (serial transmission is disabled)
  When the SCR.TE bit is changed from 0 to 1, the TEND flag is not affected and retains the value 1.
- When the TDR register is not updated at the time of transmission of the tail-end bit of a character being transmitted

[Clearing condition]

- When transmit data are written to the TDR register while the SCR.TE bit is 1

PER Flag (Parity Error Flag)

Indicates that a parity error has occurred during reception in asynchronous mode and the reception ends abnormally.
[Setting condition]

- When a parity error is detected during reception
  Although receive data when the parity error occurs is transferred to RDR, no RXI interrupt request occurs. Note that when the PER flag is being set to 1, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 0 is written to PER after reading PER = 1
  Even when the SCR.RE bit is set to 0 (serial reception is disabled), the PER flag is not affected and retains its previous value.

FER Flag (Framing Error Flag)

Indicates that a framing error has occurred during reception in asynchronous mode and the reception ends abnormally.
[Setting condition]

- When the stop bit is 0
  In 2-stop-bit mode, only the first stop bit is checked whether it is 1 but the second stop bit is not checked. Note that although receive data when the framing error occurs is transferred to RDR, no RXI interrupt request occurs. In addition, when the FER flag is being set to 1, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 0 is written to FER after reading FER = 1
  Even when the SCR.RE bit is set to 0, the FER flag is not affected and retains its previous value.

ORER Flag (Overrun Error Flag)

Indicates that an overrun error has occurred during reception and the reception ends abnormally.
[Setting condition]

- When the next data is received before receive data is read from RDR
  In RDR, receive data prior to an overrun error occurrence is retained, but data received after the overrun error occurrence is lost. When the ORER flag is set to 1, subsequent serial reception cannot be performed. Note that, in clock synchronous mode, serial transmission also cannot continue.

[Clearing condition]

- When 0 is written to ORER after reading ORER = 1
  Even when the SCR.RE bit is set to 0, the ORER flag is not affected and retains its previous value.

RDRF Flag (Receive Data Full Flag)

Indicates whether RDR has received data.
[Setting condition]

- When data has been received normally, and transferred from RSR to RDR

[Clearing condition]

- When data is transferred from RDR

TDRE Flag (Transmit Data Empty Flag)

Indicates whether TDR has data to be transmitted.

[Setting condition]

- When data is transferred from TDR to TSR

[Clearing condition]

- When data is transferred to TDR

(2) Smart Card Interface Mode (SCMR.SMIF = 1)

	b7	b6	b5	b4	b3	b2	b1	b0
	TDRE	RDRF	ORER	ERS	PER	TEND	MPB	MPBT
Value after reset:	1	0	0	0	0	1	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MPBT	Multi-Processor Bit Transfer	This bit should be set to 0 in smart card interface mode.	R/W
b1	MPB	Multi-Processor	This bit is not used in smart card interface mode. It should be set to 0.	R
b2	TEND	Transmit End Flag	0: A character is being transmitted. 1: Character transfer has been completed.	R
b3	PER	Parity Error Flag	0: No parity error occurred 1: A parity error has occurred	R/(W)*1
b4	ERS	Error Signal Status Flag	0: Low error signal not responded 1: Low error signal responded	R/(W)*1
b5	ORER	Overrun Error Flag	0: No overrun error occurred 1: An overrun error has occurred	R/(W)*1
b6	RDRF	Receive Data Full Flag	0: When data is transferred from RDR 1: When data has been received normally, and transferred from RSR to RDR	R/(W)*2
b7	TDRE	Transmit Data Empty Flag	0: When data is transferred to TDR 1: When data is transferred from TDR to TSR	R/(W)*2

Note 1. Only 0 can be written to this bit, to clear the flag.
Note 2. Write 1 when writing is necessary.

TEND Flag (Transmit End Flag)

With no error signal from the receiving side, this bit is set to 1 when further data for transfer is ready to be transferred to the TDR register.

[Setting conditions]

- When the SCR.TE bit = 0 (serial transmission is disabled)
  When the SCR.TE bit is changed from 0 to 1, the TEND flag is not affected and retains the value 1.
- When a specified period has elapsed after the latest transmission of 1 byte, the ERS flag is 0, and the TDR register is not updated
  The set timing is determined by register settings as listed below.
  When SMR.GM = 0 and SMR.BLK = 0, 12.5 etu after the start of transmission
  When SMR.GM = 0 and SMR.BLK = 1, 11.5 etu after the start of transmission
  When SMR.GM = 1 and SMR.BLK = 0, 11.0 etu after the start of transmission
  When SMR.GM = 1 and SMR.BLK = 1, 11.0 etu after the start of transmission

[Clearing condition]

- When transmit data are written to the TDR register while the SCR.TE bit is 1

PER Flag (Parity Error Flag)

Indicates that a parity error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When a parity error is detected during reception
  Although receive data when the parity error occurs is transferred to RDR, no RXI interrupt request occurs. Note that when the PER flag is being set to 1, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 0 is written to PER after reading PER = 1

Even when the SCR.RE bit is set to 0 (serial reception is disabled), the PER flag is not affected and retains its previous value.

ERS Flag (Error Signal Status Flag)

[Setting condition]

- When a low error signal is sampled

[Clearing condition]

- When 0 is written to ERS after reading ERS = 1
  Even when the SCR.RE bit is set to 0, the ERS flag is not affected and retains its previous value.

ORER Flag (Overrun Error Flag)

Indicates that an overrun error has occurred during reception and the reception ends abnormally.

[Setting condition]

- When the next data is received before receive data is read from RDR
  In RDR, the receive data prior to an overrun error occurrence is retained, but data received following the overrun error occurrence is lost. When the ORER flag is set to 1, subsequent serial reception cannot be performed.

[Clearing condition]

- When 0 is written to ORER after reading ORER = 1
  Even when the SCR.RE bit is set to 0, the ORER flag is not affected and retains its previous value.

RDRF Flag (Receive Data Full Flag)

Indicates whether RDR has received data.

[Setting condition]

- When data has been received normally, and transferred from RSR to RDR

[Clearing condition]

- When data is transferred from RDR

TDRE Flag (Transmit Data Empty Flag)

Indicates whether TDR has data to be transmitted.

[Setting condition]

- When data is transferred from TDR to TSR

[Clearing condition]

- When data is transferred to TDR

18.2.10 Smart Card Mode Register (SCMR)

	b7	b6	b5	b4	b3	b2	b1	b0
	BCP2	—	—	CHR1	SDIR	SINV	—	SMIF
Value after reset:	1	1	1	1	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	SMIF	Smart Card Interface Mode Select	0: Non-smart card interface mode (Asynchronous mode, clock synchronous mode,) 1: Smart card interface mode	R/W*1
b1	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b2	SINV	Transmitted/Received Data Invert	0: TDR contents are transmitted as they are. Receive data is stored as it is in RDR. 1: TDR contents are inverted before being transmitted. Receive data is stored in inverted form in RDR.	R/W*1
b3	SDIR	Transmitted/Received Data Transfer Direction	This bit can be used in the following modes. • Smart card interface mode • Asynchronous mode (multi-processor mode) • Clock synchronous mode 0: Transfer with LSB first 1: Transfer with MSB first	R/W*1
b4	CHR1	Character Length 1	(Only valid in asynchronous mode)*2 Selects in combination with the SMR.CHR bit. CHR1 CHR 0 0: Transmit/receive in 9-bit data length 0 1: Transmit/receive in 9-bit data length 1 0: Transmit/receive in 8-bit data length (initial value) 1 1: Transmit/receive in 7-bit data length*3	R/W*1
b6, b5	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b7	BCP2	Base Clock Pulse 2	Selects the number of base clock cycles in combination with the SMR.BCP[1:0] bits. Table 18.5 lists the combinations of the SCMR.BCP2 bit and SMR.BCP[1:0] bits.	R/W*1

Note 1. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).
Note 2. The setting is invalid and a fixed data length of 8 bits is used in modes other than asynchronous mode.
Note 3. LSB first should be selected and the value of MSB (b7) in TDR cannot be transmitted.

SMIF Bit (Smart Card Interface Mode Select)

When this bit is set to 1, smart card interface mode is selected.

When this bit is set to 0, non-smart card interface mode, i.e., asynchronous mode (including multi-processor mode), clock synchronous mode is selected.

SINV Bit (Transmitted/Received Data Invert)

Inverts the transmit/receive data logic level. This bit does not affect the logic level of the parity bit. To invert the parity bit, invert the PM bit in SMR.

CHR1 bit (Character Length 1)

Selects the data length of transmit/receive data.

Selects in combination with the CHR bit in SMR.

A fixed data length of 8 bits is used in modes other than asynchronous mode.

BCP2 Bit (Base Clock Pulse 2)

Selects the number of base clock cycles in a 1-bit data transfer time in smart card interface mode. Set this bit in combination with the SMR.BCP[1:0] bits.

Table 18.5 Combinations of the SCMR.BCP2 Bit and SMR.BCP[1:0] Bits

SCMR.BCP2 Bit	SMR.BCP[1:0] Bits		Number of Base Clock Cycles for 1-Bit Transfer Period
0	0	0	93 clock cycles (S = 93)*1
0	0	1	128 clock cycles (S = 128)*1
0	1	0	186 clock cycles (S = 186)*1
0	1	1	512 clock cycles (S = 512)*1
1	0	0	32 clock cycles (S = 32)*1 (Initial Value)
1	0	1	64 clock cycles (S = 64)*1
1	1	0	372 clock cycles (S = 372)*1
1	1	1	256 clock cycles (S = 256)*1

Note 1. S is the value of S in BRR (refer to section 18.2.11, Bit Rate Register (BRR)).

18.2.11 Bit Rate Register (BRR)

BRR is an 8-bit register that adjusts the bit rate.
As each SCI channel has independent baud rate generator control, different bit rates can be set for each. Table 18.6 shows the relationship between the setting (N) in the BRR and the bit rate (B) for normal asynchronous mode, multi-processor communication, clock synchronous mode, and smart card interface mode.
The initial value of BRR is FFh.
BRR can be read from by the CPU, but it can be written to only when the TE and RE bits in SCR are 0.

Table 18.6 Relationship between N Setting in BRR and Bit Rate B

Mode	SEMR Settings		BRR Setting	Error
	BGDM Bit	ABCS Bit		
Asynchronous or multi-processor communication	0	0	$N = \frac{P1\varphi \times 10^6}{64 \times 2^{2n-1} \times B} - 1$	$Error\ (\%) = \{\frac{P1\varphi \times 10^6}{B \times 64 \times 2^{2n-1} \times (N+1)} - 1\} \times 100$
	0	1	$N = \frac{P1\varphi \times 10^6}{32 \times 2^{2n-1} \times B} - 1$	$Error\ (\%) = \{\frac{P1\varphi \times 10^6}{B \times 32 \times 2^{2n-1} \times (N+1)} - 1\} \times 100$
	1	0		
	1	1	$N = \frac{P1\varphi \times 10^6}{16 \times 2^{2n-1} \times B} - 1$	$Error\ (\%) = \{\frac{P1\varphi \times 10^6}{B \times 16 \times 2^{2n-1} \times (N+1)} - 1\} \times 100$
Clock synchronous			$N = \frac{P1\varphi \times 10^6}{8 \times 2^{2n-1} \times B} - 1$	
Smart card interface			$N = \frac{P1\varphi \times 10^6}{S \times 2^{2n+1} \times B} - 1$	$Error\ (\%) = \{\frac{P1\varphi \times 10^6}{B \times S \times 2^{2n+1} \times (N+1)} - 1\} \times 100$

B: Bit rate (bps)
N: BRR setting for on-chip baud rate generator ($0 \le N \le 255$)
P1φ: Operating frequency (MHz)
n and S: Determined by the settings of the SMR and SCMR registers as listed in the table below.

Table 18.7 Clock Source Settings

SMR.CKS[1:0] Bit Setting	Clock Source	n
0 0	P1φ clock	0
0 1	P1φ/4 clock	1
1 0	P1φ/16 clock	2
1 1	P1φ/64 clock	3

Table 18.8 Base Clock Settings in Smart Card Interface Mode

SCMR.BCP2 Bit Setting	SMR.BCP[1:0] Bit Setting	Base Clock Cycles for 1-bit Period	S
0	0 0	93 clock cycles	93
0	0 1	128 clock cycles	128
0	1 0	186 clock cycles	186
0	1 1	512 clock cycles	512
1	0 0	32 clock cycles	32
1	0 1	64 clock cycles	64
1	1 0	372 clock cycles	372
1	1 1	256 clock cycles	256

Table 18.9 lists examples of N settings in BRR in normal asynchronous mode. Table 18.10 lists the maximum bit rate settable for each operating frequency. Examples of BRR (N) settings in clock synchronous mode are listed in Table 18.12. Examples of BRR (N) settings in smart card interface mode are listed in Table 18.14. In smart card interface mode, the number of base clock cycles S in a 1-bit data transfer time can be selected. For details, refer to section 18.6.4, Receive Data Sampling Timing and Reception Margin. Table 18.11 and Table 18.13 list the maximum bit rates with external clock input.

When either the asynchronous mode base clock select bit (ABCS) or the baud rate generator double-speed mode select bit (BGDM) in the serial extended mode register (SEMR) is set to 1 in asynchronous mode, the bit rate becomes twice that listed in Table 18.9. When both of those registers are set to 1, the bit rate becomes four times the listed value.

Table 18.9 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode)

Bit Rate (bps)	Operating Frequency P1φ (MHz)														
	8			9.8304			10			12			12.288		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	141	0.03	2	174	–0.26	2	177	–0.25	2	212	0.03	2	217	0.08
150	2	103	0.16	2	127	0.00	2	129	0.16	2	155	0.16	2	159	0.00
300	1	207	0.16	1	255	0.00	2	64	0.16	2	77	0.16	2	79	0.00
600	1	103	0.16	1	127	0.00	1	129	0.16	1	155	0.16	1	159	0.00
1200	0	207	0.16	0	255	0.00	1	64	0.16	1	77	0.16	1	79	0.00
2400	0	103	0.16	0	127	0.00	0	129	0.16	0	155	0.16	0	159	0.00
4800	0	51	0.16	0	63	0.00	0	64	0.16	0	77	0.16	0	79	0.00
9600	0	25	0.16	0	31	0.00	0	32	–1.36	0	38	0.16	0	39	0.00
19200	0	12	0.16	0	15	0.00	0	15	1.73	0	19	–2.34	0	19	0.00
31250	0	7	0.00	0	9	–1.70	0	9	0.00	0	11	0.00	0	11	2.40
38400	—	—	—	0	7	0.00	0	7	1.73	0	9	–2.34	0	9	0.00

Bit Rate (bps)	Operating Frequency P1φ (MHz)														
	14			16			17.2032			18			19.6608		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	248	–0.17	3	70	0.03	3	75	0.48	3	79	–0.12	3	86	0.31
150	2	181	0.16	2	207	0.16	2	223	0.00	2	233	0.16	2	255	0.00
300	2	90	0.16	2	103	0.16	2	111	0.00	2	116	0.16	2	127	0.00
600	1	181	0.16	1	207	0.16	1	223	0.00	1	233	0.16	1	255	0.00
1200	1	90	0.16	1	103	0.16	1	111	0.00	1	116	0.16	1	127	0.00
2400	0	181	0.16	0	207	0.16	0	223	0.00	0	233	0.16	0	255	0.00
4800	0	90	0.16	0	103	0.16	0	111	0.00	0	116	0.16	0	127	0.00
9600	0	45	–0.93	0	51	0.16	0	55	0.00	0	58	–0.69	0	63	0.00
19200	0	22	–0.93	0	25	0.16	0	27	0.00	0	28	1.02	0	31	0.00
31250	0	13	0.00	0	15	0.00	0	16	1.20	0	17	0.00	0	19	–1.70
38400	—	—	—	0	12	0.16	0	13	0.00	0	14	–2.34	0	15	0.00

Bit Rate (bps)	Operating Frequency P1φ (MHz)														
	20			25			30			33			40		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	3	88	–0.25	3	110	–0.02	3	132	0.13	3	145	0.33	3	177	–0.25
150	3	64	0.16	3	80	0.47	3	97	–0.35	3	106	0.39	3	129	0.16
300	2	129	0.16	2	162	–0.15	2	194	0.16	2	214	–0.07	3	64	0.16
600	2	64	0.16	2	80	0.47	2	97	–0.35	2	106	0.39	2	129	0.16
1200	1	129	0.16	1	162	–0.15	1	194	0.16	1	214	–0.07	2	64	0.16
2400	1	64	0.16	1	80	0.47	1	97	–0.35	1	106	0.39	1	129	0.16
4800	0	129	0.16	0	162	–0.15	0	194	0.16	0	214	–0.07	1	64	0.16
9600	0	64	0.16	0	80	0.47	0	97	–0.35	0	106	0.39	0	129	0.16
19200	0	32	–1.36	0	40	–0.76	0	48	–0.35	0	53	–0.54	0	64	0.16
31250	0	19	0.00	0	24	0.00	0	29	0.00	0	32	0.00	0	39	0.00
38400	0	15	1.73	0	19	1.73	0	23	1.73	0	26	–0.54	0	32	–1.36

Bit Rate (bps)	Operating Frequency P1φ (MHz)								
	50			60			66		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	3	221	–0.02						
150	3	162	–0.15	3	194	0.16	3	214	-0.07
300	3	80	0.47	3	97	–0.35	3	106	0.39
600	2	162	–0.15	3	48	–0.35	2	214	-0.07
1200	2	80	0.47	2	97	–0.35	2	106	0.39
2400	1	162	–0.15	2	48	–0.35	1	214	-0.07
4800	1	80	0.47	1	97	–0.35	1	106	0.39
9600	0	162	–0.15	1	48	–0.35			
19200	0	80	0.47	0	97	–0.35	0	106	0.39
31250	0	49	0.00	0	59	0.00	0	65	0.00
38400	0	40	–0.76	0	48	–0.35	0	53	-0.54

Note: This is an example when the ABCS and BGDM bits in SEMR are 0.
When either the ABCS bit or BGDM bit is set to 1, the bit rate doubles.
When both ABCS and BGDM bits in SEMR are set to 1, the bit rate increases four times.

Table 18.10 Maximum Bit Rate for Each Operating Frequency (Asynchronous Mode)

P1φ (MHz)	SEMR Settings				Maximum Bit Rate (bps)	P1φ (MHz)	SEMR Settings				Maximum Bit Rate (bps)
	BGDM Bit	ABCS Bit	n	N			BGDM Bit	ABCS Bit	n	N	
8	0	0	0	0	250000	19.6608	0	0	0	0	614400
		1	0	0	500000			1	0	0	1228800
	1	0	0	0			1	0	0	0	
		1	0	0	1000000			1	0	0	2457600
9.8304	0	0	0	0	307200	20	0	0	0	0	625000
		1	0	0	614400			1	0	0	1250000
	1	0	0	0			1	0	0	0	
		1	0	0	1228800			1	0	0	2500000
10	0	0	0	0	312500	25	0	0	0	0	781250
		1	0	0	625000			1	0	0	1562500
	1	0	0	0			1	0	0	0	
		1	0	0	1250000			1	0	0	3125000
12	0	0	0	0	375000	30	0	0	0	0	937500
		1	0	0	750000			1	0	0	1875000
	1	0	0	0			1	0	0	0	
		1	0	0	1500000			1	0	0	3750000
12.288	0	0	0	0	384000	33	0	0	0	0	1031250
		1	0	0	768000			1	0	0	2062500
	1	0	0	0			1	0	0	0	
		1	0	0	1536000			1	0	0	4125000
14	0	0	0	0	437500	40	0	0	0	0	1250000
		1	0	0	875000			1	0	0	2500000
	1	0	0	0			1	0	0	0	
		1	0	0	1750000			1	0	0	5000000

	SEMR Settings						SEMR Settings				
P1φ (MHz)	BGDM Bit	ABCS Bit	n	N	Maximum Bit Rate (bps)	P1φ (MHz)	BGDM Bit	ABCS Bit	n	N	Maximum Bit Rate (bps)
16	0	0	0	0	500000	50	0	0	0	0	1562500
		1	0	0	1000000			1	0	0	3125000
	1	0	0	0			1	0	0	0	
		1	0	0	2000000			1	0	0	6250000
17.2032	0	0	0	0	537600	60	0	0	0	0	1875000
		1	0	0	1075200			1	0	0	3750000
	1	0	0	0			1	0	0	0	
		1	0	0	2150400			1	0	0	7500000
18	0	0	0	0	562500	66	0	0	0	0	2062500
		1	0	0	1125000			1	0	0	4125000
	1	0	0	0			1	0	0	0	
		1	0	0	2250000			1	0	0	8250000

Table 18.11 Maximum Bit Rate with External Clock Input (Asynchronous Mode)

P1φ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bps) SEMR.ABCS Bit = 0	SEMR.ABCS Bit = 1
8	2.0000	125000	250000
9.8304	2.4576	153600	307200
10	2.5000	156250	312500
12	3.0000	187500	375000
12.288	3.0720	192000	384000
14	3.5000	218750	437500
16	4.0000	250000	500000
17.2032	4.3008	268800	537600
18	4.5000	281250	562500
19.6608	4.9152	307200	614400
20	5.0000	312500	625000
25	6.2500	390625	781250
30	7.5000	468750	937500
33	8.2500	515625	1031250
40	10.0000	625000	1250000
50	12.5000	781250	1562500
60	15.0000	937500	1875000
66	16.5000	1031250	2062500

Table 18.12 BRR Settings for Various Bit Rates (Clock Synchronous Mode)

Bit Rate	Operating Frequency P1φ (MHz)																					
	8		10		16		20		25		30		33		40		50		60		66	
(bps)	n	N	n	N	n	N	n	N	n	N	n	N	n	N	n	N	n	N	n	N	n	N
110																						
250	3	124	—	—	3	249																
500	2	249	—	—	3	124	—	—			3	233										
1 k	2	124	—	—	2	249	—	—	3	97	3	116	3	128	3	155	3	194	3	233		
2.5 k	1	199	1	249	2	99	2	124	2	155	2	187	2	205	2	249	3	77	3	93	3	102
5 k	1	99	1	124	1	199	1	249	2	77	2	93	2	102	2	124	2	155	3	46	2	205
10 k	0	199	0	249	1	99	1	124	1	155	1	187	1	205	1	249	2	77	2	93	2	102
25 k	0	79	0	99	0	159	0	199	0	249	1	74	1	82	1	99	1	124	1	149	1	164
50 k	0	39	0	49	0	79	0	99	0	124	0	149	0	164	1	49	1	61	1	74	1	82
100 k	0	19	0	24	0	39	0	49	0	62	0	74	0	82	0	99	0	124	0	149	0	164
250 k	0	7	0	9	0	15	0	19	0	24	0	29	0	32	0	39	0	49	0	59	0	65
500 k	0	3	0	4	0	7	0	9	—	—	0	14	—	—	0	19	0	24	0	29	0	32
1 M	0	1	—	—	0	3	0	4	—	—	—	—	—	—	0	9	—	—	0	14	0	16
2.5 M			0	0*1			0	1	—	—	0	2	—	—	0	3	0	4	0	5	0	6
5 M							0	0*1							0	1			0	2	0	2
7.5 M																			0	1	0	1

Space: Setting prohibited.

—: Can be set, but an error will occur.

Note 1. Continuous transmission or reception is impossible. After transmitting/receiving one frame of data, there is an interval of a 1-bit period before starting transmitting/receiving the next frame of data. The output of the synchronization clock is stopped for a 1-bit period. For this reason, it takes 9 bits worth of time to transfer one frame (8 bits) of data, and the average transfer rate is 8/9 times the bit rate.

Table 18.13 Maximum Bit Rate with External Clock Input (Clock Synchronous Mode)

P1φ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (Mbps)
8	1.3333	1.3333
10	1.6667	1.6667
12	2.0000	2.0000
14	2.3333	2.3333
16	2.6667	2.6667
18	3.0000	3.0000
20	3.3333	3.3333
25	4.1667	4.1667
30	5.0000	5.0000
33	5.5000	5.5000
40	6.6667	6.6667
50	8.3333	8.3333
60	10.0000	10.0000
66	11,0000	11,0000

Table 18.14 BRR Settings for Various Bit Rates (Smart Card Interface Mode, n = 0, S = 372)

Bit Rate (bps)	P1φ (MHz)	n	N	Error (%)
9600	7.1424	0	0	0.00
	10.00	0	1	–30
	10.7136	0	1	–25
	13.00	0	1	–8.99
	14.2848	0	1	0.00
	16.00	0	1	12.01
	18.00	0	2	–15.99
	20.00	0	2	–6.66
	25.00	0	3	–12.49
	30.00	0	3	5.01
	33.00	0	4	–7.59
	40.00	0	5	–6.66
	50.00	0	6	0.01
	60.00	0	7	5.01
	66.00	0	8	2.67

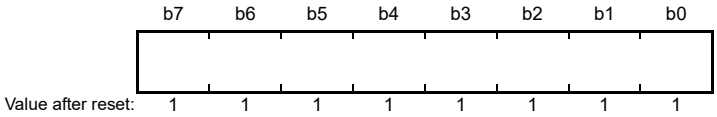
Table 18.15 Maximum Bit Rate for Each Operating Frequency (Smart Card Interface Mode, S = 32)

P1φ (MHz)	Maximum Bit Rate (bps)	n	N
10.00	156250	0	0
10.7136	167400	0	0
13.00	203125	0	0
16.00	250000	0	0
18.00	281250	0	0
20.00	312500	0	0
25.00	390625	0	0
30.00	468750	0	0
33.00	515625	0	0
40.00	625000	0	0
50.00	781250	0	0
60.00	937500	0	0
66.00	1031250	0	0

Table 18.16 Maximum Bit Rate for Each Operating Frequency (Smart Card Interface Mode, S = 372)

P1φ (MHz)	Maximum Bit Rate (bps)	n	N
10.00	13441	0	0
10.7136	14400	0	0
13.00	17473	0	0
16.00	21505	0	0
18.00	24194	0	0
20.00	26882	0	0
25.00	33602	0	0
30.00	40323	0	0
66.00	88710	0	0

18.2.12 Modulation Duty Register (MDDR)



MDDR corrects the bit rate adjusted by the BRR register.
When the BRME bit in SEMR is set to 1, the bit rate generated by the on-chip baud rate generator is evenly corrected according to the settings of MDDR (M/256). The relationship between the MDDR setting (M) and the bit rate (B) is given in Table 18.17.
The initial value of MDDR is FFh. Bit 7 in this register is fixed to 1.
The CPU can read the MDDR register, but this register is only writable when the TE and RE bits in SCR are 0.

Table 18.17 Relationship between MDDR Setting (M) and Bit Rate (B)
When Bit Rate Modulation Function is Used

Mode	SEMR Settings BGDM Bit	SEMR Settings ABCS Bit	BRR Setting	Error
Asynchronous or multi-processor communication	0	0	$N = \frac{P1\varphi \times 10^6}{64 \times 2^{2n-1} \times (256/M) \times B} - 1$	$Error\ (\%) = \{\frac{P1\varphi \times 10^6}{B \times 64 \times 2^{2n-1} \times (256/M) \times (N+1)} - 1\} \times 100$
	0	1	$N = \frac{P1\varphi \times 10^6}{32 \times 2^{2n-1} \times (256/M) \times B} - 1$	$Error\ (\%) = \{\frac{P1\varphi \times 10^6}{B \times 32 \times 2^{2n-1} \times (256/M) \times (N+1)} - 1\} \times 100$
	1	0		
	1	1	$N = \frac{P1\varphi \times 10^6}{16 \times 2^{2n-1} \times (256/M) \times B} - 1$	$Error\ (\%) = \{\frac{P1\varphi \times 10^6}{B \times 16 \times 2^{2n-1} \times (256/M) \times (N+1)} - 1\} \times 100$
Clock synchronous*1			$N = \frac{P1\varphi \times 10^6}{8 \times 2^{2n-1} \times (256/M) \times B} - 1$	
Smart card interface			$N = \frac{P1\varphi \times 10^6}{S \times 2^{2n+1} \times (256/M) \times B} - 1$	$Error\ (\%) = \{\frac{P1\varphi \times 10^6}{B \times S \times 2^{2n+1} \times (256/M) \times (N+1)} - 1\} \times 100$

B: Bit rate (bps)
M: MDDR setting (128 ≤ MDDR ≤ 256)
N: BRR setting for baud rate generator (0 ≤ N ≤ 255)
P1φ: Operating frequency (MHz)
n and S: Determined by the settings of the SMR and SCMR registers as listed in Table 18.7 and Table 18.8, section 18, Bit Rate Register (BRR).

Note 1. Do not use this function in clock synchronous mode with the highest speed setting (SMR.CKS[1:0] = 00b, SCR.CKE[1] = 0, and BRR = 0).

18.2.13 Serial Extended Mode Register (SEMR)

	b7	b6	b5	b4	b3	b2	b1	b0
	RXDESEL	BGDM	NFEN	ABCS	—	BRME	—	ACS0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ACS0	Asynchronous Mode Clock Source Select	(Valid only in asynchronous mode) 0: External clock input 1: Reserved bit	R/W*1
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	BRME	Bit Rate Modulation Enable	0: Bit rate modulation function is disabled. 1: Bit rate modulation function is enabled.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ABCS	Asynchronous Mode Base Clock Select	(Valid only in asynchronous mode) 0: Selects 16 base clock cycles for 1-bit period. 1: Selects 8 base clock cycles for 1-bit period.	R/W*1
b5	NFEN	Digital Noise Filter Function Enable	(In asynchronous mode) 0: Noise cancellation function for the RXDn input signal is disabled. 1: Noise cancellation function for the RXDn input signal is enabled.	R/W*1
b6	BGDM	Baud Rate Generator Double-Speed Mode Select	(Only valid the CKE[1] bit in SCR is 0 in asynchronous mode). 0: Baud rate generator outputs the clock with normal frequency. 1: Baud rate generator outputs the clock with doubled frequency.	R/W
b7	RXDESEL	Asynchronous Start Bit Edge Detection Select	(Valid only in asynchronous mode) 0: The low level on the RXDn pin is detected as the start bit. 1: A falling edge on the RXDn pin is detected as the start bit.	R/W*1

Note 1. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

SEMR selects the clock source for 1-bit period in asynchronous mode.

ACS0 Bit (Asynchronous Mode Clock Source Select)

Selects the clock source in the asynchronous mode.

The ACS0 bit is valid in asynchronous mode (CM bit in SMR = 0) and when an external clock input is selected (CKE[1:0] bits in SCR = 10b or 11b).

Set the ACS0 bit to 0 in other than asynchronous mode.

BRME bit (Bit Rate Modulation Enable)

Enables and disables the bit rate modulation function. The bit rate generated by on-chip baud rate generator is evenly corrected when this function is enabled.

NFEN Bit (Digital Noise Filter Function Enable)

This bit enables or disables the digital noise filter function.

When the function is enabled, noise cancellation is applied to the RXDn input signal in asynchronous mode.

In any mode other than above, set the NFEN bit to 0 to disable the digital noise filter function.

When the function is disabled, input signals are transferred as is, as internal signals.

BGDM bit (Baud Rate Generator Double-Speed Mode Select)

Selects the cycle of output clock for the baud rate generator.

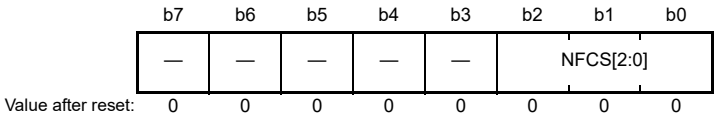
This bit is valid when the on-chip baud rate generator is selected as the clock source (SCR.CKE[1] = 0) in asynchronous mode (SMR.CM = 0). For the clock output from the baud rate generator, either normal or doubled frequency can be selected. The base clock is generated by the clock output from the baud rate generator. When the BGDM bit is set to 1,

the base clock cycle is halved and the bit rate is doubled.
Set this bit to 0 in modes other than asynchronous mode.

RXDESEL Bit (Asynchronous Start Bit Edge Detection Select)

Selects the detection method of the start bit for reception in asynchronous mode. When a break occurs, data receiving operation depends on the settings of this bit. Set this bit to 1 when reception should be stopped while a break occurs or when reception should be started without retaining the RXDn pin input at high level for the period of one data frame or longer after completion of the break.
Set this bit to 0 in modes other than asynchronous mode.

18.2.14 Noise Filter Setting Register (SNFR)



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	NFCS[2:0]	Noise Filter Clock Select	In asynchronous mode, the standard setting for the base clock is as follows. b2 b0 0 0 0: The clock signal divided by 1 is used with the noise filter. Settings other than above are prohibited.	R/W*1
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Writing to these bits is only possible when the RE and TE bits in the SCR are 0 (serial reception and transmission disabled).

NFCS[2:0] Bits (Noise Filter Clock Select)

These bits select the sampling clock for the digital noise filter. To use the noise filter in asynchronous mode, set these bits to 000b.

18.2.15 Extended function control register (SECR)

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	CTSE	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b1	CTSE	CTS Enable	0: CTS function is disabled (RTS output function is enabled). 1: CTS function is enabled.	R/W*1
b2 to b7	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Writing to the bit is only possible when the SCR.RE and SCR.TE bits are 0.

SECR is used to select the extension settings in asynchronous and clock synchronous modes.

CTSE Bit (CTS Enable)

Set this bit to 1 if the SSn# pin is to be used for inputting of the CTS control signal to control of transmission and reception. The RTS signal is output when this bit is set to 0. Set this bit to 0 in smart card interface mode. Do not set both the CTSE and SSE bits to enabled (even if this setting is made, operation is the same as that when these bits are set to 0).

18.3 Operation in Asynchronous Mode

Figure 18.2 shows the general format for asynchronous serial communications.
One frame consists of a start bit (low level), transmit/receive data, a parity bit, and stop bits (high level).
In asynchronous serial communications, the communications line is usually held in the mark state (high level).
The SCI monitors the communications line. When the SCI detects a low, it regards that as a start bit and starts serial communication.
Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communications. Both the transmitter and the receiver also have a double-buffered structure, so that data can be read or written during transmission or reception, enabling continuous data transmission and reception.

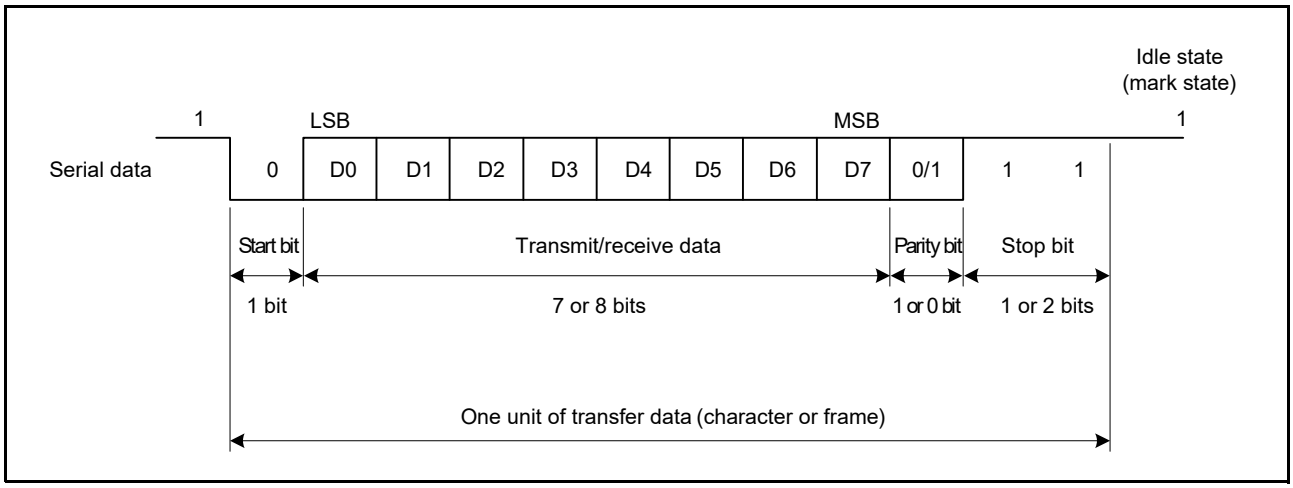


Figure 18.2 Data Format in Asynchronous Serial Communications
(Example with 8-Bit Data, Parity, 2 Stop Bits)

18.3.1 Serial Data Transfer Format

Table 18.18 lists the serial data transfer formats that can be used in asynchronous mode.
Any of 18 transfer formats can be selected according to the SMR and SCMR setting. For details of multi-processor function, refer to section 18.4, Multi-Processor Communications Function.

Table 18.18 Serial Transfer Formats (Asynchronous Mode)

SCMR Setting	SMR Setting				Serial Transfer Format and Frame Length
CHR1	CHR	PE	MP	STOP	1 2 3 4 5 6 7 8 9 10 11 12 13
0	0	0	0	0	S 9-bit data STOP
0	0	0	0	1	S 9-bit data STOP STOP
0	0	1	0	0	S 9-bit data P STOP
0	0	1	0	1	S 9-bit data P STOP STOP
1	0	0	0	0	S 8-bit data STOP
1	0	0	0	1	S 8-bit data STOP STOP
1	0	1	0	0	S 8-bit data P STOP
1	0	1	0	1	S 8-bit data P STOP STOP
1	1	0	0	0	S 7-bit data STOP
1	1	0	0	1	S 7-bit data STOP STOP
1	1	1	0	0	S 7-bit data P STOP
1	1	1	0	1	S 7-bit data P STOP STOP
0	0	—	1	0	S 9-bit data MPB STOP
0	0	—	1	1	S 9-bit data MPB STOP STOP
1	0	—	1	0	S 8-bit data MPB STOP
1	0	—	1	1	S 8-bit data MPB STOP STOP
1	1	—	1	0	S 7-bit data MPB STOP
1	1	—	1	1	S 7-bit data MPB STOP STOP

S: Start bit
STOP: Stop bit
P: Parity bit
MPB: Multi-processor bit

18.3.2 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCI operates on a base clock with a frequency of 16 times*1 the bit rate.
In reception, the SCI samples the falling edge of the start bit using the base clock, and performs internal synchronization. Since receive data is sampled at the rising edge of the 8th pulse*1 of the base clock, data is latched at the middle of each bit, as shown in Figure 18.3. Thus the reception margin in asynchronous mode is determined by formula (1) below.

$$M = \left| (0.5 - \frac{1}{2N}) - (L - 0.5)\ F - \frac{D - 0.5}{N}\ (1 + F) \right| \times 100\ [\%] \cdots \text{Formula (1)}$$

M: Reception margin
N: Ratio of bit rate to clock (N = 16 when ABCS in SEMR = 0, N = 8 when ABCS in SEMR = 1)
D: Duty cycle of clock (D = 0.5 to 1.0)
L: Frame length (L = 9 to 13)
F: Absolute value of clock frequency deviation

Assuming values of F = 0 and D = 0.5 in formula (1), the reception margin is determined by the formula below.

$M = \{0.5 - 1/(2 \times 16)\} \times 100\ (\%) = 46.875\%$

However, this is only the computed value, and a margin of 20% to 30% should be allowed in system design.

Note 1. This is an example when the ABCS bit in SEMR is 0. When the ABCS bit is 1, a frequency of 8 times the bit rate is used as a base clock and receive data is sampled at the rising edge of the 4th pulse of the base clock.

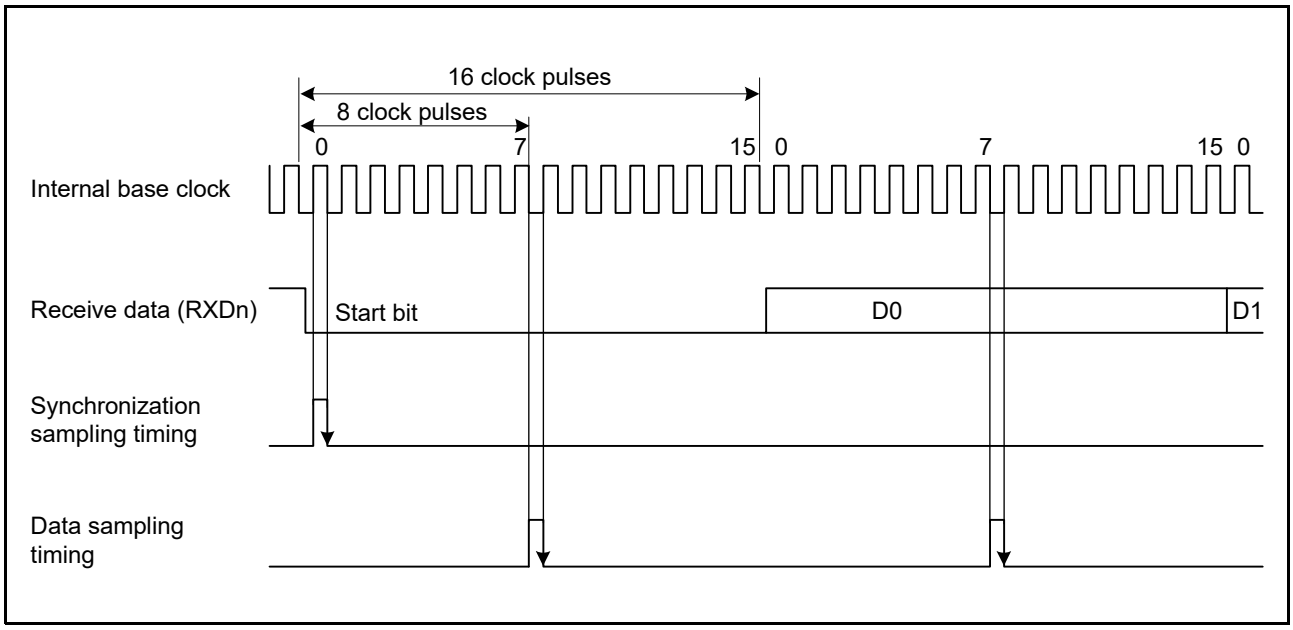


Figure 18.3 Receive Data Sampling Timing in Asynchronous Mode

18.3.3 Clock

Either an internal clock generated by the on-chip baud rate generator or an external clock input to the SCKn pin can be selected as the SCI's transfer clock, according to the setting of the CM bit in SMR and the CKE[1:0] bits in SCR.
When an external clock is input to the SCKn pin, the clock frequency should be 16 times the bit rate (when ABCS in SEMR = 0) and 8 times the bit rate (when ABCS in SEMR = 1).
When the SCI is operated on an internal clock, the clock can be output from the SCKn pin. The frequency of the clock output in this case is equal to the bit rate, and the phase is such that the rising edge of the clock is in the middle of the transmit data, as shown in Figure 18.4.

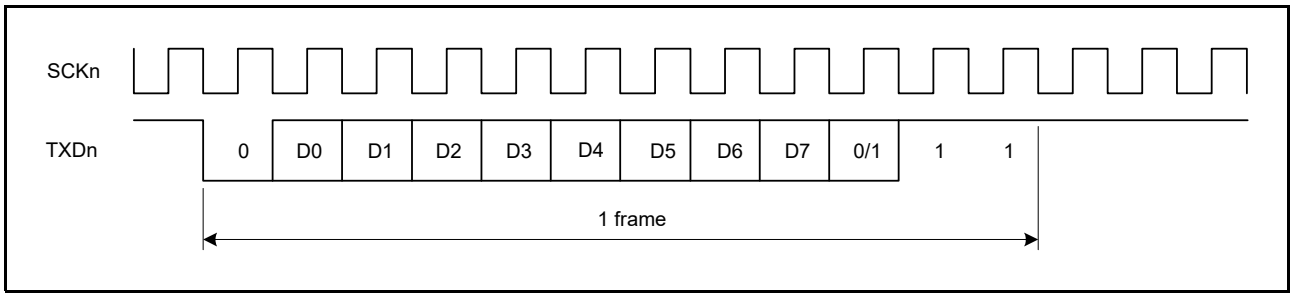


Figure 18.4 Phase Relationship between Output Clock and Transmit Data
(Asynchronous Mode: SMR.CHR = 0, PE = 1, MP = 0, STOP = 1)

18.3.4 Double-Speed Mode

The output clock frequency of the on-chip baud rate generator is doubled by setting the SEMR.BGDM bit to 1, enabling high-speed communication at a doubled bit rate. If the SEMR.ABCS bit is set to 1 under the above condition, the number of base clock cycles changes from 16 to 8, so the bit rate becomes four times faster than the initial state.
As shown by Formula (1) in section 18.3.2, Receive Data Sampling Timing and Reception Margin in Asynchronous Mode, setting the SEMR.ABCS bit to 1 changes the number of cycles to 8, and the sampling interval becomes longer. This causes the reception margin to decrease. Therefore, setting the SEMR.BGDM bit to 1 and the SEMR.ABCS bit to 0 is recommended instead of setting the SEMR.BGDM bit to 0 and the SEMR.ABCS bit to 1 for high-speed operation at a doubled bit rate.

18.3.5 CTS and RTS Functions

The CTS function is the use of input on the CTSn# pin in transmission control. Setting the SECR.CTSE bit to 1 enables the CTS function. When the CTS function is enabled, placing the low level on the CTSn# pin causes transmission to start.
Applying the high level to the CTS# pin while transmission is in progress does not affect transmission of the current frame, which continues.
In the RTS function, by using the function of output on the RTSn# pin, a low level is output when reception becomes possible. Conditions for output of the low and high level are shown below.

[Conditions for low-level output]
Satisfaction of all conditions listed below

- The value of the RE bit in the SCR is 1
- Reception is not in progress
- There are no received data yet to be read
- The ORER, FER, and PER flags in the SSR are all 0

[Condition for high-level output]

- The conditions for low-level output have not been satisfied.

18.3.6 SCI Initialization (Asynchronous Mode)

Before transmitting and receiving data, start by writing the initial value 00h to SCR and then continue through the procedure for SCI given in Figure 18.5. Whenever the operating mode or transfer format is changed, SCR must be initialized before the change is made.

When the external clock is used in asynchronous mode, ensure that the clock signal is supplied even during initialization.

Note that setting the SCR.RE bit to 0 initializes neither the ORER, FER, and PER flags in SSR nor RDR, RDRH, and RDRL.

Moreover, note that switching the value of the SCR.TE bit from 1 to 0 or 0 to 1 while the SCR.TIE bit is 1 leads to the generation of a TXI interrupt request.

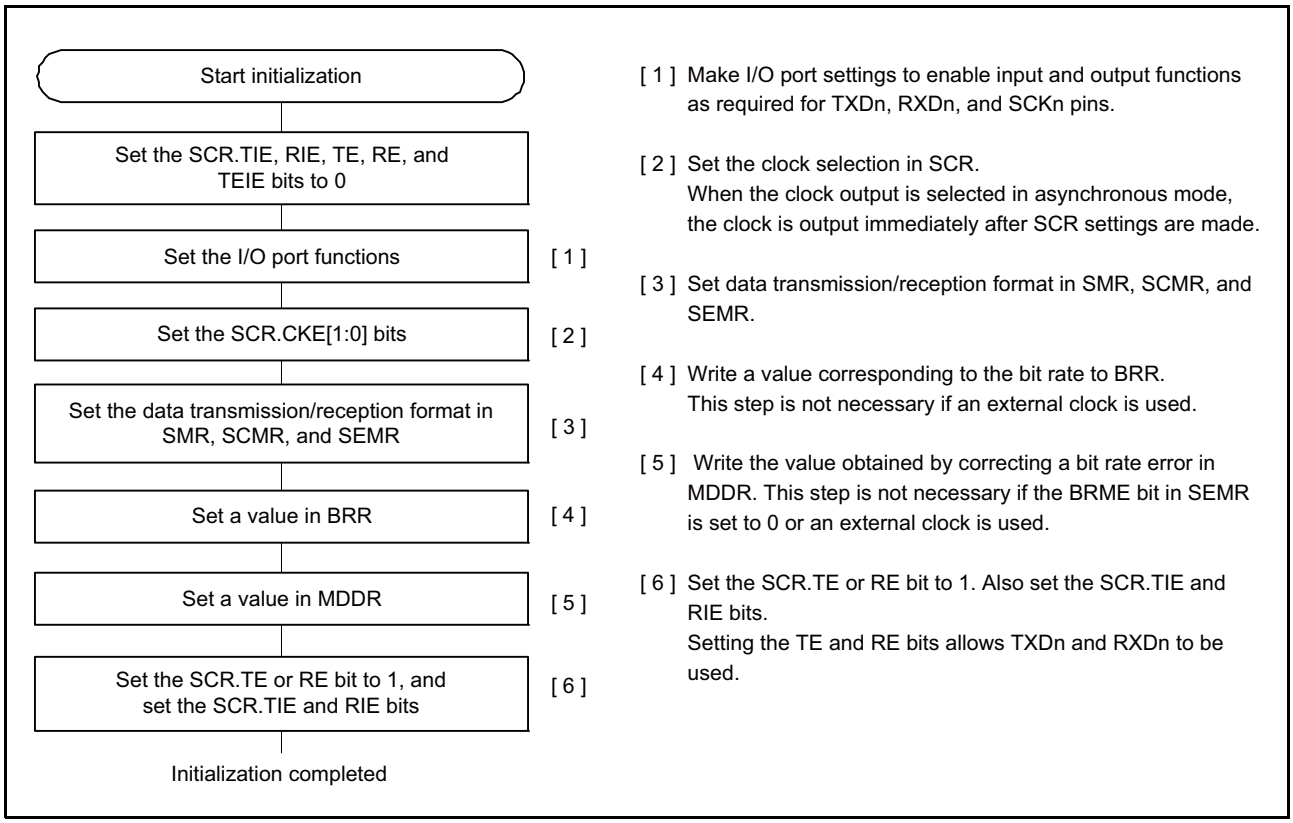


Figure 18.5 Sample SCI Initialization Flowchart (Asynchronous Mode)

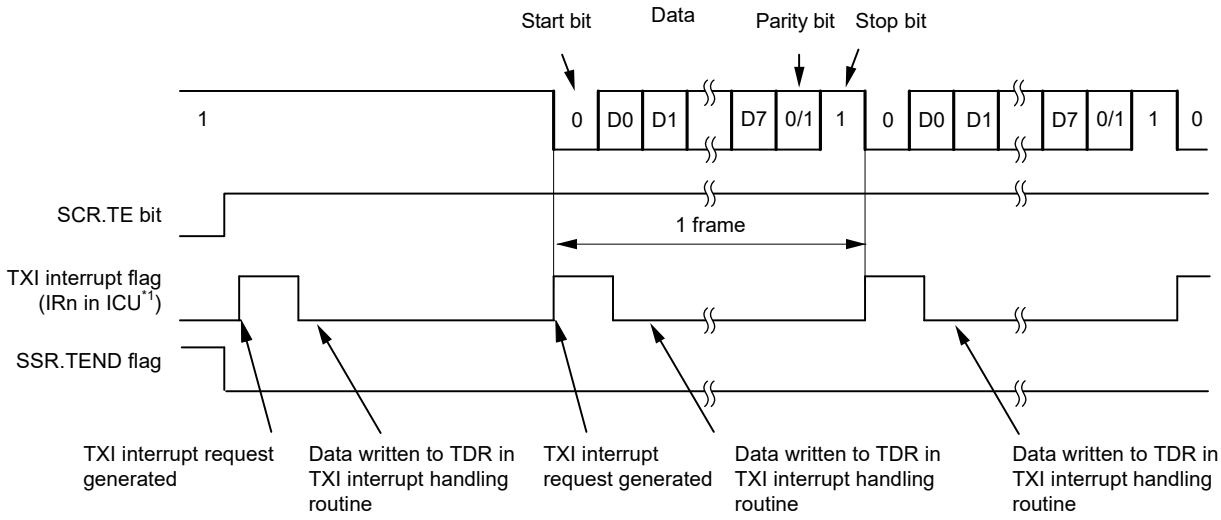
18.3.7 Serial Data Transmission (Asynchronous Mode)

Figure 18.6 to Figure 18.8 show an example of the operation for serial transmission in asynchronous mode.
In serial transmission, the SCI operates as described below.

1. The SCI transfers data from TDR*1 to TSR when data is written to TDR*1 in the TXI interrupt handling routine. The TXI interrupt request at the beginning of transmission is generated when the TE bit in SCR is set to 1 after the TIE bit in SCR is set to 1 or when these 2 bits are set to 1 simultaneously by a single instruction.
2. Transmission starts after the CTSE bit in SECR is set to 0 (CTS function is disabled) and a low level on the CTSn# pin causes data transfer from TDR*1 to TSR. If the TIE bit in SCR is 1 at this time, a TXI interrupt request is generated. Continuous transmission is obtainable by writing the next transmit data to TDR*1 in the TXI interrupt handling routine before transmission of the current transmit data is completed. When TEI interrupt requests are in use, set the SCR.TIE bit to 0 (a TXI interrupt request is disabled) and the SCR.TEIE bit to 1 (a TEI interrupt request is enabled) after the last of the data to be transmitted are written to the TDR*1*2 from the handling routine for TXI requests.
3. Data is sent from the TXDn pin in the following order: start bit, transmit data, parity bit or multi-processor bit (may be omitted depending on the format), and stop bit.
4. The SCI checks for updating of (writing to) TDR*3 at the time of stop bit output.
5. When TDR*3 is updated, setting of the CTSE bit in SECR to 0 (CTS function is disabled) or a low level input on the CTSn# pin cause the next transfer of the next transmit data from TDR*1 to TSR and sending of the stop bit, after which serial transmission of the next frame starts.
6. If TDR*3 is not updated, the TEND flag in SSR is set to 1, the stop bit is sent, and then the mark state is entered in which 1 is output. If the TEIE bit in SCR is 1 at this time, the TEND flag in SSR is set to 1 and a TEI interrupt request is generated.

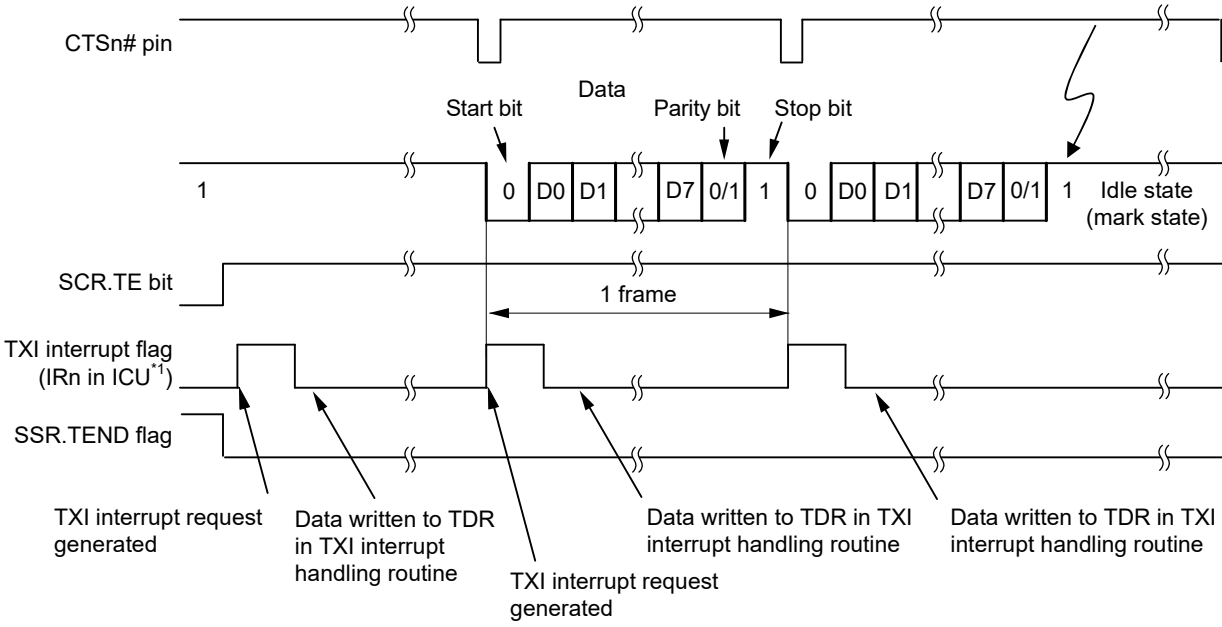
Note 1. Write data not to TDR but to the TDRH and TDRL registers when 9-bit data length is selected.
Note 2. Write data in the order from TDRH to TDRL when 9-bit data length is selected.
Note 3. The SCI checks for updating of the TDRL register only and does not check for updating of the TDRH register when 9-bit data length is selected.

Figure 18.9 shows a sample flowchart for serial transmission in asynchronous mode.



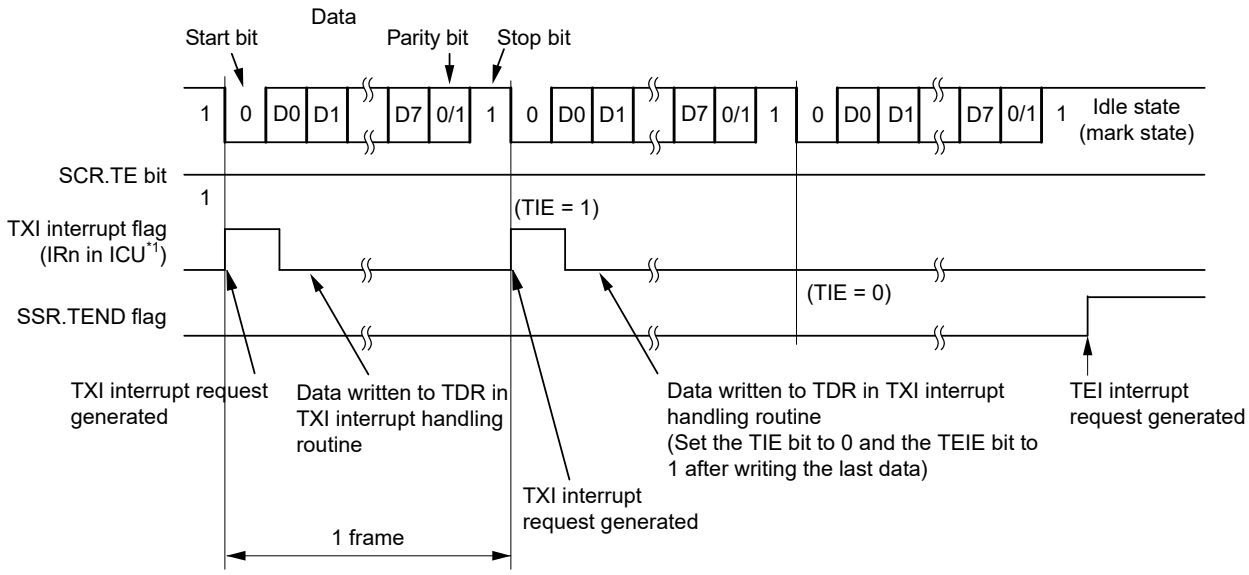
Note 1. Refer to section 7, Interrupt Controller for details on the corresponding interrupt vector number.

Figure 18.6 Example of Operation for Serial Transmission in Asynchronous Mode (1)
(with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Not Used, at the Beginning of Transmission)



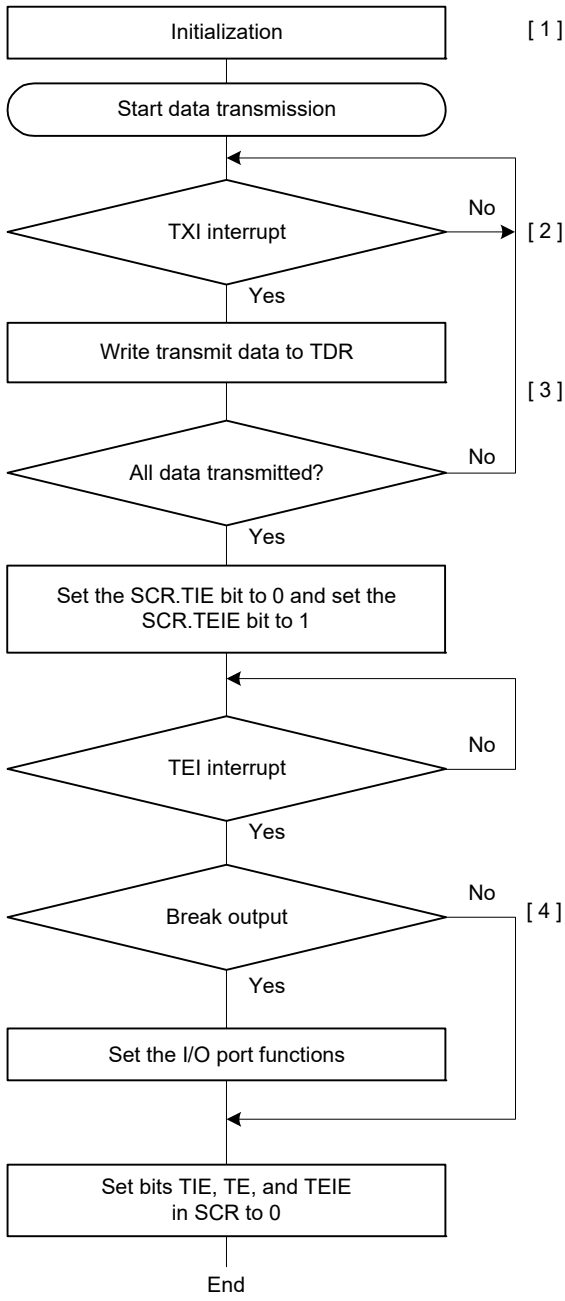
Note 1. Refer to section 7, Interrupt Controller for details on the corresponding interrupt vector number.

Figure 18.7 Example of Operation for Serial Transmission in Asynchronous Mode (2)
(with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Used, at the Beginning of Transmission)



Note 1. Refer to section 7, Interrupt Controller for details on the corresponding interrupt vector number.

Figure 18.8 Example of Operation for Serial Transmission in Asynchronous Mode (3)
(with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Not Used, from the Middle of Transmission until Transmission Completion)



[1] SCI initialization:
Set data transmission.
After the TE bit in SCR is set to 1, 1 is output for a frame, and transmission is enabled.

[2] Transmit data write to TDR by a TXI interrupt request:
When transmit data is transferred from TDR to TSR, a transmit data empty interrupt (TXI) request is generated.
Write transmit data to TDR once in the TXI interrupt handling routine.

[3] Serial transmission continuation procedure:
To continue serial transmission, write transmit data to TDR once using a TXI interrupt request.
Transmit data can also be written to TDR by activating the DMAC.
When TEI interrupt requests are in use, set the SCR.TIE bit to 0 and the SCR.TEIE bit to 1 after the last of the data to be transmitted are written to the TDR

[4] Break output at the end of serial transmission:
To output a break in serial transmission, set the I/O port function corresponding to the TXDn pin (corresponding to output of the low level), and after switching the TXDn pin to the general I/O port function, set the TE bit in the SCR to 0.

Note: The TDR register becomes the TDRH and TDRL registers when 9-bit data length is selected. Write data in the order from TDRH to TDRL.

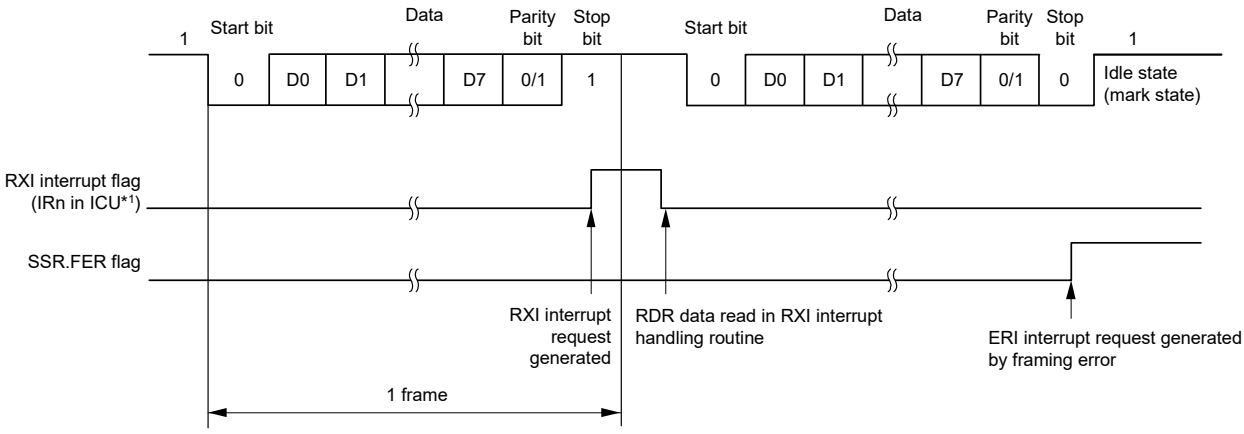
Figure 18.9 Example of Serial Transmission Flowchart in Asynchronous Mode

18.3.8 Serial Data Reception (Asynchronous Mode)

Figure 18.10 and Figure 18.11 show an example of the operation for serial data reception in asynchronous mode. In serial data reception, the SCI operates as described below.

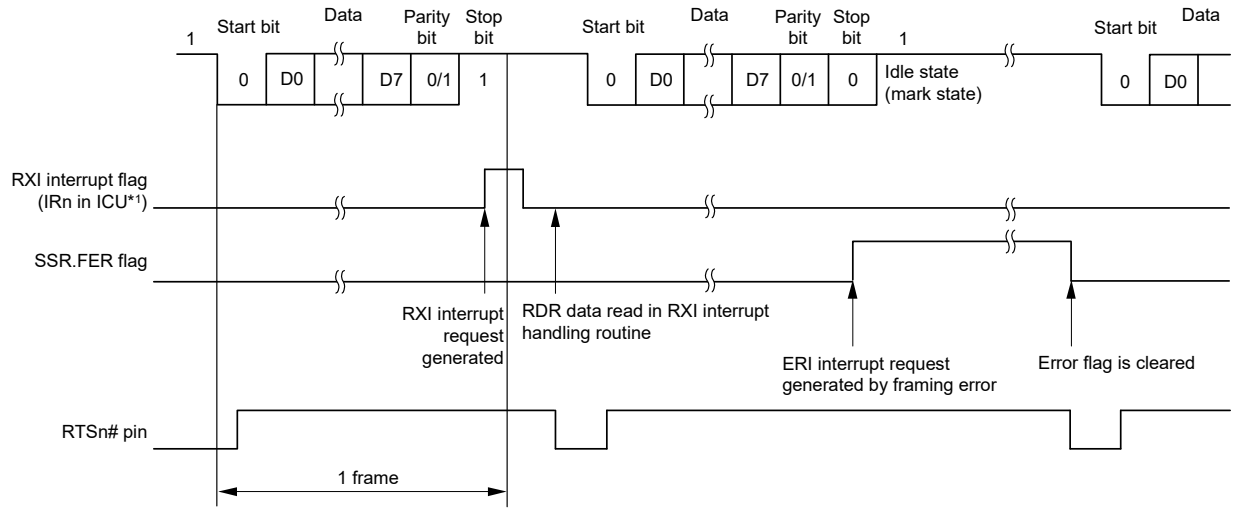
1. When the value of the RE bit in SCR becomes 1, the output signal on the RTSn# pin goes to the low level.
2. When the SCI monitors the communications line and detects a start bit, it performs internal synchronization, stores receive data in RSR, and checks the parity bit and stop bit.
3. If an overrun error occurs, the ORER flag in SSR is set to 1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR*1.
4. If a parity error is detected, the PER bit in SSR is set to 1 and receive data is transferred to RDR*1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated.
5. If a framing error (when the stop bit is 0) is detected, the FER bit in SSR is set to 1 and receive data is transferred to RDR*1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated.
6. When reception finishes successfully, receive data is transferred to RDR*1. If the RIE bit in SCR is 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to RDR*1 in this RXI interrupt handling routine before reception of the next receive data is completed. Reading the received data that have been transferred to RDR*1 causes the RTSn# pin to output the low level.

Note 1. Read data not in RDR but in the RDRH and RDRL registers when 9-bit data length is selected.
Note 2. The SCI checks for reading of the RDRL register only and does not check for reading of the RDRH register when 9-bit data length is selected.



Note 1. Refer to section 7, Interrupt Controller for details on the corresponding interrupt vector number.

Figure 18.10 Example of SCI Operation for Serial Reception in Asynchronous Mode (1) (When RTS Function is Not Used) (Example with 8-Bit Data, Parity, 1 Stop Bit)



Note 1. Refer to section 7, Interrupt Controller for details on the corresponding interrupt vector number.

Figure 18.11 Example of SCI Operation for Serial Reception in Asynchronous Mode (2) (When RTS Function is Used) (Example with 8-Bit Data, Parity, 1 Stop Bit)

Table 18.19 lists the states of the flags in the SSR status register and receive data handling when a receive error is detected.

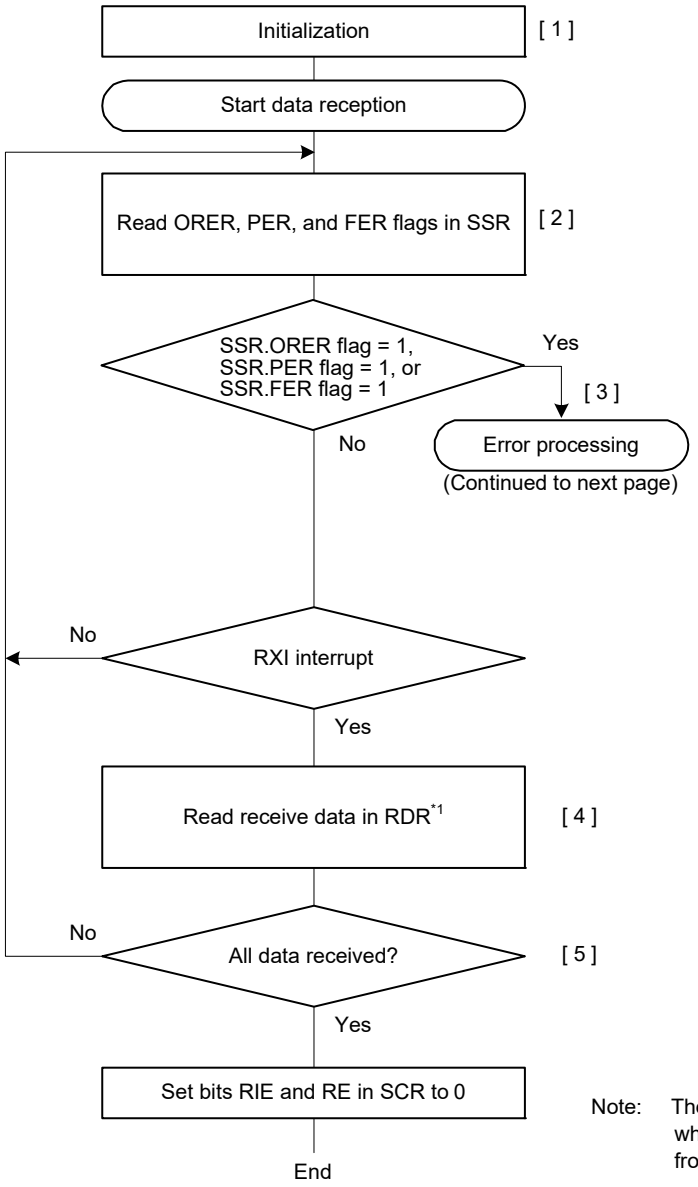
If a receive error is detected, an ERI interrupt request is generated but an RXI interrupt request is not generated. Data reception cannot be resumed while the receive error flag is 1. Accordingly, set the ORER, FER, and PER bits to 0 before resuming reception. Moreover, be sure to read the RDR (or the RDRL) during overrun error processing. When a reception is forcibly terminated by setting the SCR.RE bit to 0 during operation, read the RDR (or the RDRL) register because received data which has not yet been read may be left in RDR (or the RDRL).

Figure 18.12 and Figure 18.13 show samples of flowcharts for serial data reception.

Table 18.19 Flags in the SSR Status Register and Receive Data Handling

Flags in the SSR Status Register				
ORER	FER	PER	Receive Data	Receive Error Type
1	0	0	Lost	Overrun error
0	1	0	Transferred to RDR*1	Framing error
0	0	1	Transferred to RDR*1	Parity error
1	1	0	Lost	Overrun error + framing error
1	0	1	Lost	Overrun error + parity error
0	1	1	Transferred to RDR*1	Framing error + parity error
1	1	1	Lost	Overrun error + framing error + parity error

Note 1. Read data not in RDR but in the RDRH and RDRL registers when 9-bit data length is selected.



[1] SCI initialization:
Set data reception.

[2] [3] Receive error processing and break detection:
If a receive error occurs, an ERI interrupt is generated. An error is identified by reading the ORER, PER, and FER flags in SSR. After performing the appropriate error processing, be sure to set the ORER, PER, and FER flags to 0. Reception cannot be resumed if any of these flags is set to 1. In the case of a framing error, a break can be detected by reading the value of the input port corresponding to the RXDn pin.

[4] Read the receive data in RDR once in the RXI interrupt handling routine.

[5] Serial reception continuation procedure:
To continue serial reception, before the stop bit of the current frame is received, read data from RDR in the RXI interrupt processing routine. The RDR data can also be read by activating the DMAC.

Note: The RDR register becomes the RDRH and RDRL registers when 9-bit data length is selected. Read data in the order from RDRH to RDRL.

Figure 18.12 Example Flowchart of Serial Reception in Asynchronous Mode (1)

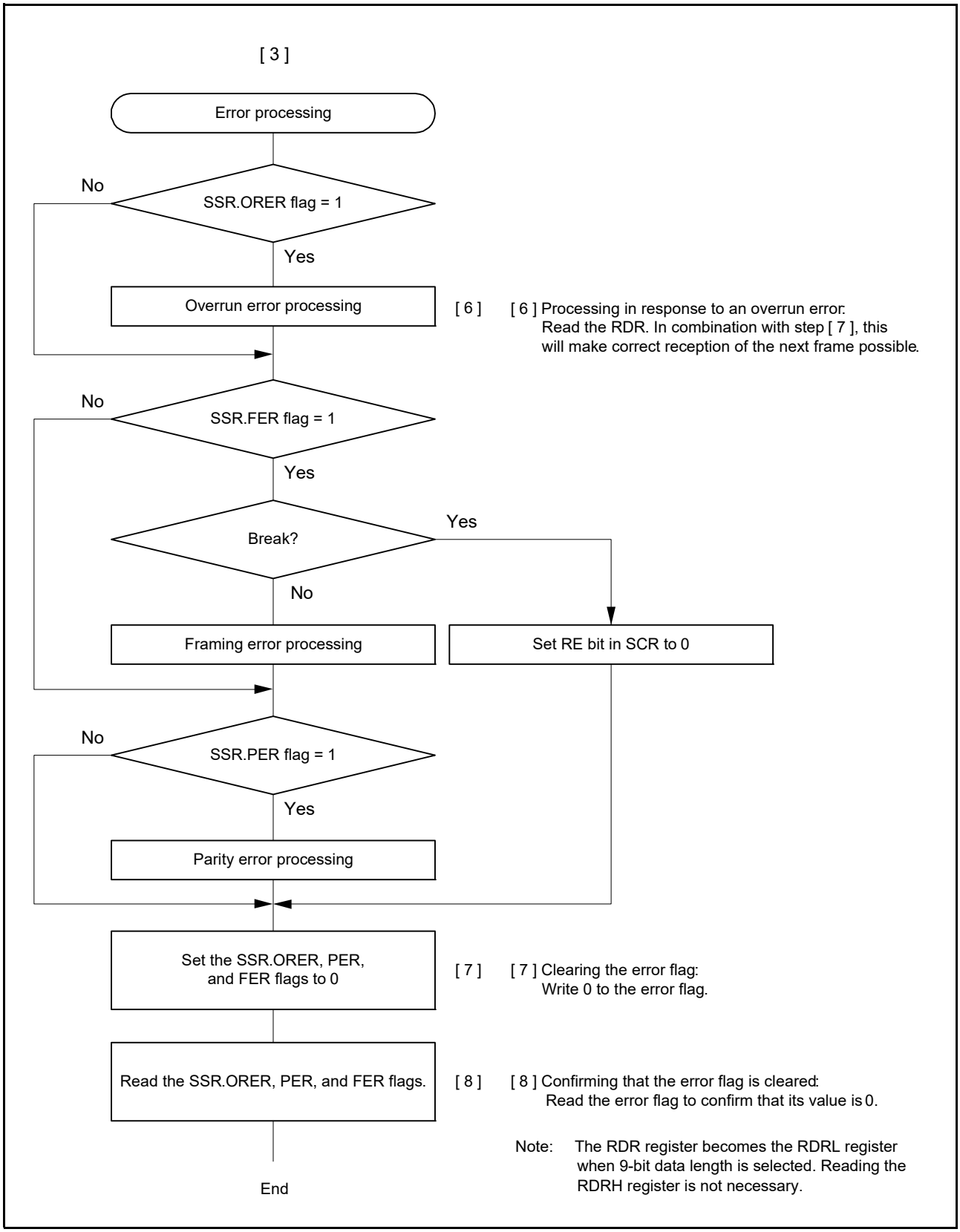


Figure 18.13 Example Flowchart of Serial Reception in Asynchronous Mode (2)

18.4 Multi-Processor Communications Function

Using the multi-processor communication functions enables to transmit and receive data by sharing a communication line between multiple processors by using asynchronous serial communication in which the multi-processor bit is added. In multi-processor communication, a unique ID code is allocated to each receiving station. Serial communication cycles consist of an ID transmission cycle to specify the receiving station and a data transmission cycle to transmit data to the specified receiving station. The multi-processor bit is used to distinguish between the ID transmission cycle and the data transmission cycle. When the multi-processor bit is set to 1, it indicates the ID transmission cycle and when the multi-processor bit is set to 0, it indicates the data transmission cycle. Figure 18.14 shows an example of communication between processors by using a multi-processor format. First, a transmitting station transmits communication data in which the multi-processor bit set to 1 is added to the ID code of the receiving station. Next, the transmitting station transmits the communication data in which the multi-processor bit set to 0 is added to the transmit data. Upon receiving the communication data in which the multi-processor bit is set to 1, the receiving station compares the received ID with the ID of the receiving station itself and if the two match, receives the communication data that is subsequently transmitted. If the received ID does not match with the ID of the receiving station, the receiving station skips the communication data until again receiving the communication data in which the multi-processor bit is set to 1.
For supporting this function, the SCI provides the MPIE bit in SCR. When the MPIE bit is set to 1, transfer of receive data from the RSR to the RDR (the RDRH and RDRL registers when 9-bit data length is selected), detection of a receive error, and setting the respective status flags ORER and FER in SSR are disabled until reception of data in which the multi-processor bit is set to 1. Upon receiving a reception character in which the multi-processor bit is set to 1, the MPBT bit in SSR is set to 1 and the MPIE bit in SCR is automatically cleared, thus returning to a normal reception operation. During this time, an RXI interrupt is generated if the RIE bit in SCR is set.
When the multi-processor format is specified, specification of the parity bit is disabled. Apart from this, there is no difference from the operation in the normal asynchronous mode. A clock which is used for the multi-processor communication is also the same as the clock used in the normal asynchronous mode.

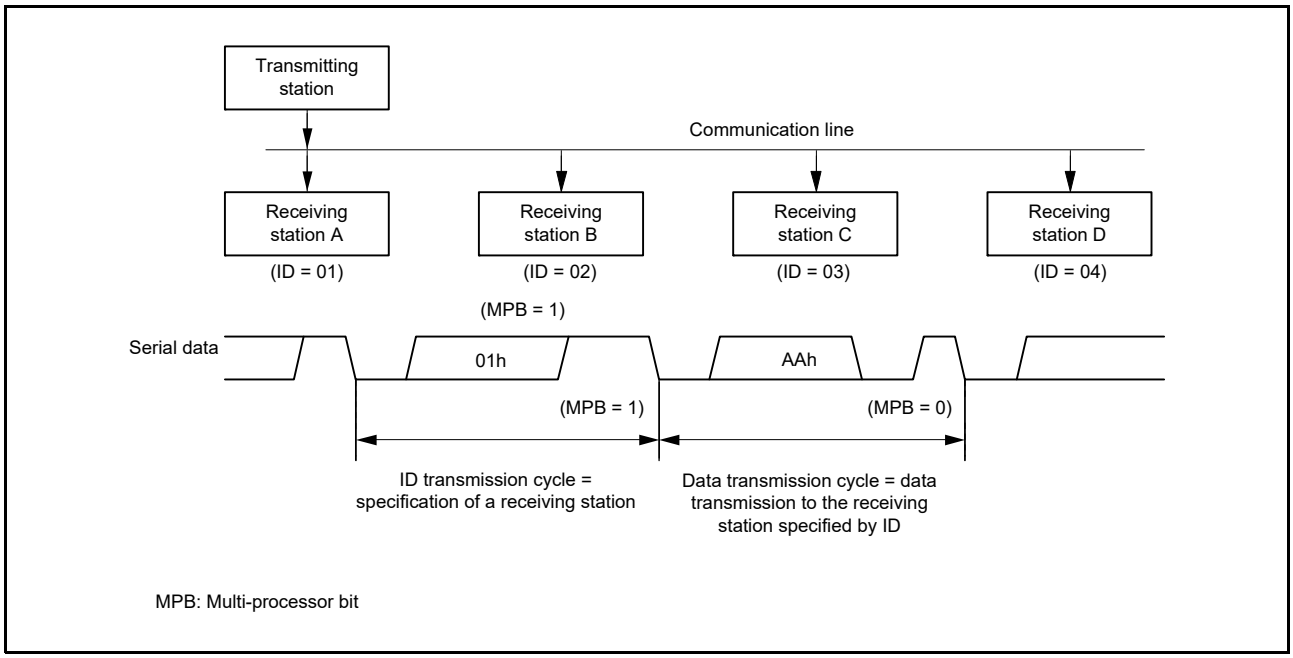


Figure 18.14 An Example of Communication using the Multi-Processor Format (Example of Transmission of Data AAh to Receiving Station A)

18.4.1 Multi-Processor Serial Data Transmission

Figure 18.15 is a sample flowchart of multi-processor data transmission. In the ID transmission cycle, the ID should be transmitted with the MPBT bit in SSR set to 1. In the data transmission cycle, the data should be transmitted with the MPBT bit set to 0. The other operations are the same as the operations in asynchronous mode.

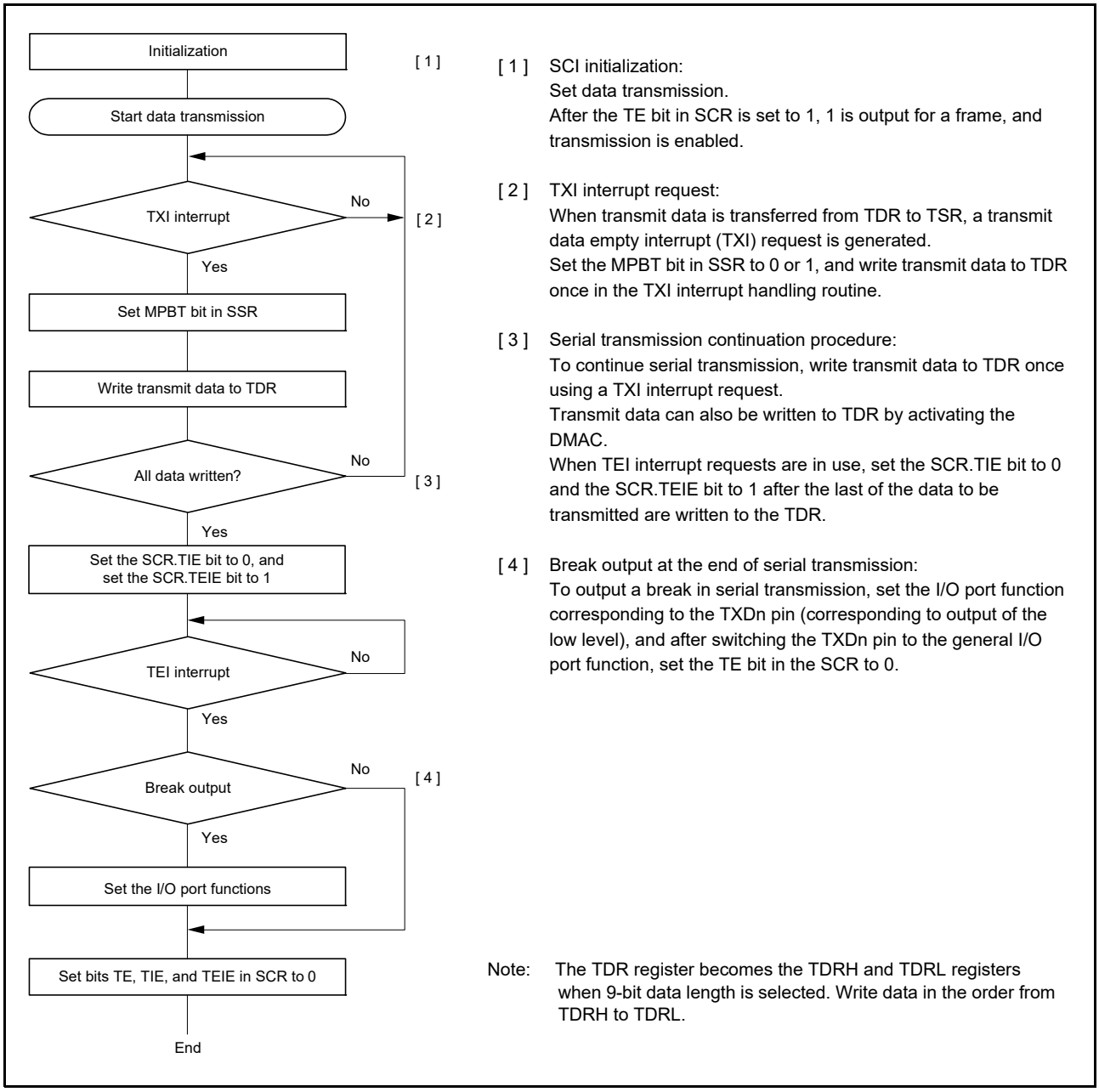
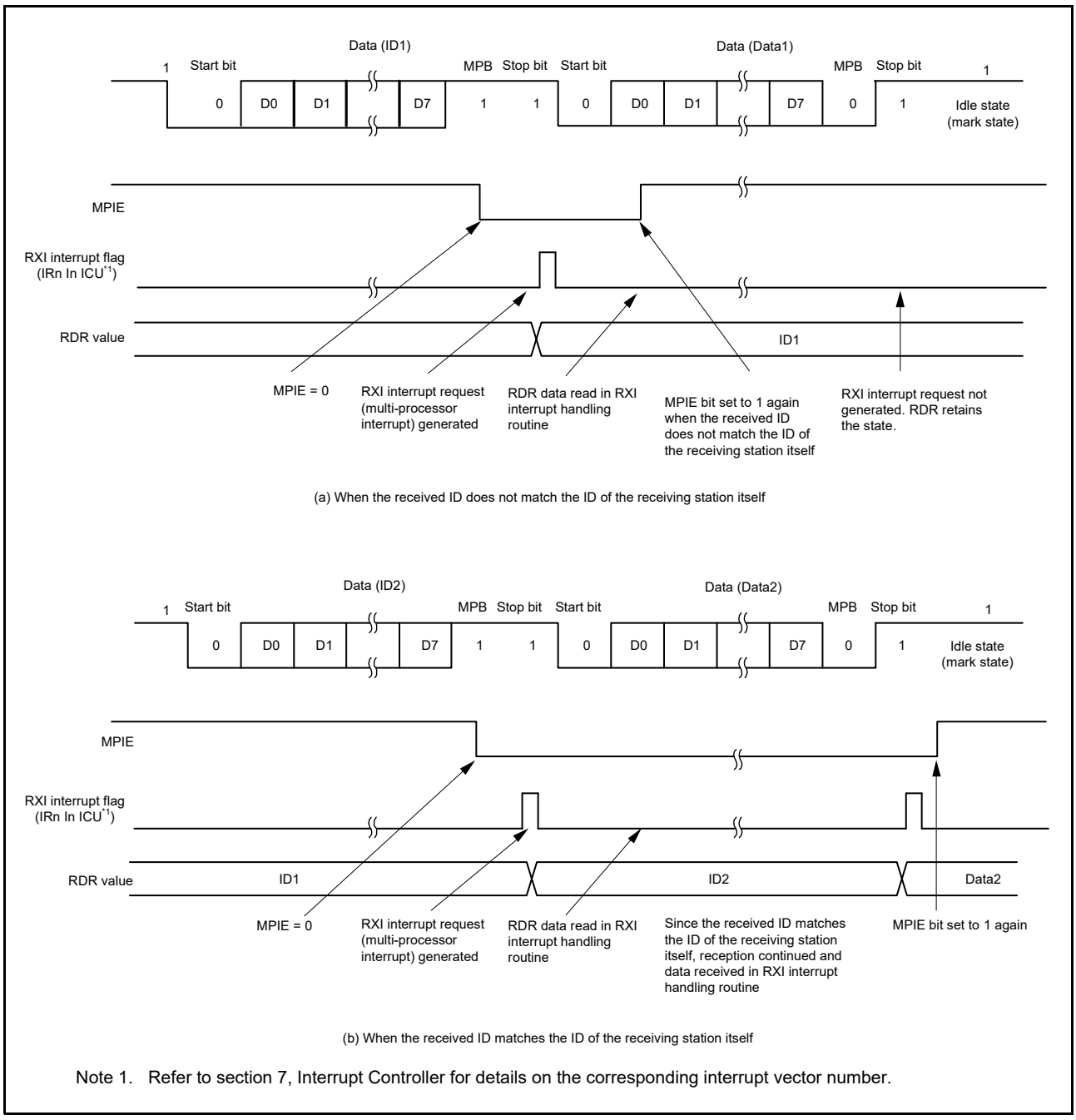


Figure 18.15 Example of Multi-Processor Serial Transmission Flowchart

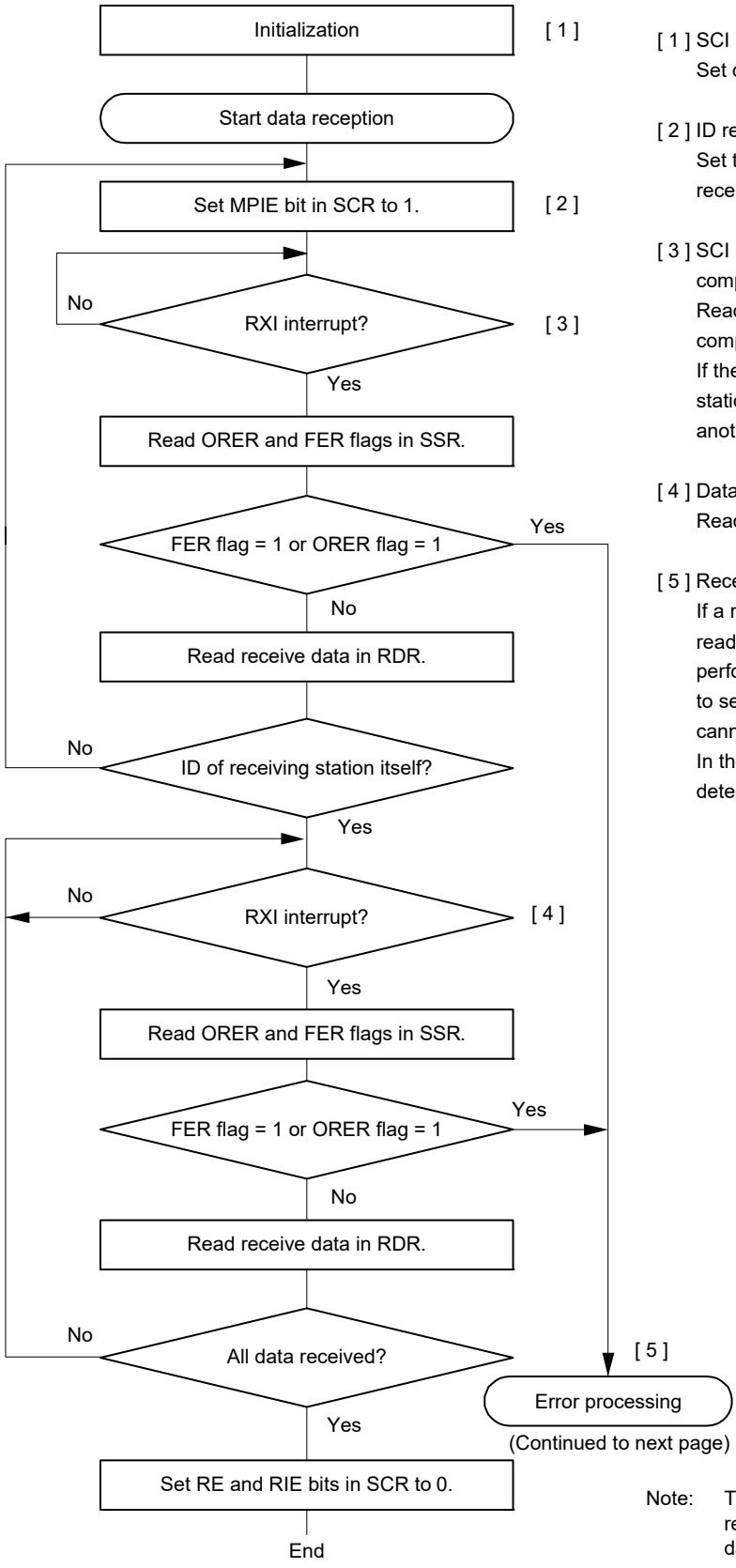
18.4.2 Multi-Processor Serial Data Reception

Figure 18.17 and Figure 18.18 are sample flowcharts of multi-processor data reception. When the MPIE bit in SCR is set to 1, reading the communication data is skipped until reception of the communication data in which the multi-processor bit is set to 1. When the communication data in which the multi-processor bit is set to 1 is received, the received data is transferred to RDR (the RDRH and RDRL registers when 9-bit data length is selected). During this time, the RXI interrupt request is generated. The other operations are the same as the operations in asynchronous mode. Figure 18.16 is the example of operation for reception.



Note 1. Refer to section 7, Interrupt Controller for details on the corresponding interrupt vector number.

Figure 18.16 Example of SCI Reception (8-Bit Data/Multi-Processor Bit/1 Stop Bit)



[1] SCI initialization:
Set data reception.

[2] ID reception cycle:
Set the MPIE bit in SCR to 1 and wait for ID reception.

[3] SCI status confirmation and reception and comparison of ID:
Read data in RDR at the first RXI interrupt, and compare it with the ID of the receiving station itself. If the ID does not match the ID of the receiving station itself, set the MPIE bit to 1 again, and wait for another RXI interrupt RXI interrupt request.

[4] Data reception at an RXI interrupt:
Read data in RDR once in the RXI interrupt routine.

[5] Receive error processing and break detection:
If a receive error occurs, an error is identified by reading the ORER and FER flags in SSR. After performing the appropriate error processing, be sure to set the ORER and FER flags to 0. Reception cannot be resumed if any of these flags is set to 1. In the case of a framing error, a break can be detected by reading the value of the RXDn pin.

Note: The RDR register becomes the RDRH and RDRL registers when 9-bit data length is selected. Read data in the order from RDRH to RDRL.

Figure 18.17 Example of Multi-Processor Serial Reception Flowchart (1)

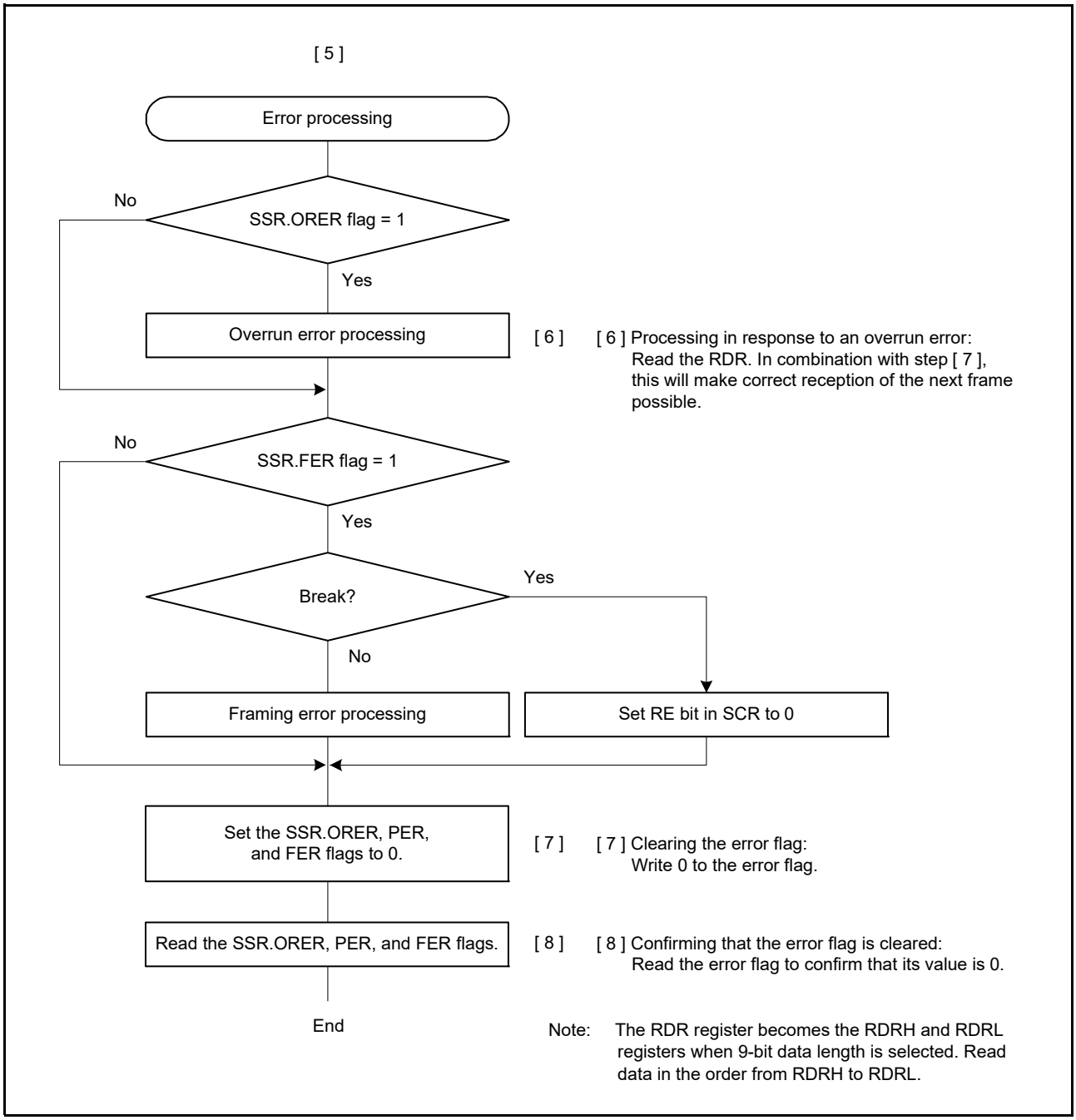


Figure 18.18 Example of Multi-Processor Serial Reception Flowchart (2)

18.5 Operation in Clock Synchronous Mode

Figure 18.19 shows the data format for clock synchronous serial data communications.
In clock synchronous mode, data is transmitted or received in synchronization with clock pulses. One character in transfer data consists of 8-bit data. In clock synchronous mode, no parity bit can be added.
In data transmission, the SCI outputs data from one falling edge of the synchronization clock to the next. In data reception, the SCI receives data in synchronization with the rising edge of the synchronization clock. After 8-bit data is output, the communication line holds the last bit output state.
Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communications by use of a common clock. Both the transmitter and the receiver also have a double-buffered structure, so that the next transmit data can be written during transmission or the previous receive data can be read during reception, enabling continuous data transfer.

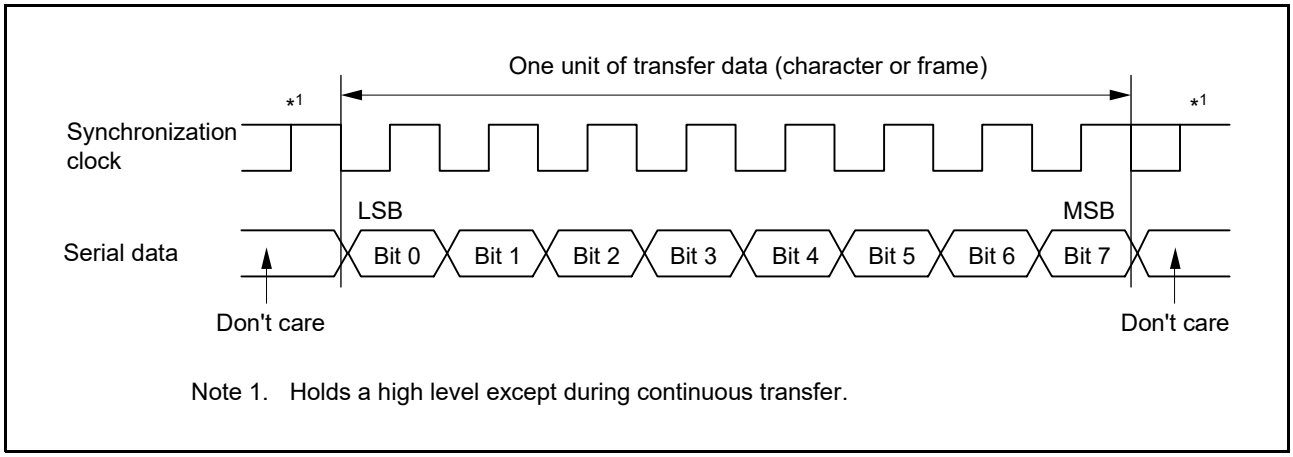


Figure 18.19 Data Format in Clock Synchronous Serial Communications (LSB First)

18.5.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external synchronization clock input at the SCKn pin can be selected, according to the setting of the SCR.CKE[1:0] bits.
When the SCI is operated on an internal clock, the synchronization clock is output from the SCKn pin. Eight synchronization clock pulses are output in the transfer of one character, and when no transfer is performed the clock is held high. However, when only data reception is performed while the CTS function is disabled, the synchronization clock output is started at the same time when the SCR.RE bit set to 1. The synchronization clock is stopped at the high level when an overrun error occurs or the SCR.RE bit is set to 0.
When only data reception is performed and the CTS function is enabled, the clock output is not started even when the SCR.RE bit set to 1 if the CTSn# pin input is high when the SCR.RE bit is 0. The synchronization clock output is started when the SCR.RE bit is set to 1 and the CTSn# pin input is low. After that, if the CTSn# pin input is high on completion of the frame reception, the synchronization clock output is stopped at the high level. If the CTSn# pin input continues to be low, the synchronization clock is stopped at the high level when an overrun error occurs or the SCR.RE bit is set to 0.

18.5.2 CTS and RTS Functions

In the CTS function, CTSn# pin input is used to control reception/transmission start when the clock source is the internal clock. Setting the SECR.CTSE bit to 1 enables the CTS function.
When the CTS function is enabled, placing the low level on the CTSn# pin causes reception/transmission to start.
In the RTS function, RTSn# pin output is used to request reception/transmission start when the clock source is an external synchronizing clock. A low level is output when serial communications become possible. Conditions for output of the low and high level are shown below.

[Conditions for low-level output]
Satisfaction of all conditions listed below

- The value of the RE or TE bit in the SCR is 1
- Neither transmission nor reception is in progress
- There are no received data yet to be read (when the SCR.RE bit is 1)
- Transmit data has been written (when the SCR.TE bit is 1)
- ORER flag in SSR is 0

[Condition for high-level output]
The conditions for low-level output have not been satisfied.

18.5.3 SCI Initialization (Clock Synchronous Mode)

Before transmitting and receiving data, start by writing the initial value 00h to the SCR and then continue through the procedure for SCI given in Figure 18.20. Whenever the operating mode or transfer format is changed, the SCR must be initialized before the change is made.
Note that setting the SCR.RE bit to 0 initializes neither the ORER, FER, and PER flags in SSR nor RDR.
Moreover, note that switching the value of the SCR.TE bit from 1 to 0 or 0 to 1 while the SCR.TIE bit is 1 leads to the generation of a TXI interrupt request.

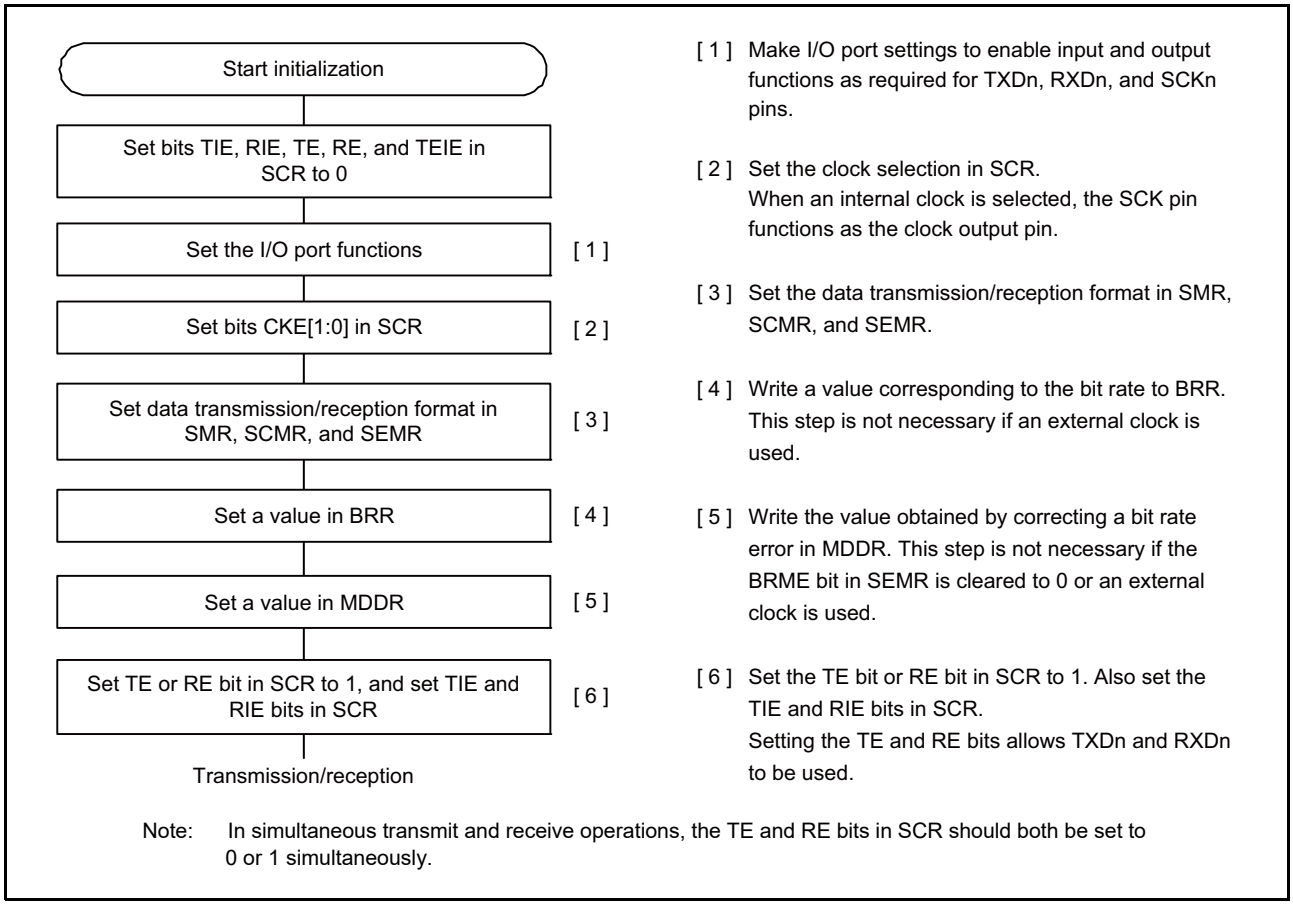


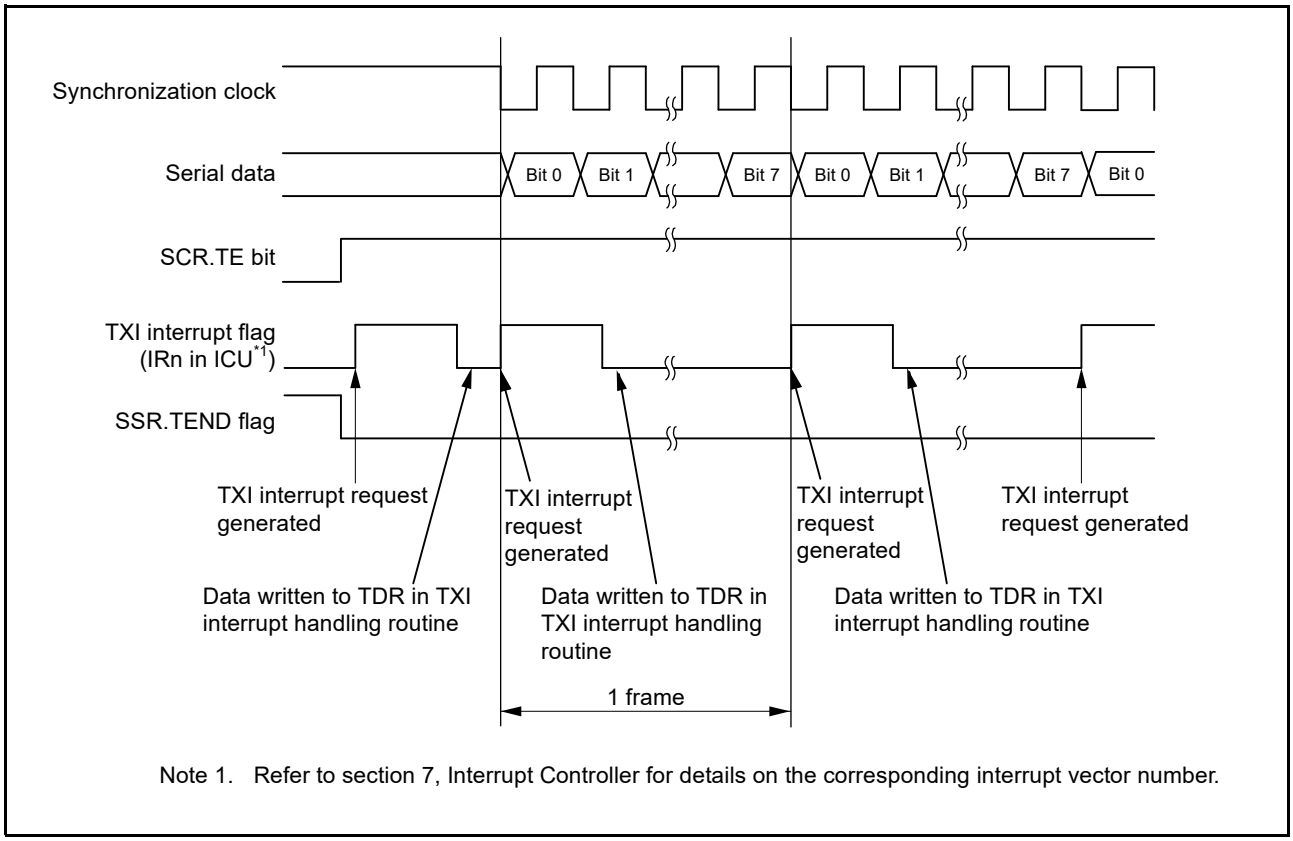
Figure 18.20 Example of SCI Initialization Flowchart (Clock Synchronous Mode)

18.5.4 Serial Data Transmission (Clock Synchronous Mode)

Figure 18.20, Figure 18.21, and Figure 18.22 show an example of the operation for serial transmission in clock synchronous mode.
In serial data transmission, the SCI operates as described below.

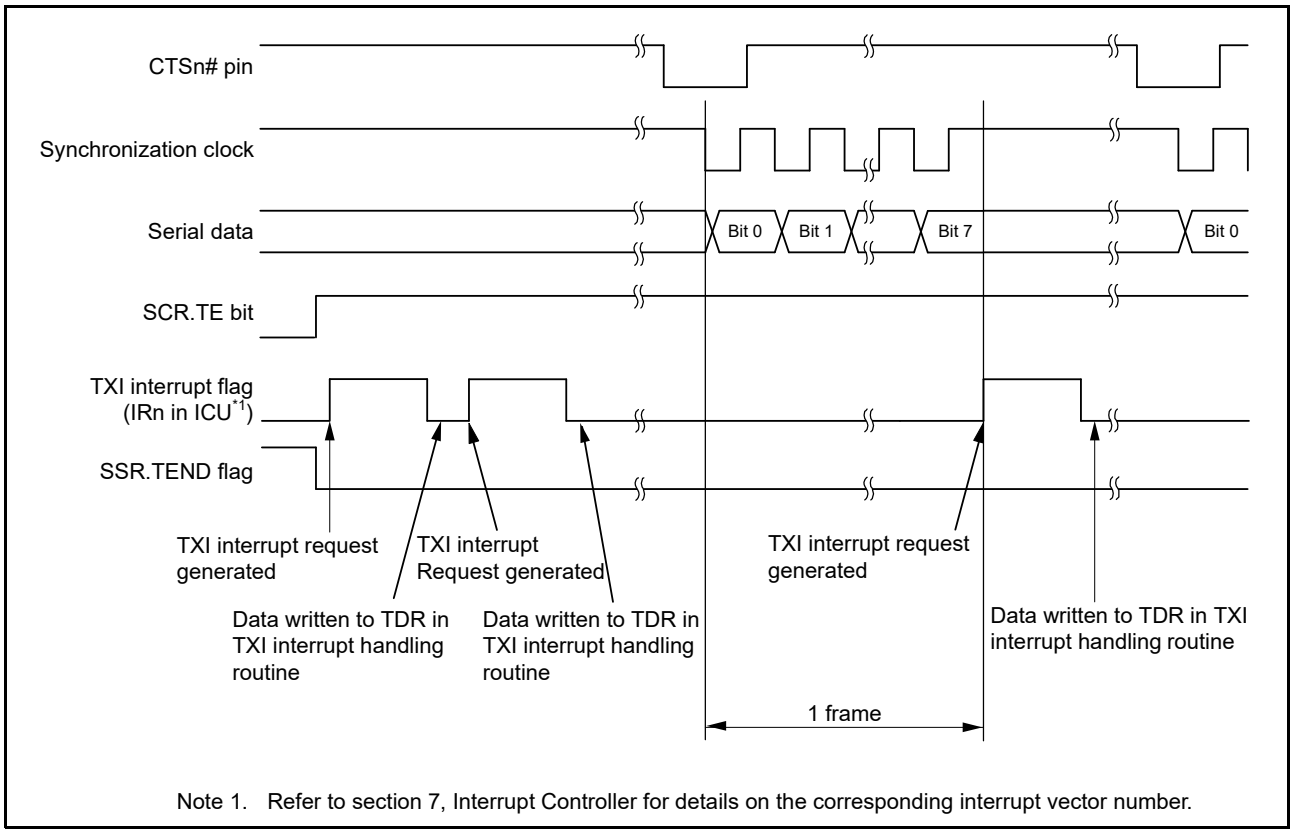
1. The SCI transfers data from TDR to TSR when data is written to TDR in the TXI interrupt handling routine. The TXI interrupt request at the beginning of transmission is generated when the TE bit in SCR is set to 1 after the TIE bit in SCR is set to 1 or when these 2 bits are set to 1 simultaneously by a single instruction.
2. After transferring data from TDR to TSR, the SCI starts transmission. When the SCR.TIE bit is set to 1 at this time, a TXI interrupt request is generated. Continuous transmission is enabled by writing the next transmit data to TDR in this TXI interrupt handling routine before transmission of the current transmit data has finished. When TEI interrupt requests are in use, set the SCR.TIE bit to 0 (a TXI interrupt request is disabled) and the SCR.TEIE bit to 1 (a TEI interrupt request is enabled) after the last of the data to be transmitted are written to the TDR from the handling routine for TXI requests.
3. 8-bit data is sent from the TXDn pin in synchronization with the output clock when clock output mode has been specified and in synchronization with the input clock when use of an external clock has been specified. Output of the clock signal is suspended until the input CTS signal is at the low level while the CTSE bit in SECR is 1 (CTS function is enabled).
4. The SCI checks for updating of (writing to) the TDR at the time of the last bit output.
5. When TDR is updated, the next transmit data is transferred from TDR to TSR, and serial transmission of the next frame is started.
6. If TDR is not updated, set the SSR.TEND flag to 1 and the TXDn pin retains the output state of the last bit. If the TEIE bit in SCR is 1 at this time, a TEI interrupt request is generated. The SCKn pin is held high.

Figure 18.24 shows a sample flowchart of serial data transmission.
Transmission will not start while a receive error flag (ORER, FER, or PER in SSR) is set to 1. Be sure to set the receive error flags to 0 before starting transmission. Note that setting the RE bit in SCR to 0 does not clear the receive error flags.



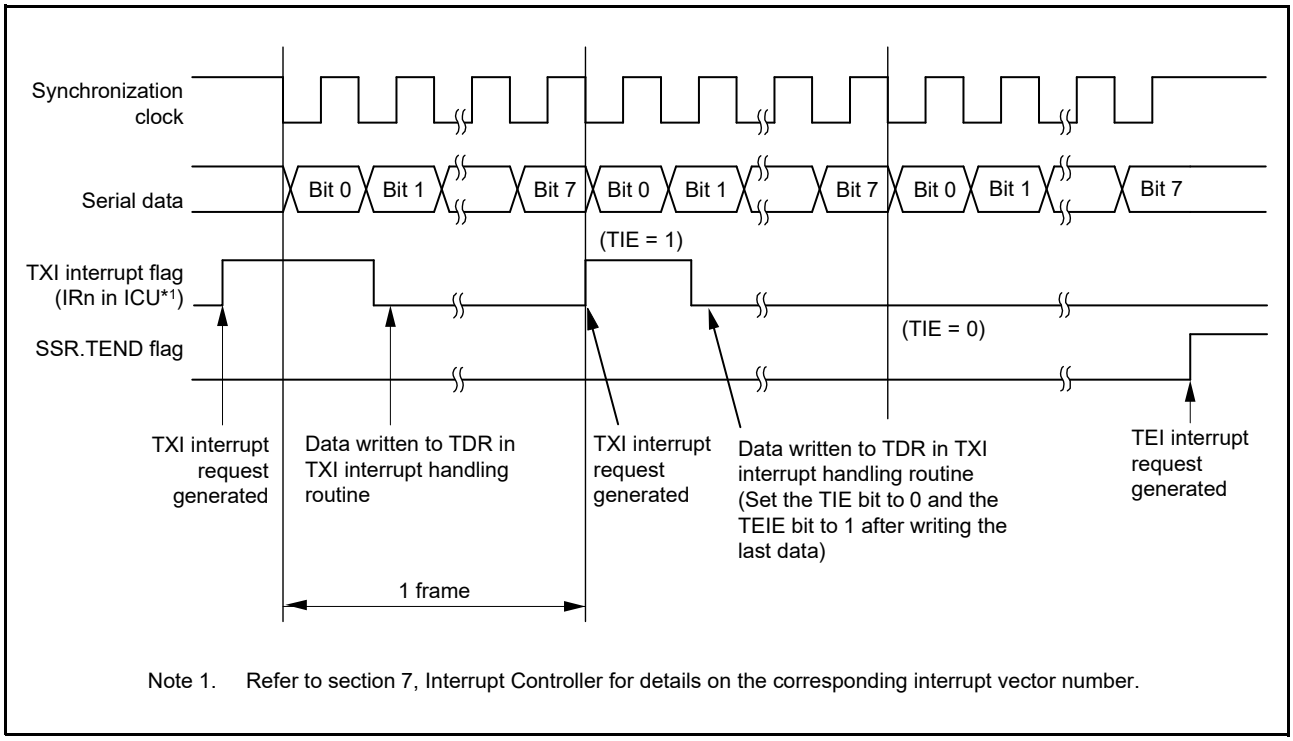
Note 1. Refer to section 7, Interrupt Controller for details on the corresponding interrupt vector number.

Figure 18.21 Example of Serial Data Transmission in Clock Synchronous Mode (1)
(When the CTS Function is Not Used at the Beginning of Transmission)



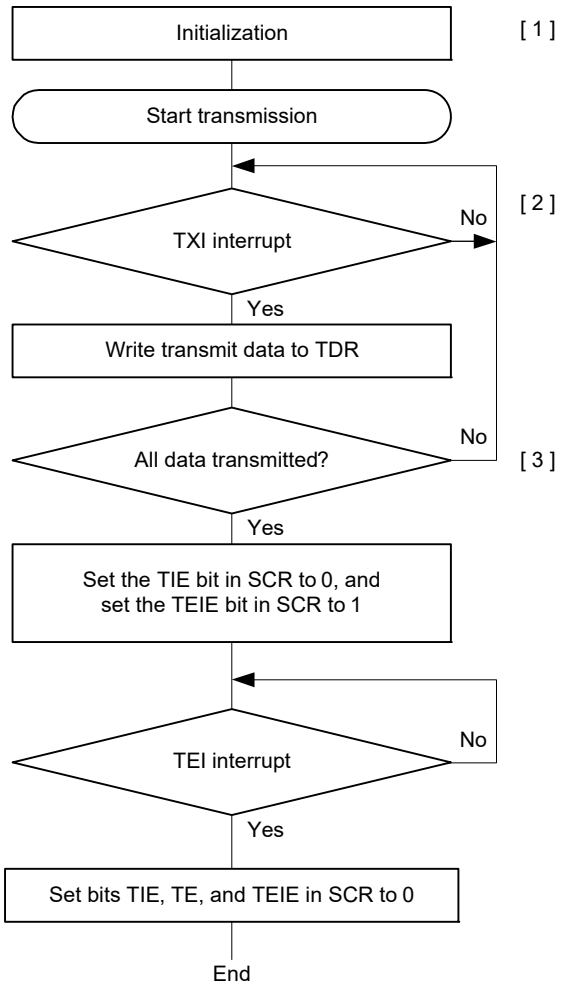
Note 1. Refer to section 7, Interrupt Controller for details on the corresponding interrupt vector number.

Figure 18.22 Example of Serial Data Transmission in Clock Synchronous Mode (2)
(When the CTS Function is Used at the Beginning of Transmission)



Note 1. Refer to section 7, Interrupt Controller for details on the corresponding interrupt vector number.

Figure 18.23 Example of Serial Data Transmission in Clock Synchronous Mode (3)
(From the Middle of Transmission until Transmission Completion)



[1] SCI initialization:
Set data transmission.

[2] Writing transmit data write to TDR by a TXI interrupt request:
When transmit data is transferred from TDR to TSR, a transmit data empty interrupt (TXI) request is generated.
Transmit data is written to TDR once from the handling routine for TXI requests.

[3] Serial transmission continuation procedure:
To continue serial transmission, write transmit data to TDR upon accepting a transmit data empty interrupt (TXI) request. Transmit data can also be written to TDR by activating the DMAC by the TXI interrupt request.
When TEI interrupt requests are in use, set the SCR.TIE bit to 0 and the SCR.TEIE bit to 1 after the last of the data to be transmitted are written to the TDR.

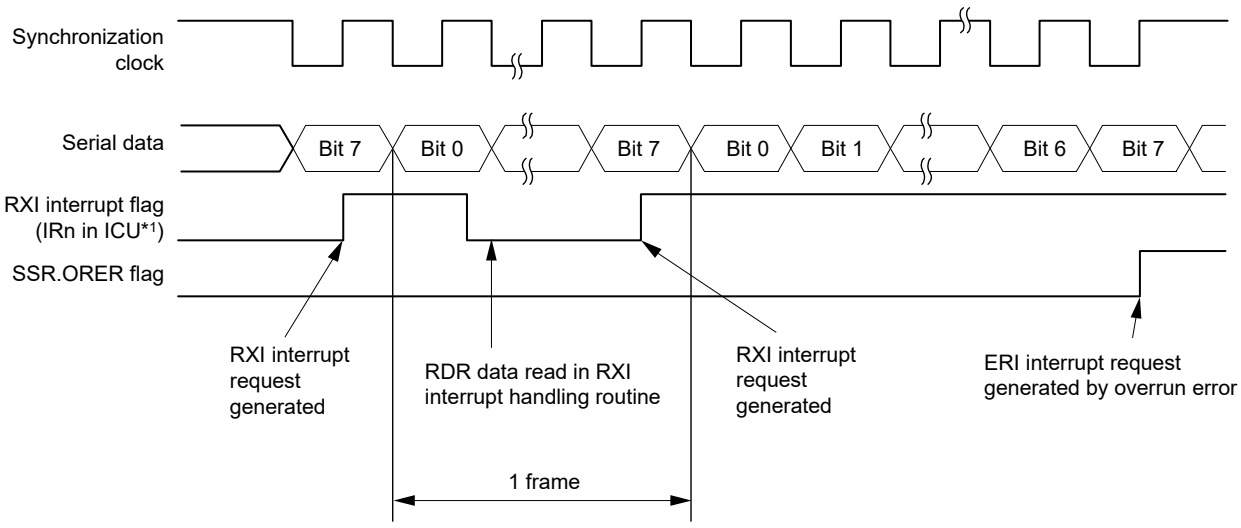
Note: When the external clock is in use (the value of the SCR.CKE[1:0] bits is 10b or 11b), the rising edge on the SCK pin for the last bit sets the SSR.TEND flag to 1. Setting the SCR.TE bit to 0 immediately after this may lead to insufficient received-data hold time on the receiver side.

Figure 18.24 Example Flowchart of Serial Transmission in Clock Synchronous Mode

18.5.5 Serial Data Reception (Clock Synchronous Mode)

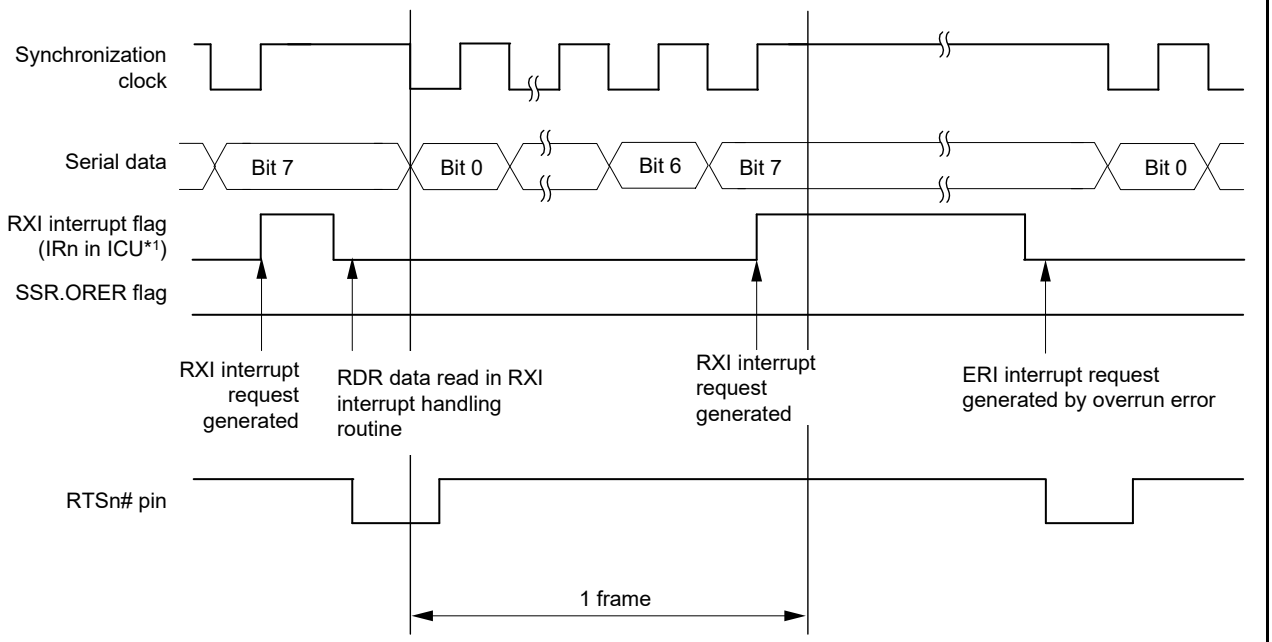
Figure 18.25 and Figure 18.26 show an example of SCI operation for serial reception in clock synchronous mode. In serial data reception, the SCI operates as described below.

1. The value of the RE bit in SCR becoming 1 places the signal output on the RTSn# pin at the low level (when the RTS function is in use).
2. The SCI performs internal initialization and starts receiving data in synchronization with a synchronization clock input or output, and stores the receive data in RSR.
3. If an overrun error occurs, the ORER bit in SSR is set to 1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR.
4. When reception finishes successfully, receive data is transferred to RDR. If the RIE bit in SCR is 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to RDR in this RXI interrupt handling routine before reception of the next receive data is completed. Reading out the received data that have been transferred to RDR causes the RTSn# pin to output the low level (when the RTS function is in use).



Note 1. Refer to section 7, Interrupt Controller for details on the corresponding interrupt vector number.

Figure 18.25 Example of Operation for Serial Reception in Clock Synchronous Mode (1) (When RTS Function is Not Used)

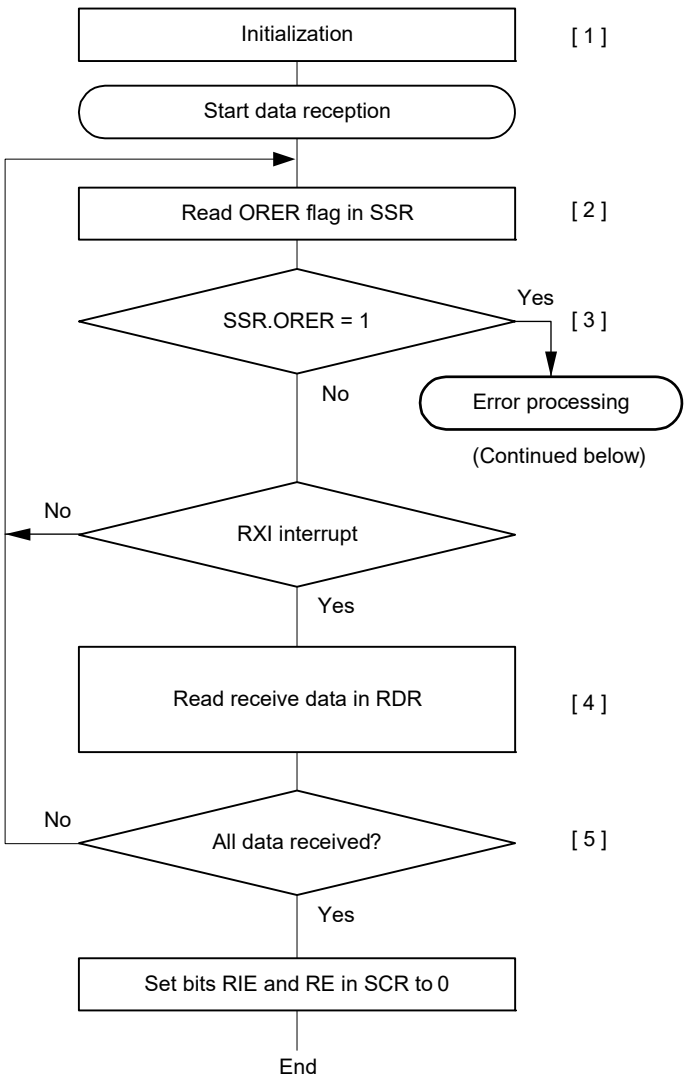


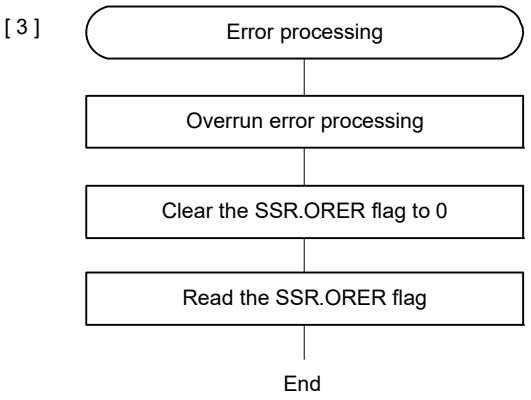
Note 1. Refer to section 7, Interrupt Controller for details on the corresponding interrupt vector number.

Figure 18.26 Example of Operation for Serial Reception in Clock Synchronous Mode (2) (When RTS Function is Used)

Data transfer cannot be resumed while a receive error flag is 1. Accordingly, clear the ORER, FER, and PER bits in SSR to 0 before resuming reception. Moreover, be sure to read the RDR during overrun error processing. When a reception is forcibly terminated by setting the SCR.RE bit to 0 during operation, read RDR because received data which has not yet been read may be left in RDR.

Figure 18.27 shows a sample flowchart for serial data reception.





[1] SCI initialization:
Make input port-pin settings for pins to be used as RXDn pins.

[2] [3] Receive error processing:
If a receive error occurs, read the ORER flag in SSR, perform the relevant error processing, and then set the ORER flag to 0. Data reception cannot be resumed while the ORER flag is 1.

[4] Read the receive data in RDR once in the receive data full interrupt (RXI) request handling routine.

[5] Serial reception continuation procedure:
To continue serial reception, before the MSB (bit 7) of the current frame is received, finish reading the receive data in RDR. The RDR data can also be read by activating the DMAC by an RXI interrupt request.

[6] Processing in response to an overrun error:
Read the RDR. In combination with step [7], this will make correct reception of the next frame possible.

[7] Clearing the error flag:
Write 0 to the error flag.

[8] Confirming that the error flag is cleared:
Read the error flag to confirm that its value is 0.

Figure 18.27 Example Flowchart of Serial Reception in Clock Synchronous Mode

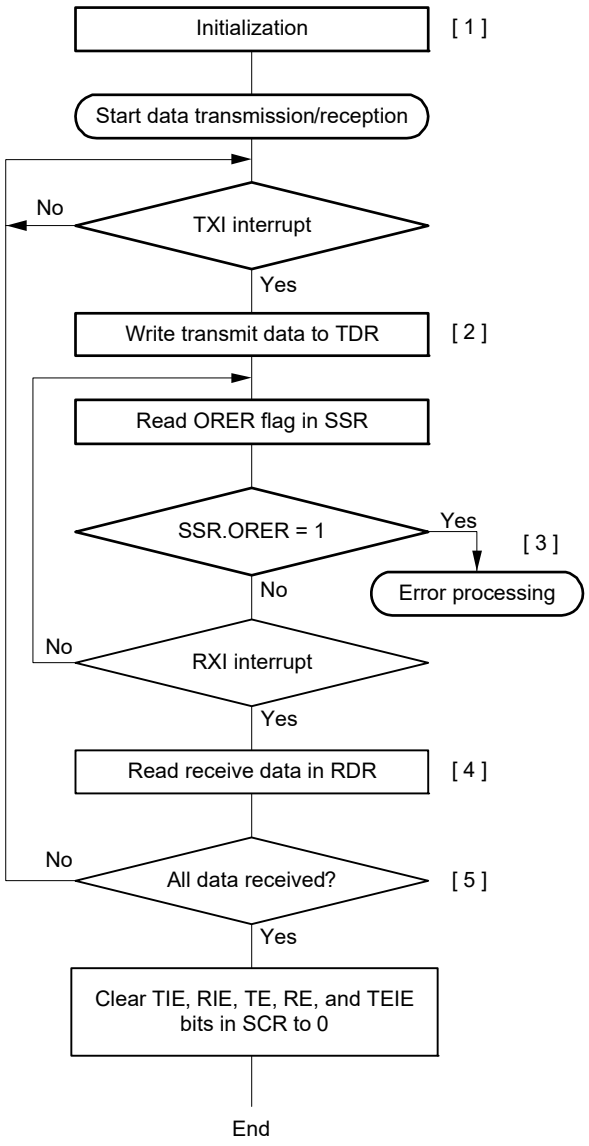
18.5.6 Simultaneous Serial Data Transmission and Reception (Clock Synchronous Mode)

Figure 18.28 shows a sample flowchart for simultaneous serial transmit and receive operations in clock synchronous mode.

After initializing the SCI, the following procedure should be used for simultaneous serial data transmit and receive operations.

To switch from transmit mode to simultaneous transmit and receive mode, check that the SCI has finished transmission by reading that the TEND flag in SSR is 1, and then initialize the SCR register. Then set the TIE, RIE, TE, and RE bits in SCR to 1 simultaneously by a single instruction.

To switch from receive mode to simultaneous transmit and receive mode, check that the SCI has finished reception, and then set the RIE and RE bits to 0. Then check that the receive error flags (ORER, FER, and PER in SSR) are 0, and then set the TIE, RIE, TE, and RE bits in SCR to 1 simultaneously by a single instruction.



[1] SCI initialization:
The TXDn pin can act as the output pin for transmitted data and the RXDn pin can act as the input pin for received data at the same time.

[2] Transmit data write:
Write transmit data to TDR once in the TXI interrupt handling routine.

[3] Receive error processing:
If a receive error occurs, read the ORER flag in SSR, perform the relevant error processing, and then set the ORER flag to 0. Data reception cannot be resumed while the ORER flag is 1.

[4] Reading receive data:
Read the receive data in RDR once in the RXI interrupt request handling routine.

[5] Serial transmission/reception continuation procedure:
To continue serial transmission and reception, before the MSB (bit 7) of the current frame is received, finish reading the receive data in RDR by the RXI interrupt. Also, before the MSB (bit 7) of the current frame is transmitted, write data to TDR by the TXI interrupt.
Transmit data can also be written to TDR by activating the DMAC by a transmit data empty interrupt (TXI) request. Similarly, the RDR data can also be read by activating the DMAC by a receive data full interrupt (RXI) request.

Note: When switching from transmit or receive operation to simultaneous transmit and receive operations first set the TIE, RIE, TE, RE, and TEIE bits in SCR to 0, and then set TIE, RIE, TE, and RE bits to 1 simultaneously.

Figure 18.28 Example Flowchart of Simultaneous Serial Transmission and Reception in Clock Synchronous Mode

18.6 Operation in Smart Card Interface Mode

The SCI supports smart card (IC card) interfaces conforming to ISO/IEC 7816-3 (standard for Identification Cards), as an extended function of the SCI.
Smart card interface mode can be selected using the appropriate register.

18.6.1 Sample Connection

Figure 18.29 shows a sample connection between a smart card (IC card) and this MCU.
As in the figure, since this MCU communicates with an IC card using a single transmission line, interconnect the TXDn and RXDn pins and pull up the data transmission line to VCC using a resistor.
Setting the TE and RE bits in SCR to 1 with an IC card disconnected enables closed transmission/reception allowing self-diagnosis.
To supply an IC card with the clock pulses generated by the SCI, input the SCKn pin output to the CLK pin of an IC card.
The output port of the this MCU can be used to output a reset signal.

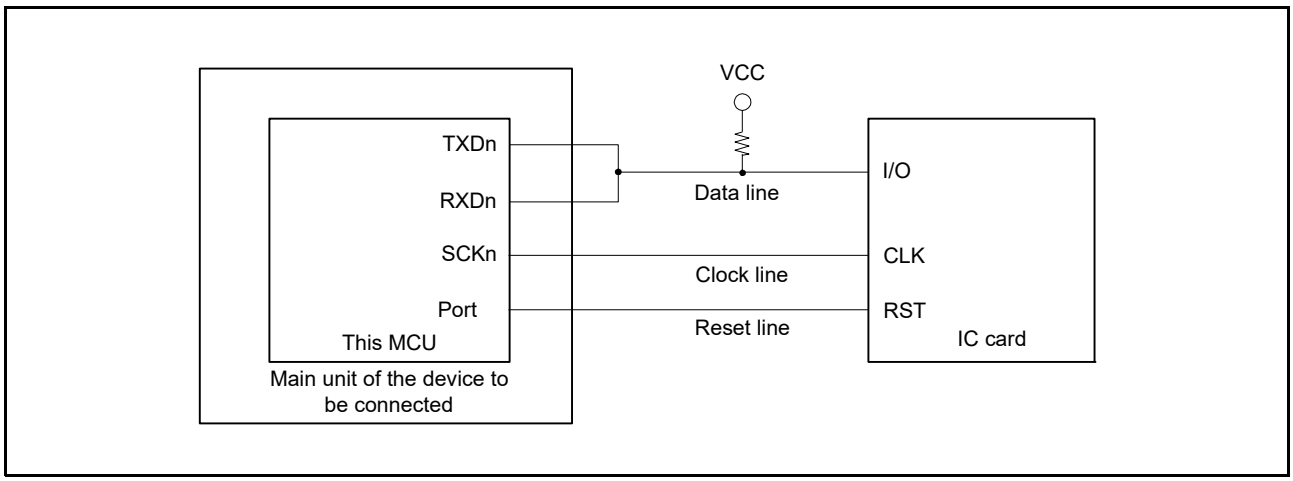


Figure 18.29 Sample Connection with a Smart Card (IC Card)

18.6.2 Data Format (Except in Block Transfer Mode)

Figure 18.30 shows the data transfer formats in smart card interface mode.

- One frame consists of 8-bit data and a parity bit in asynchronous mode.
- During transmission, at least 2 etu (elementary time unit: time required for transferring 1 bit) is secured as a guard time from the end of the parity bit until the start of the next frame.
- If a parity error is detected during reception, a low-level error signal is output for 1 etu after 10.5 etu has passed from the start bit.
- If an error signal is sampled during transmission, the same data is automatically retransmitted after at least 2 etu.

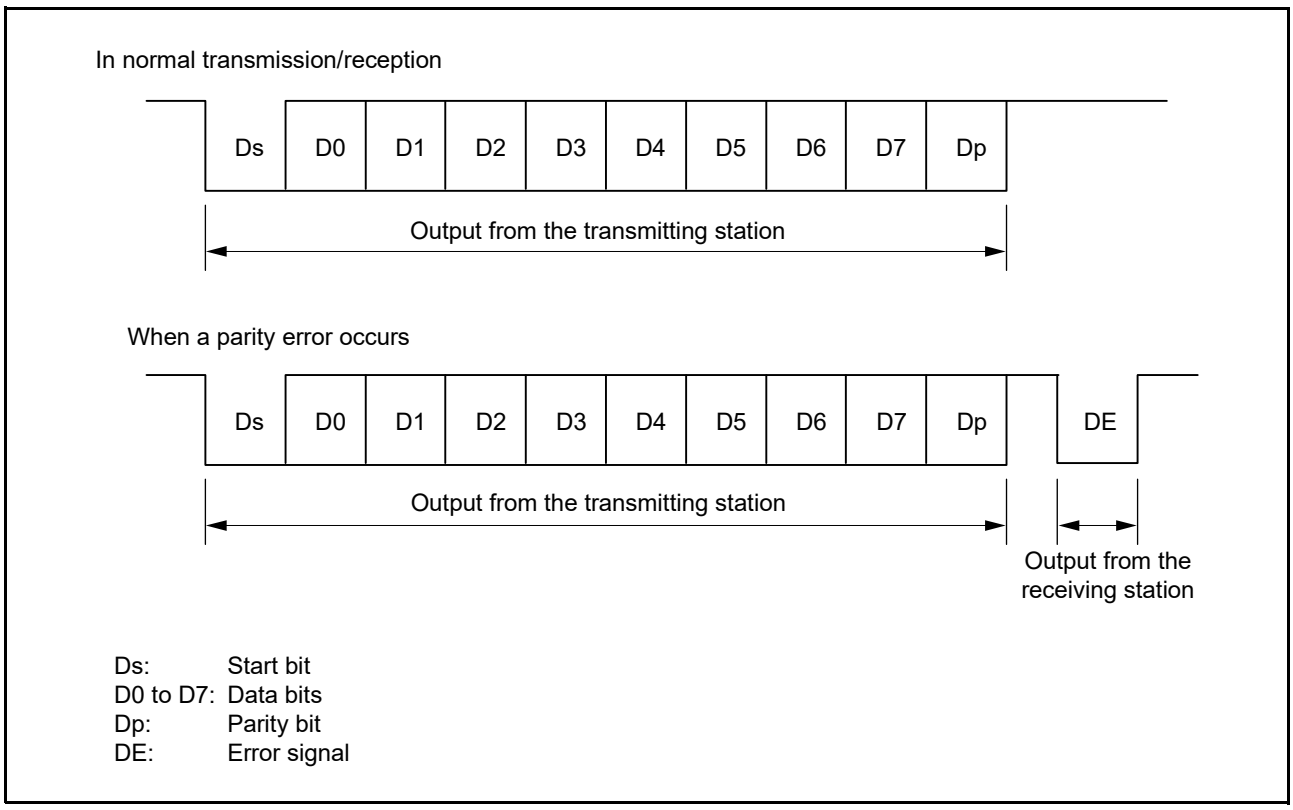


Figure 18.30 Data Formats in Smart Card Interface Mode

For communications with IC cards of the direct convention type and inverse convention type, follow the procedure below.

(1) Direct Convention Type

For the direct convention type, logic levels 1 and 0 correspond to states Z and A, respectively, and data is transferred with LSB first as the start character, as shown in Figure 18.31. Therefore, data in the start character in the figure is 3Bh. When using the direct convention type, write 0 to both the SDIR and SINV bits in SCMR. Write 0 to the PM bit in SMR in order to use even parity, which is prescribed by the smart card standard.

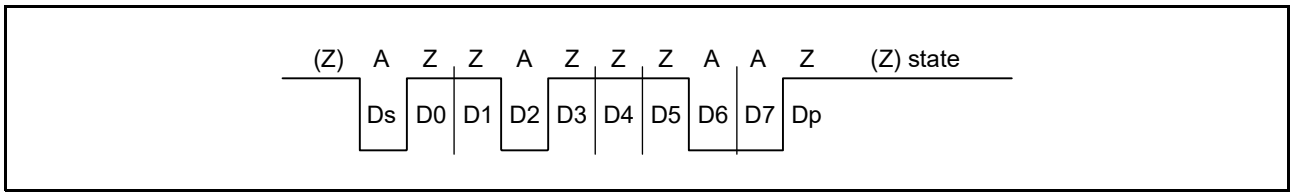


Figure 18.31 Direct Convention (SDIR in SCMR = 0, SINV in SCMR = 0, PM in SMR = 0)

(2) Inverse Convention Type

For the inverse convention type, logic levels 1 and 0 correspond to states A and Z, respectively and data is transferred with MSB first as the start character, as shown in Figure 18.32. Therefore, data in the start character in the figure is 3Fh. When using the inverse convention type, write 1 to both the SDIR and SINV bits in SCMR. The parity bit is logic level 0 to produce even parity, which is prescribed by the smart card standard, and corresponds to state Z. Since the SINV bit of the this MCU only inverts data bits D7 to D0, write 1 to the PM bit in SMR to invert the parity bit for both transmission and reception.

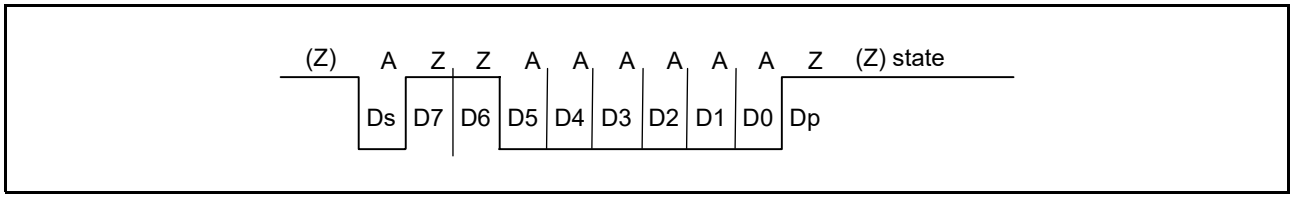


Figure 18.32 Inverse Convention (SDIR in SCMR = 1, SINV in SCMR = 1, PM in SMR = 1)

18.6.3 Block Transfer Mode

Block transfer mode is different from normal smart card interface mode in the following respects.

- Even if a parity error is detected during reception, no error signal is output. Since the PER bit in SSR is set by error detection, clear the PER bit before receiving the parity bit of the next frame.
- During transmission, at least 1 etu is secured as a guard time from the end of the parity bit until the start of the next frame.
- Since the same data is not retransmitted during transmission, the TEND flag in SSR is set 11.5 etu after transmission start.
- In block transfer mode, the ERS flag in SSR indicates the error signal status as in normal smart card interface mode, but the flag is read as 0 because no error signal is transferred.

18.6.4 Receive Data Sampling Timing and Reception Margin

Only the internal clock generated by the on-chip baud rate generator can be used as a transfer clock in smart card interface mode.

In this mode, the SCI can operate on a base clock with a frequency of 32, 64, 372, 256, 93, 128, 186, or 512 times the bit rate according to the settings of the BCP2 bit in SCMR and the BCP[1:0] bits in SMR (the frequency is always 16 times the bit rate in normal asynchronous mode).

For data reception, the falling edge of the start bit is sampled with the base clock to perform internal synchronization. Receive data is sampled on the 16th, 32nd, 186th, 128th, 46th, 64th, 93rd, and 256th rising edges of the base clock so that it can be latched at the middle of each bit as shown in Figure 18.33. The reception margin here is determined by the following formula.

$$M = \left| (0.5 - \frac{1}{2N}) - (L - 0.5)\ F - \frac{D - 0.5}{N}\ (1 + F) \right| \times 100\ [\%]$$

M: Reception margin (%)
N: Ratio of bit rate to clock (N = 32, 64, 372, 256)
D: Duty cycle of clock (D = 0 to 1.0)
L: Frame length (L = 10)
F: Absolute value of clock frequency deviation

Assuming values of F = 0, D = 0.5, and N = 372 in the above formula, the reception margin is determined by the formula below.

$$M = \{0.5 - 1/(2 \times 372)\} \times 100\ [\%] = 49.866\%$$

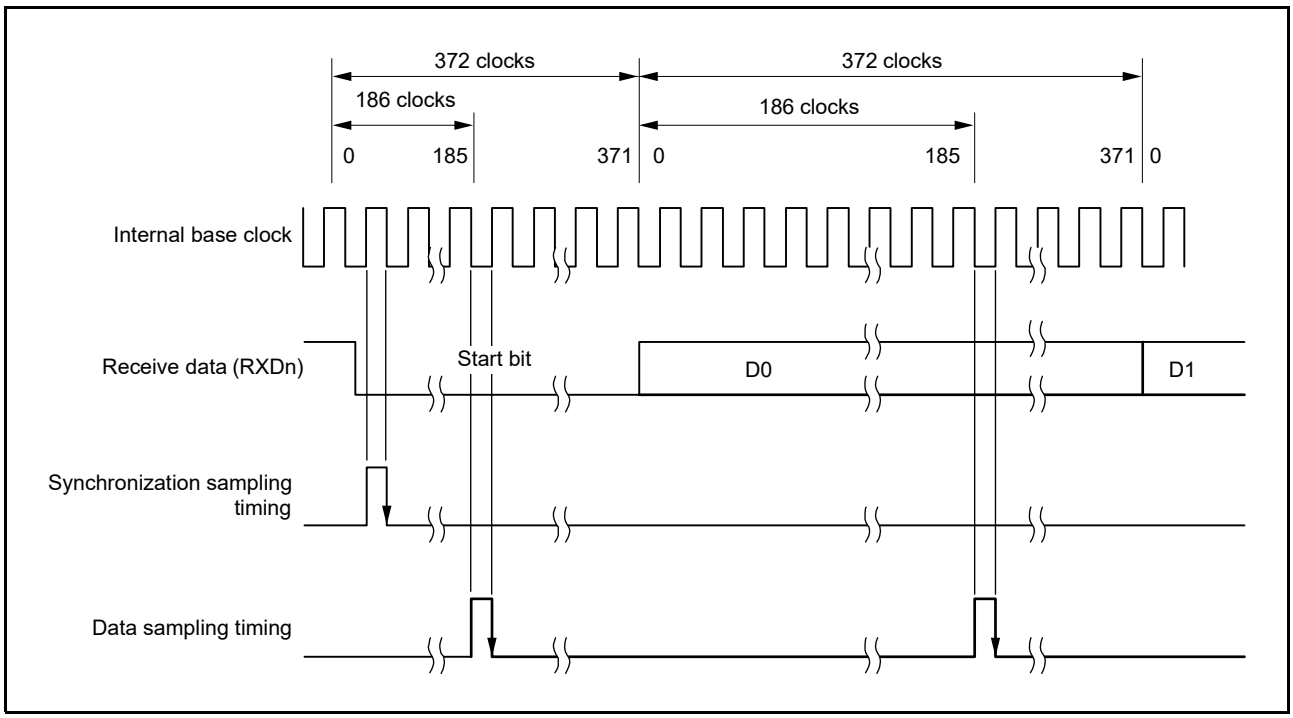


Figure 18.33 Receive Data Sampling Timing in Smart Card Interface Mode (When Clock Frequency is 372 Times the Bit Rate)

18.6.5 SCI Initialization (Smart Card Interface Mode)

Initialize the SCI following the example of flowchart shown in Figure 18.34.
Be sure to initialize the SCI before switching from transmission mode to reception mode and vice versa. Even if the RE bit is set to 0, the RDR register is not initialized.

To change reception mode to transmission mode, first check that reception has completed, and then initialize the SCI. At the end of initialization, set TE = 1 and RE = 0. Reception completion can be verified by reading the RXI request, ORER, or PER flag in SSR.
To change transmission mode to reception mode, first check that transmission has completed, and then initialize the SCI. At the end of initialization, set TE = 0 and RE = 1. Transmission completion can be verified by reading the TEND flag in SSR.

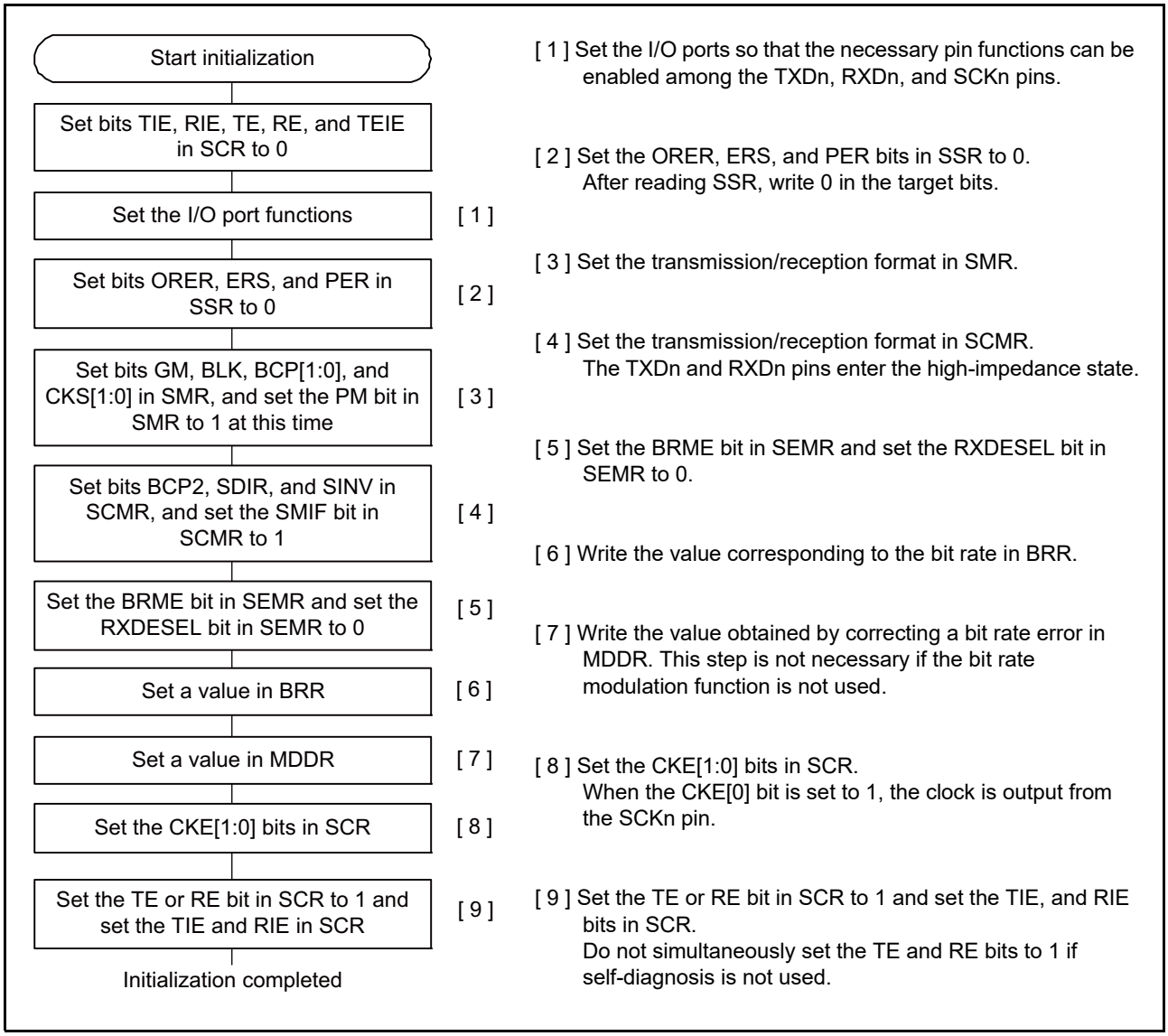


Figure 18.34 Example of SCI Initialization Flowchart (Smart Card Interface Mode)

18.6.6 Serial Data Transmission (Except in Block Transfer Mode)

Serial data transmission in smart card interface mode (except in block transfer mode), in that an error signal is sampled and data can be retransmitted, is different from that in non-smart card interface mode. Figure 18.35 shows the data retransfer operation during transmission.

1. When an error signal from the receiver end is sampled after one-frame data has been transmitted, the ERS flag in SSR is set to 1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Clear the ERS flag to 0 before the next parity bit is sampled.
2. For a frame in which an error signal is received, the TEND flag in SSR is not set. Data is retransferred from TDR to TSR allowing automatic data retransmission.
3. If no error signal is returned from the receiver, the ERS flag is not set to 1.
4. In this case, the SCI judges that transmission of one-frame data (including retransfer) has been completed, and the TEND flag is set. If the TIE bit in SCR is 1 at this time, a TXI interrupt request is generated. Writing transmit data to TDR starts transmission of the next data.

Figure 18.37 shows a sample flowchart of serial transmission. All the processing steps are automatically performed using a TXI interrupt request to activate the DMAC.
When the TEND flag in SSR is set to 1 in transmission, if the TIE bit in SCR is 1, a TXI interrupt request is generated. The DMAC is activated by a TXI interrupt request if the TXI interrupt request is specified as a source of DMAC activation beforehand, allowing transfer of transmit data. The TEND flag is automatically set to 0 when the DMAC transfers the data.
If an error occurs, the SCI automatically retransmits the same data. During this retransmission, the TEND flag is kept to 0 and the DMAC is not activated. Therefore, the SCI and DMAC automatically transmit the specified number of bytes, including retransmission in the case of error occurrence. However, since the ERS flag is not automatically cleared, set the RIE bit to 1 beforehand to enable an ERI interrupt request to be generated at error occurrence, and clear the ERS flag to 0.
When transmitting/receiving data using the DMAC, be sure to make settings to enable the DMAC before making SCI settings.
For DMAC settings, refer to section 9, Direct Memory Access Controller.

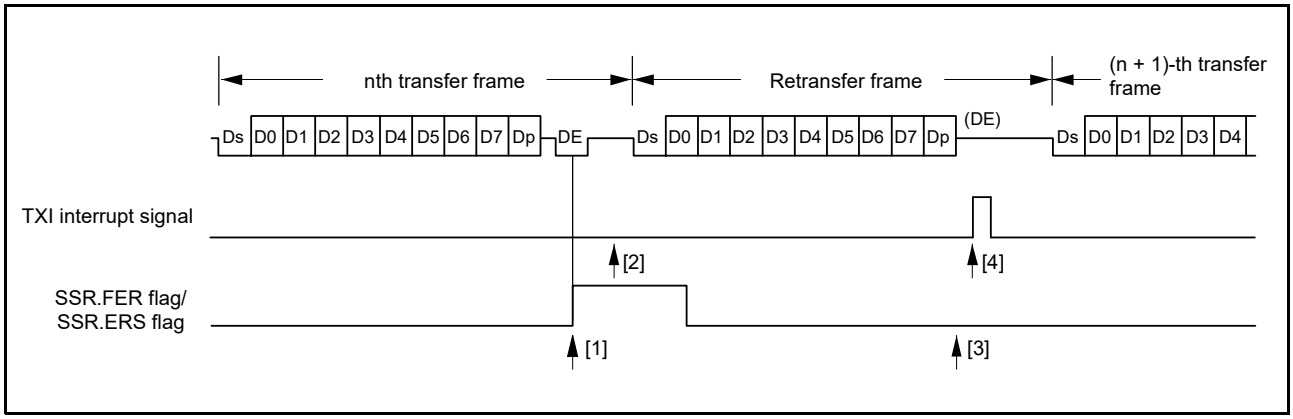


Figure 18.35 Data Retransfer Operation in SCI Transmission Mode

Note that the SSR.TEND flag is set in different timings depending on the GM bit setting in SMR. Figure 18.36 shows the TEND flag generation timing.

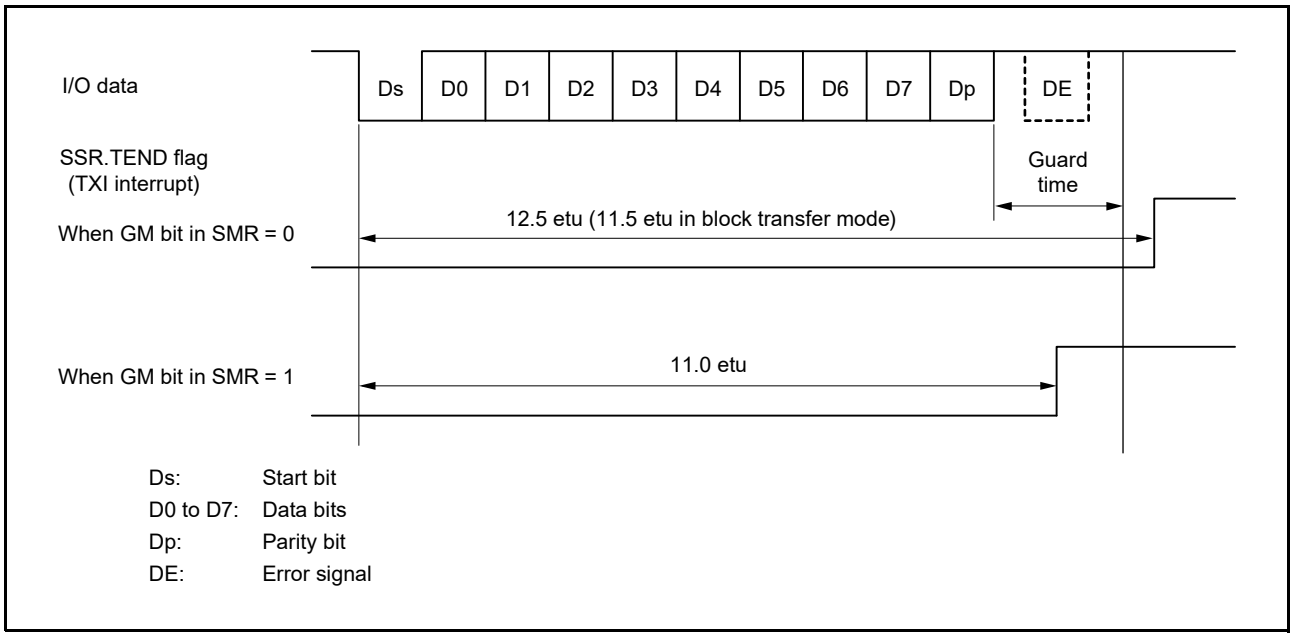


Figure 18.36 SSR.TEND Flag Generation Timing during Transmission

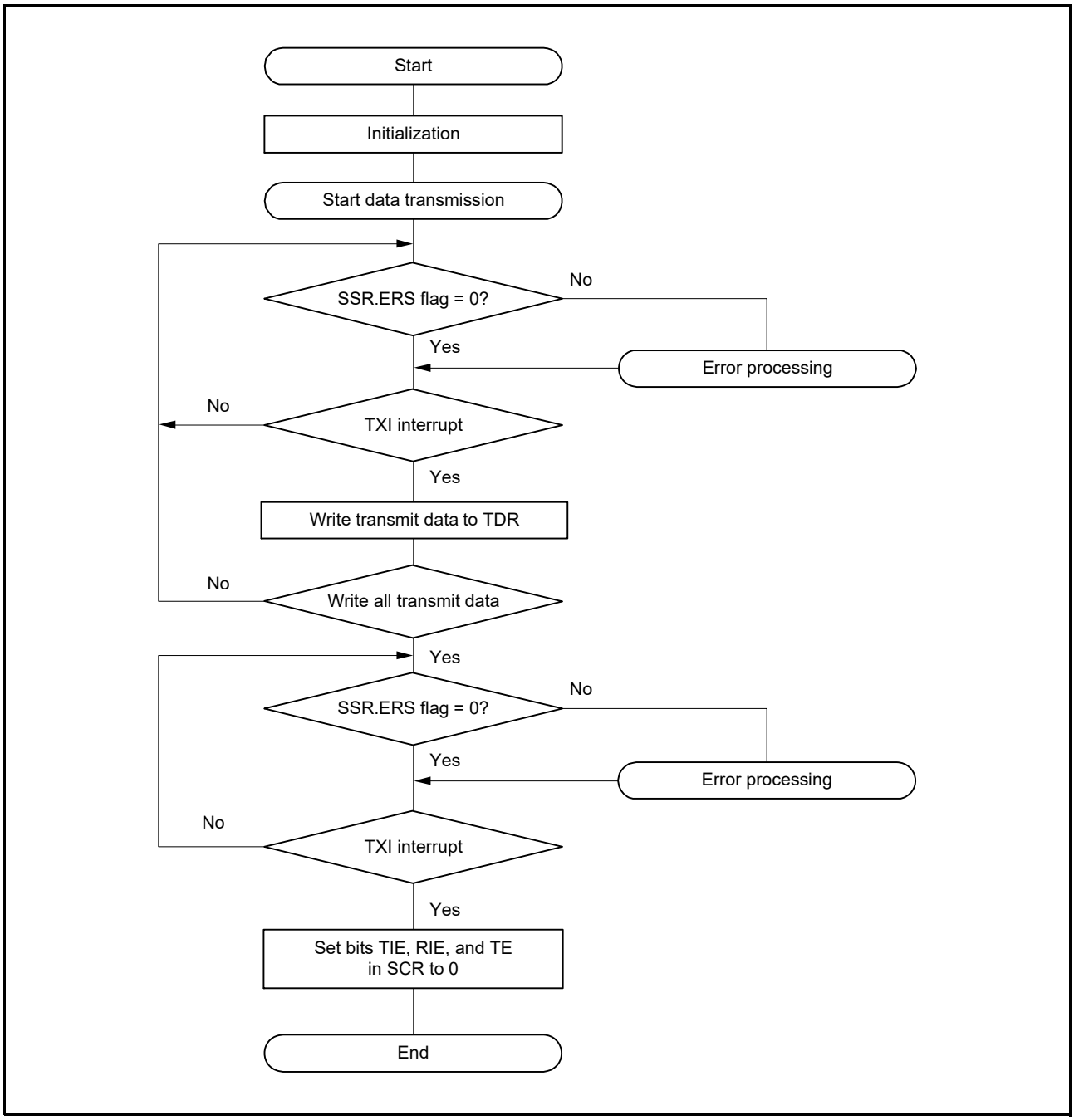


Figure 18.37 Sample Smart Card Interface Transmission Flowchart

18.6.7 Serial Data Reception (Except in Block Transfer Mode)

Serial data reception in smart card interface mode is similar to that in non-smart card interface mode. Figure 18.38 shows the data retransfer operation in reception mode.

1. If a parity error is detected in receive data, the PER flag in SSR is set to 1. When the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Clear the PER flag to 0 before the next parity bit is sampled.
2. For a frame in which a parity error is detected, no RXI interrupt is generated.
3. When no parity error is detected, the PER flag in SSR is not set to 1.
4. In this case, data is determined to have been received successfully. When the RIE bit in SCR is 1, an RXI interrupt request is generated.

Figure 18.39 shows a sample flowchart for serial data reception. All the processing steps are automatically performed using an RXI interrupt request to activate the DMAC.
In reception, setting the RIE bit to 1 allows an RXI interrupt request to be generated. The DMAC is activated by an RXI interrupt request if the RXI interrupt request is specified as a source of DMAC activation beforehand, allowing transfer of receive data.
If an error occurs during reception and either the ORER or PER flag in SSR is set to 1, a receive error interrupt (ERI) request is generated. Clear the error flag after the error occurrence. If an error occurs, the DMAC is not activated and receive data is skipped. Therefore, the number of bytes of receive data specified in the DMAC is transferred.
Even if a parity error occurs and the PER flag is set to 1 during reception, receive data is transferred to RDR, thus allowing the data to be read.
When a reception is forcibly terminated by setting the RE bit in SCR to 0 during operation, read the RDR register because the received data which has not yet been read may be left in RDR.

Note 1. For operations in block transfer mode, refer to section 18.3, Operation in Asynchronous Mode.

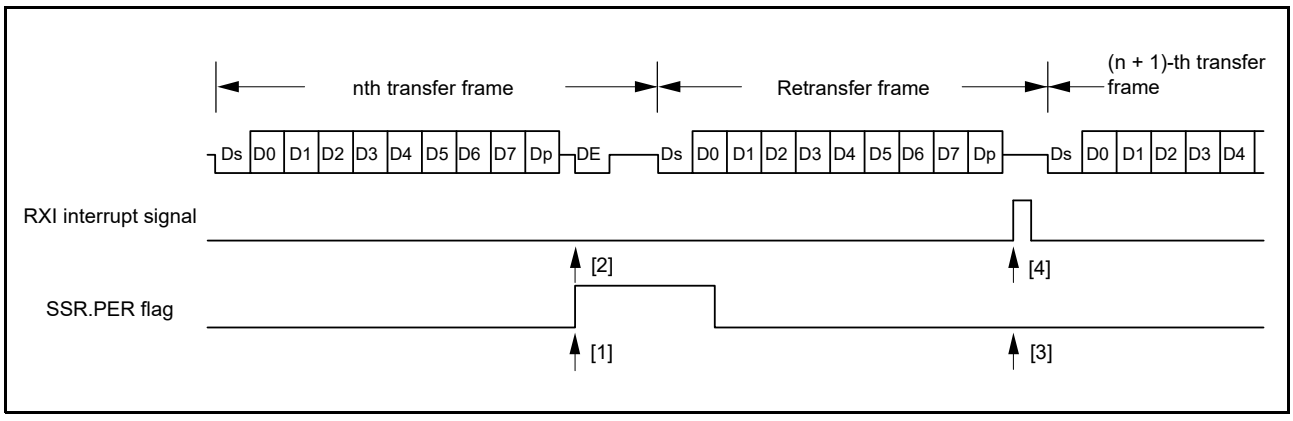


Figure 18.38 Data Retransfer Operation in SCI Reception Mode (Data Retransfer Operation during Reception)

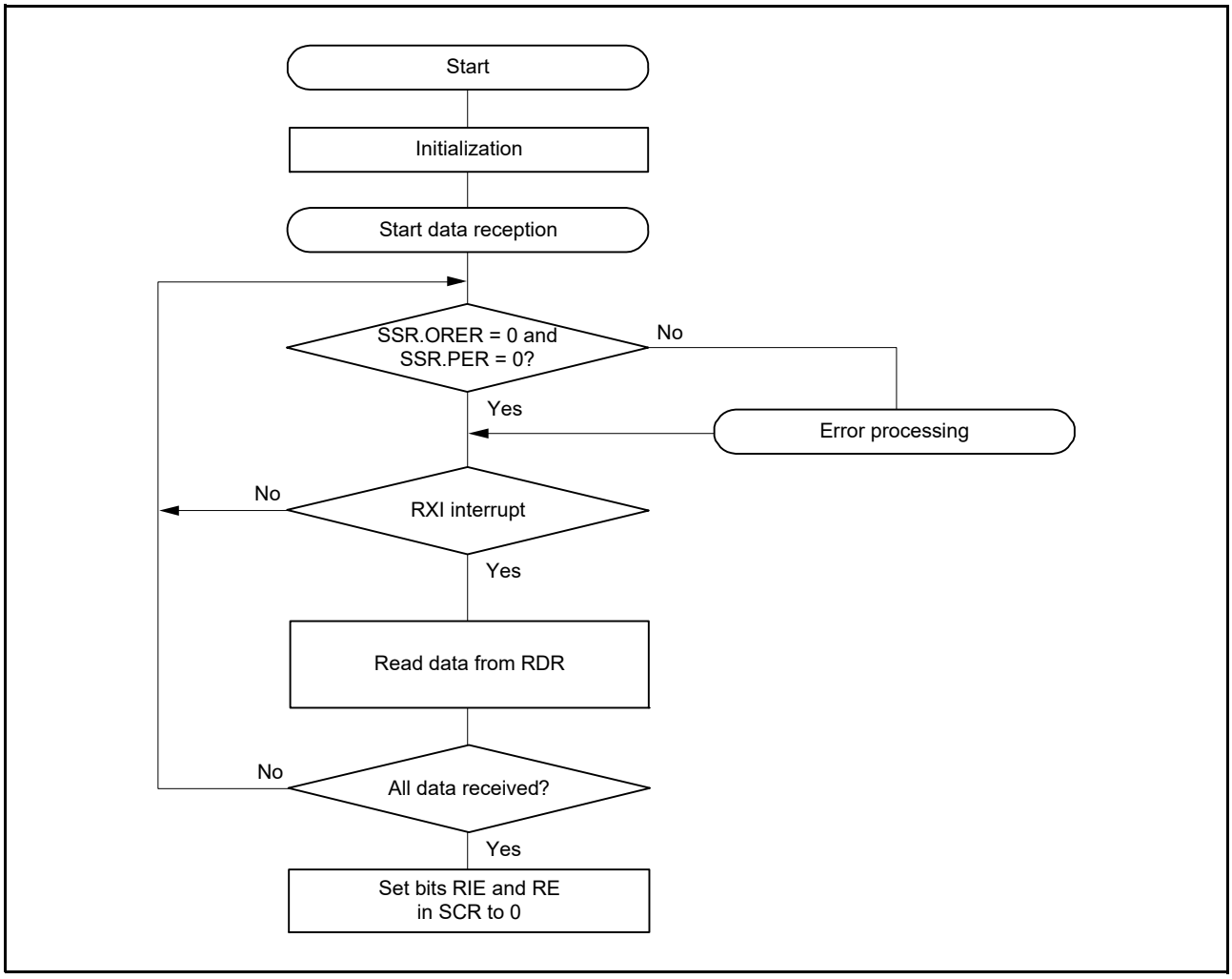


Figure 18.39 Sample Smart Card Interface Reception Flowchart

18.6.8 Clock Output Control

Clock output can be fixed using the CKE[1:0] bits in SCR when the GM bit in SMR is 1. Specifically, the minimum width of a clock pulse can be specified.

Figure 18.40 shows an example of clock output fixing timing when the CKE[0] bit is controlled with GM = 1 and CKE[1] = 0.

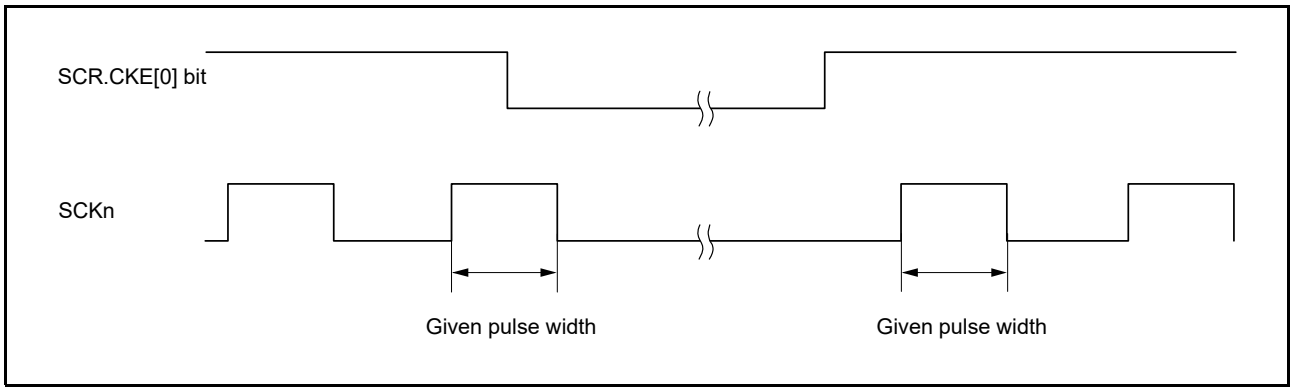


Figure 18.40 Clock Output Fixing Timing

At power-on, use the following procedure to secure the appropriate clock duty cycle.

(1) At Power-On

To secure the appropriate clock duty cycle simultaneously with power-on, use the following procedure.

1. Initially, port input is enabled in the high-impedance state. To fix the potential level, use a pull-up or pull-down resistor.
2. Fix the SCKn pin to the specified output by setting the SCR.CKE[1] bit and I/O port functions.
3. Set SMR and SCMR to enable smart card interface mode.
4. Set the SCR.CKE[0] bit to 1 to start clock output.

18.7 Noise Cancellation Function

Figure 18.41 shows the configuration of the noise filter used for noise cancellation. The noise filter consists of two stages of flip-flop circuits and a match-detection circuit. When the level on the pin matches in three consecutive samples taken at the set sampling interval, the matching level continues to be conveyed internally until the level on the pin again matches in three consecutive samples.

In asynchronous mode, the noise cancellation function can be applied on the RXDn input signal. The period of the base clock (1/16th of a bit-period when SEMR.ABCS = 0 and 1/8th of a bit-period when SEMR.ABCS = 1) is the sampling interval.

If the base clock is stopped with the noise filter enabled and then the clock input is started again, the noise filter operation resumes from where the clock was stopped. If SCR.TE and SCR.RE are set to 0 during base clock input, all of the noise filter flip-flop values are initialized to 1. Accordingly, if the input data is 1 when reception operation resumes, it is determined that a level match is detected and is conveyed to the internal signal. When the level being input corresponds to 0, the initial output of the noise filter is retained until the level matches in three consecutive samples.

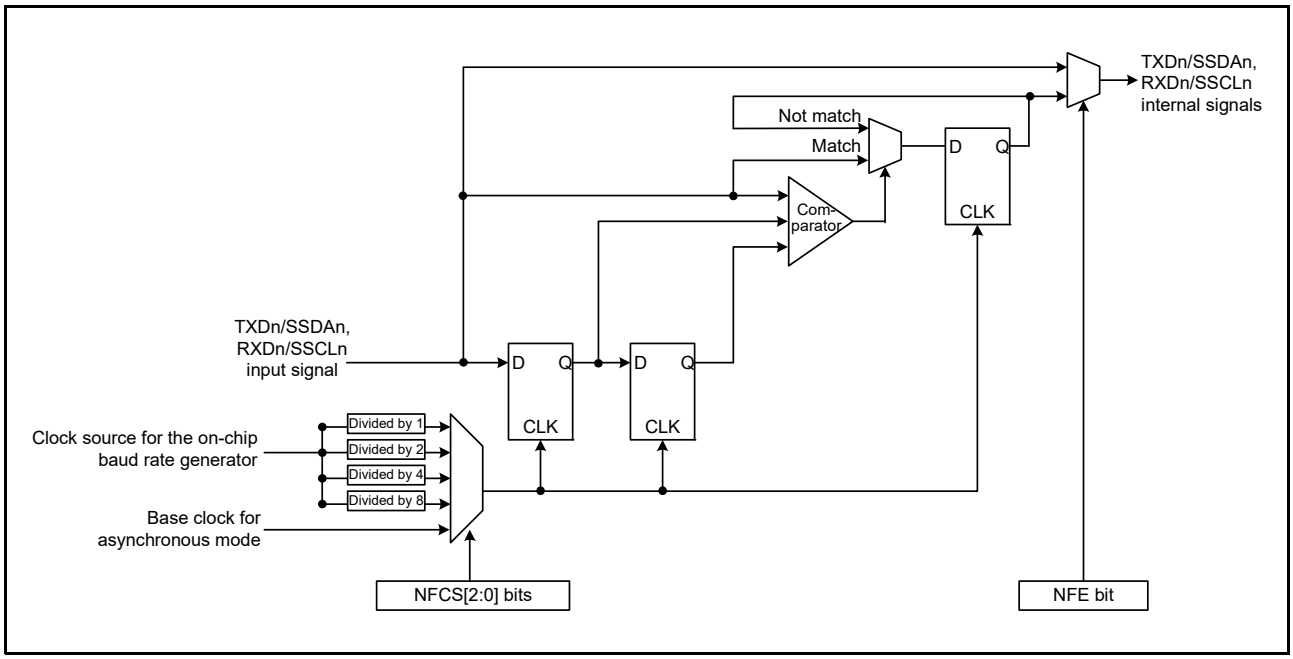


Figure 18.41 Block Diagram of Digital Noise Filter Circuit

18.8 Interrupt Sources

18.8.1 Buffer Operations for TXI and RXI Interrupts

If the conditions for a TXI and RXI interrupt are satisfied while the interrupt status flag in the interrupt controller is 1, the interrupt controller does not output the interrupt request but retains it internally (with a capacity for retention of one request per source).
When the value of the interrupt status flag in the interrupt controller becomes 0, the interrupt request retained within the interrupt controller is output. The internally retained interrupt request is automatically discarded once the actual interrupt is output. Clearing of the corresponding interrupt enable bit (the TIE or RIE bit in the SCR) can also be used to discard an internally retained interrupt request.

18.8.2 Interrupts in Asynchronous Mode and Clock Synchronous Mode

Table 18.20 lists interrupt sources in asynchronous mode and clock synchronous mode. A different interrupt vector is assigned to each interrupt source, and individual interrupt sources can be enabled or disabled with the enable bits in SCR.
If the SCR.TIE bit is 1, a TXI interrupt request is generated when transmit data is transferred from the TDR or TDRL register*1 to the TSR. A TXI interrupt request can also be generated by setting the SCR.TE bit to 1 after setting the SCR.TIE bit to 1 or by using a single instruction to set the SCR.TE and SCR.TIE bit to 1 at the same time. A TXI interrupt request can activate the DMAC to handle data transfer.
A TXI interrupt request is not generated by setting the SCR.TE bit to 1 while the setting of the SCR.TIE bit is 0 or by setting the SCR.TIE bit to 1 while the setting of the SCR.TE bit is 1.*1
When new data is not written by the time of transmission of the last bit of the current transmit data and the setting of the SCR.TEIE bit is 1, the SSR.TEND flag becomes 1 and a TEI interrupt request is generated. Furthermore, when the setting of the SCR.TE bit is 1, the SSR.TEND flag retains the value 1 until further transmit data are written to the TDR or TDRL register*1, and setting the SCR.TEIE bit to 1 leads to the generation of a TEI interrupt request.
Writing data to the TDR or TDRL register*1 leads to clearing of the SSR.TEND flag and, after a certain time, discarding of the TEI interrupt request.
If the SCR.RIE bit is 1, an RXI interrupt request is generated when received data is stored in the RDR. An RXI interrupt request can activate the DMAC to handle data transfer.
Setting of any from among the ORER, FER, and PER flags in the SSR to 1 while the SCR.RIE bit is 1 leads to the generation of an ERI interrupt request. An RXI interrupt request is not generated at this time. Clearing all three flags (ORER, FER, and PER) leads to discarding of the ERI interrupt request.

Note 1. In the case where asynchronous mode and 9-bit data length are selected
Note 2. To temporarily prohibit TXI interrupts at the time of transmission of the last of the data and so on when you wish a new round of transmission to start after handling of the transmission-completed interrupt, control prohibiting and permitting of the interrupt by using the interrupt request enable bit in the interrupt controller rather than using the SCR.TIE bit. This can prevent the suppression of TXI interrupt requests in the transfer of new data.

Table 18.20 Interrupt Sources

Name	Interrupt Source	Interrupt Flag	DMAC Activation	Priority
ERI	Receive error	ORER, FER, or PER	Not possible	High
RXI	Receive data full	—	Possible	↑
TXI	Transmit data empty	—	Possible	
TEI	Transmit end	TEND	Not possible	Low

18.8.3 Interrupts in Smart Card Interface Mode

Table 18.21 lists interrupt sources in smart card interface mode. A transmit end interrupt (TEI) request cannot be used in this mode.

Table 18.21 SCI Interrupt Sources

Name	Interrupt Source	Interrupt Flag	DMAC Activation	Priority
ERI	Receive error or error signal detection	ORER, PER, or ERS	Not possible	High
RXI	Receive data full	—	Possible	↑
TXI	Transmit data empty	TEND	Possible	Low

Data transmission/reception using the DMAC is also possible in smart card interface mode, similar to in the normal SCI mode. In transmission, when the TEND flag in SSR is set to 1, a TXI interrupt request is generated. This TXI interrupt request activates the DMAC allowing transfer of transmit data if the TXI request is specified beforehand as a source of DMAC activation. The TEND flag is automatically set to 0 when the DMAC transfers the data.
If an error occurs, the SCI automatically retransmits the same data. During the retransmission, the TEND flag is kept to 0 and the DMAC is not activated. Therefore, the SCI and DMAC automatically transmit the specified number of bytes, including retransmission in the case of error occurrence. However, the ERS flag in SSR is not automatically cleared to 0 at error occurrence. Therefore, the ERS flag must be cleared by previously setting the RIE bit in SCR to 1 to enable an ERI interrupt request to be generated at error occurrence.
When transmitting/receiving data using the DMAC, be sure to make settings to enable the DMAC before making SCI settings. For DMAC settings, refer to section 9, Direct Memory Access Controller .
In reception, an RXI interrupt request is generated when receive data is set to RDR. This RXI interrupt request activates the DMAC allowing transfer of receive data if the RXI request is specified beforehand as a source of DMAC activation. If an error occurs, the error flag is set. Therefore, the DMAC is not activated and an ERI interrupt request is issued to the CPU instead; the error flag must be cleared.

18.9 Usage Notes

18.9.1 Setting the Module Stop Function

Standby control register 4 (STBCR4) is used to stop and start SCI operations. With the value after a reset, SCI operations are stopped. Register access is enabled by releasing the module stop state. For details, refer to section 52, Power-Down Modes.

18.9.2 Break Detection and Processing

When a framing error is detected, a break can be detected by reading the RXDn pin value directly. In a break, the input from the RXDn pin becomes all 0s, and so the FER flag in SSR is set to 1 (framing error has occurred), and the PER flag in SSR may also be set to 1 (parity error has occurred). The SCI continues the receive operation even after a break is received. Therefore, note that even if the FER flag is set to 0 (no framing error occurred), it will be set to 1 again. When the SEMR.RXDESEL bit is 1, the SCI sets the SSR.FER flag to 1 and stops receiving operation until a start bit of the next data frame is detected. If the SSR.FER flag is set to 0 at this time, the SSR.FER flag retains 0 during the break. When the RXDn pin is set to 1 and the break ends, detecting the beginning of the start bit at the first falling edge of the RXDn pin allows the SCI to start the receiving operation.

18.9.3 Mark State and Production of Breaks

When the SCR.TE bit is 0 (serial transmission is disabled), setting the I/O port function makes selection of the level and direction (input or output) of the TXDn pin possible. If this is done, the TXDn pin can be placed in the mark state to send a break at the time of data transmission. Until the SCR.TE bit is set to 1 (serial transmission is enabled), the I/O port function is used to set the TXDn pin to output 1 and set the pin mode to a general I/O port pin, and thus place the communication line in the mark state (state of having the value 1). On the other hand, to output a break at the time of data transmission, set the TXDn pin to output 0 and make the pin mode settings for a general I/O port pin. When the SCR.TE bit is set to 0, the transmitter is initialized regardless of the current state of transmission.

18.9.4 Receive Error Flags and Transmit Operations (Clock Synchronous Mode)

Transmission cannot be started when a receive error flag (ORER) in SSR is set to 1, even if data is written to TDR. Be sure to set the receive error flags to 0 before starting transmission. Note also that the receive error flags cannot be set to 0 even if the RE bit in SCR is set to 0 (serial reception is disabled).

18.9.5 Writing Data to TDR

Data can be written to TDR, TDRH, and TDRL. However, if new data is written to TDR, TDRH, and TDRL when transmit data is remaining in TDR, TDRH, and TDRL, the previous data in TDR, TDRH, and TDRL is lost because it has not been transferred to TSR yet. Be sure to write transmit data to TDR, TDRH, and TDRL in the TXI interrupt request handling routine.

18.9.6 Restrictions on Clock Synchronous Transmission (Clock Synchronous Mode)

When the external clock source is used as a synchronization clock, the following restrictions apply.

(1) Start of transmission

Update TDR by the CPU or DMAC and wait for at least five P1φ cycles before allowing the transmit clock to be input (see Figure 18.42).

(2) Continuous transmission

(a) Write the next transmit data to TDR or TDRL before the falling edge of the transmit clock (bit 7) (see Figure 18.42).

(b) When updating TDR after bit 7 has started to transmit, update TDR while the synchronization clock is in the low-level period, and set the high-level width of the transmit clock (bit 7) to four P1φ cycles or longer (see Figure 18.42).

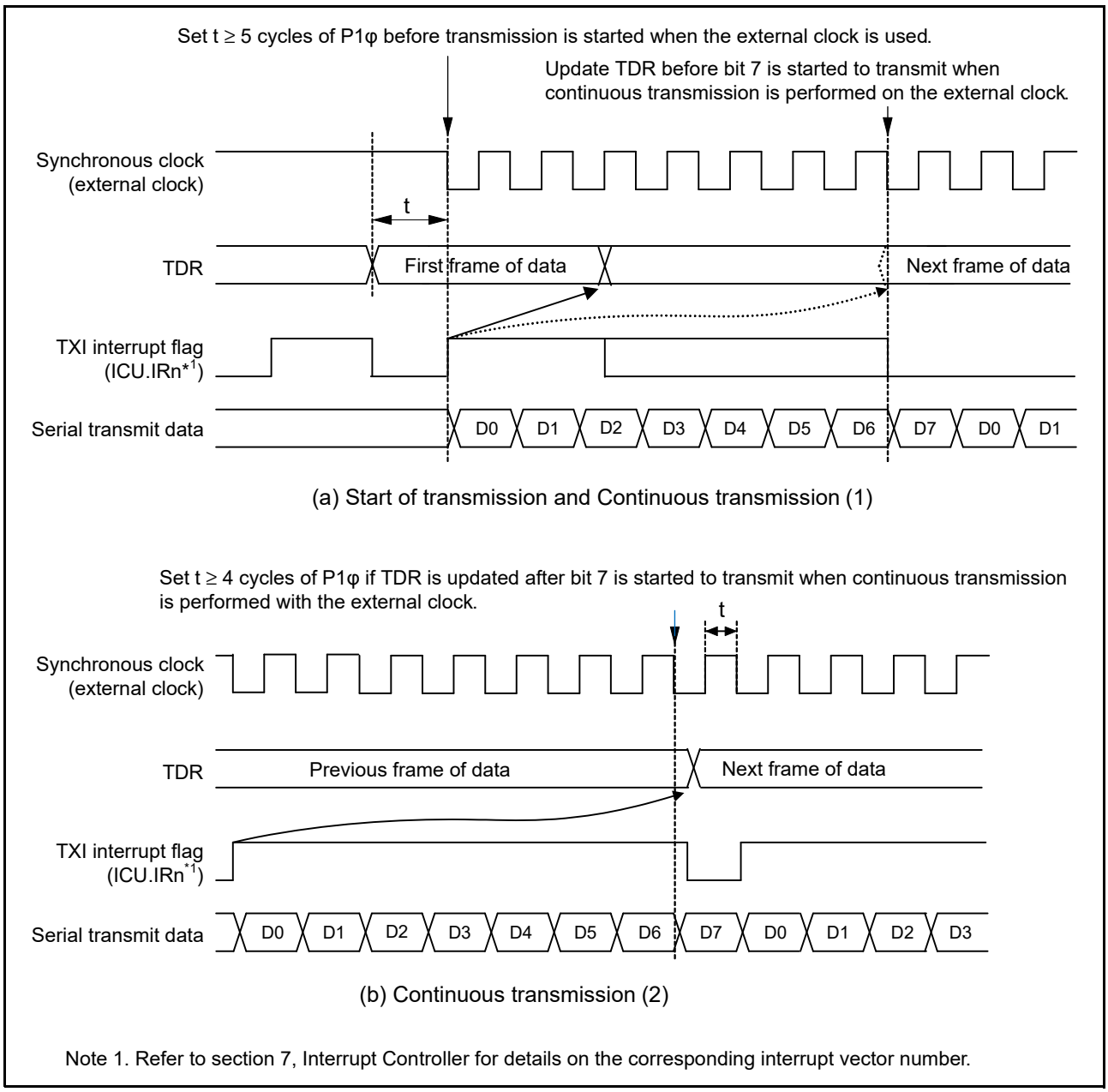


Figure 18.42 Restrictions on Use of External Clock in Clock Synchronous Transmission

18.9.7 Restrictions on Using DMAC

When using the DMAC or DTC to read RDR, RDRH, and RDRL, be sure to set the receive data full interrupt (RXI) as the activation source of the relevant SCI.

18.9.8 Notes on Starting Transfer

At the point where transfer starts when the interrupt status flag (IRn.IR bit) in the interrupt controller is 1, follow the procedure below to clear interrupt requests before permitting operations (by setting the SCR.TE or SCR.RE bit to 1). For details on the interrupt status flag, refer to section 7, Interrupt Controller.

- Confirm that transfer has stopped (the setting of the SCR.TE or SCR.RE bits is 0).
- Set the corresponding interrupt enable bit (SCR.TIE or SCR.RIE) to 0.
- Read the corresponding interrupt enable bit (SCR.TIE or SCR.RIE bit) to check that it has become 0.
- Set the interrupt status flag (IRn.IR bit) in the interrupt controller to 0.

18.9.9 SCI Operations during Low Power Consumption State

(1) Transmission

When making settings for the module stopped state or in transitions to software standby, stop operations (by setting the TIE, TE, and TEIE bits in the SCR to 0) after switching the TXDn pin to the general I/O port pin function. Setting the TE bit to 0 resets the TSR register and the TEND bit in the SSR. Depending on the port settings, output pins may output the level before a transition to the low power consumption state is made after release from the module stopped state or software standby mode. When transitions to these states are made during transmission, the data being transmitted become indeterminate.

To transmit data in the same transmission mode after cancellation of the low power consumption state, set the TE bit to 1, read SSR, and write data to TDR sequentially to start data transmission. To transmit data with a different transmission mode, initialize the SCI first.

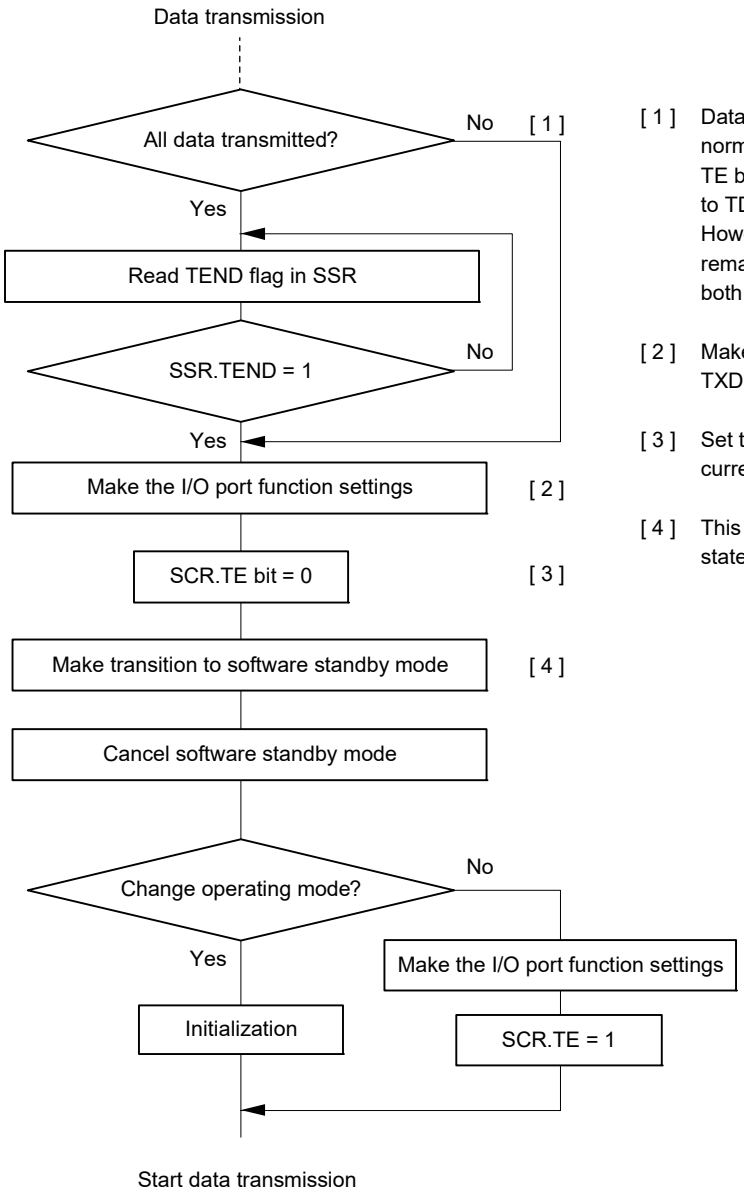
Figure 18.43 shows a sample flowchart for transition to software standby mode during transmission. Figure 18.44 and Figure 18.45 show the port pin states during transition to software standby mode.

(2) Reception

Before specifying the module stop state or making a transition to software standby mode, stop the receive operations (RE = 0 in SCR). If transition is made during data reception, the data being received will be invalid.

To receive data in the same reception mode after cancellation of the low power consumption state, set the RE bit to 1, and then start reception. To receive data in a different reception mode, initialize the SCI first.

Figure 18.46 shows a sample flowchart for transition to software standby mode during reception.



[1] Data being transmitted is lost halfway. Data can be normally transmitted from the CPU by setting the TE bit in SCR to 1, reading SSR, and writing data to TDR after canceling software standby mode. However, if the DMAC has been activated, the data remaining in the DMAC will be transmitted when both the TE and TIE bits in SCR are set to 1.

[2] Make the I/O port function settings to switch the TXDn pin to operate as a general I/O port.

[3] Set the TIE and TEIE bits in the SCR if they are currently set to 1.

[4] This includes the setting for the module stopped state.

Figure 18.43 Example of Flowchart for Transition to Software Standby Mode during Transmission

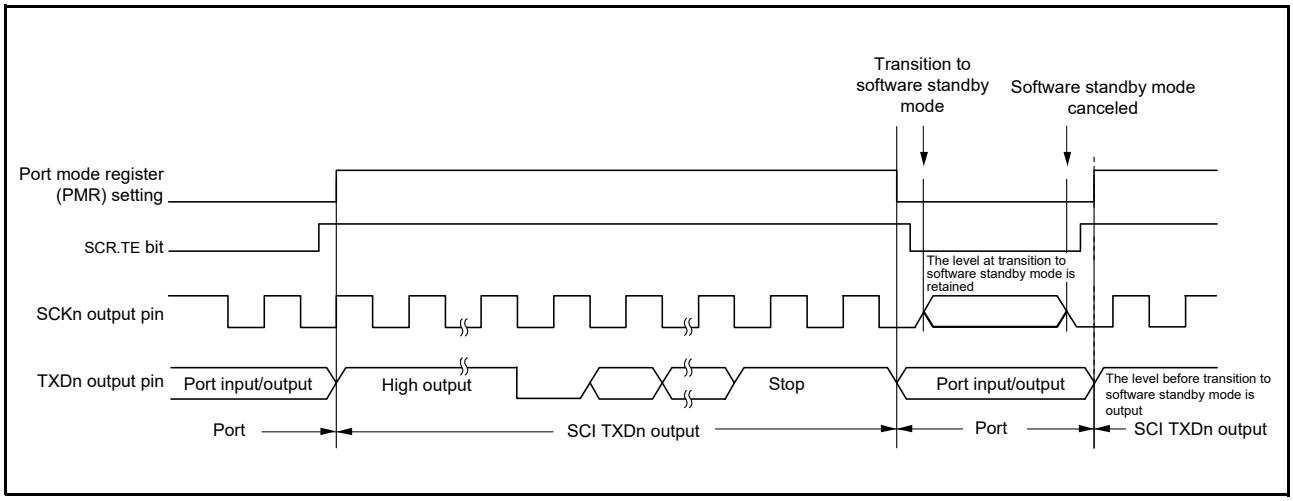


Figure 18.44 Port Pin States during Transition to Software Standby Mode (Internal Clock, Asynchronous Transmission)

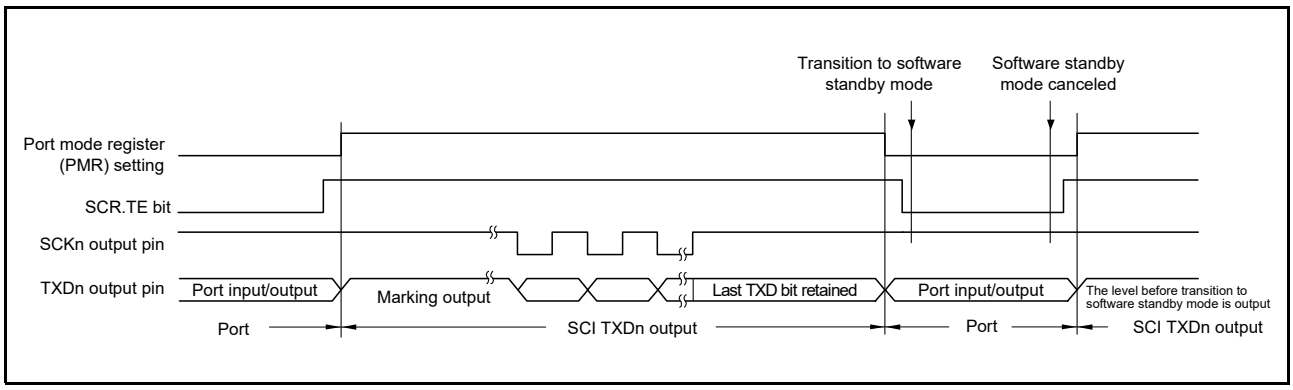


Figure 18.45 Port Pin States during Transition to Software Standby Mode (Internal Clock, Clock Synchronous Transmission)

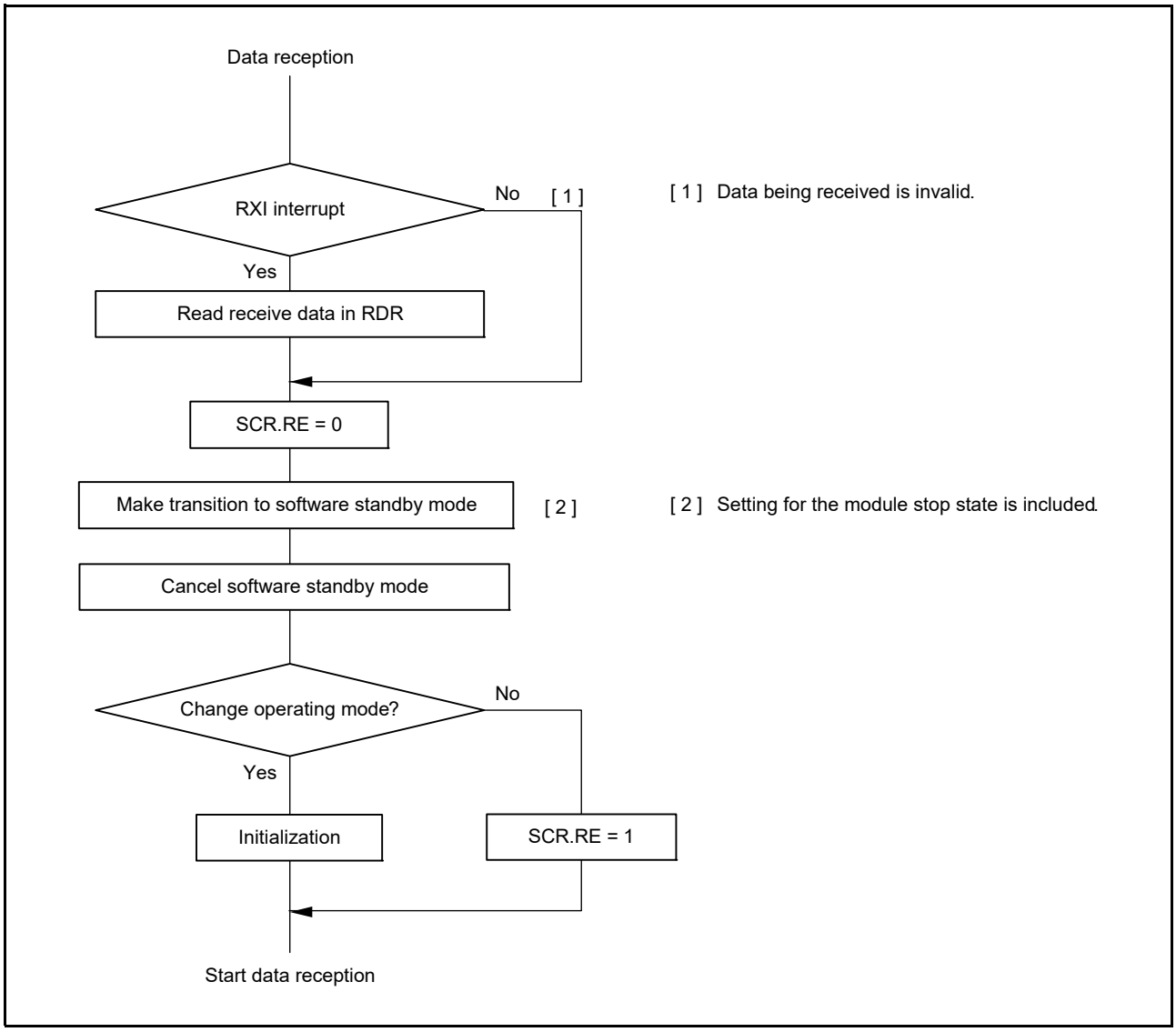


Figure 18.46 Example of Flowchart for Transition to Software Standby Mode during Reception

18.9.10 External Clock Input in Clock Synchronous Mode

In clock synchronous mode, the external clock SCKn must be input as follows:
High-pulse period, low-pulse period = 2 P1φ cycles or more, period = 6 P1φ cycles or more

18.9.11 Note on Transmit Enable Bit (TE bit)

When setting the SCR.TE bit to 0 (serial transmission is disabled) while the pin function is “TXDn”, output of the pin becomes high impedance.
Prevent the TXDn line from becoming high impedance by any of the following ways:

(1) Connect a pull-up resistor to the TXDn line.
(2) Change the pin function to “general-purpose I/O port, output” before setting the SCR.TE bit to 0.
Set the SCR.TE bit to 1 before changing the pin function to “TXDn”.

18.10 IrDA Communications

The channel 0 serial communications interface (SCI) is capable of working with the Infrared Data Association (IrDA) module to handle transfer through IrDA communications in conformance with version 1.0 of the IrDA protocol.
Using the IRE bit in IRCR to enable the IrDA function leads to encoding and decoding of the SCI_TXD0 and SCI_RXD0 signals of the channel 0 serial communications interface, respectively, to convert them to and from waveforms in conformance with version 1.0 of the IrDA protocol. Infrared transfer in conformance with version 1.0 of the IrDA protocol can be handled by connecting a transceiver for use in infrared communications.
In version 1.0 of the IrDA protocol, IrDA transfer starts at a transfer rate of 9600 bps. The transfer rate is changeable during transfer as required. The IrDA module itself does not have a function for automatically changing the transfer rate. Accordingly, change the transfer rate by changing that for the serial communications interface.

A block diagram and pin configuration for IrDA transfer are shown in Figure 18.47and Table 18.22, respectively.

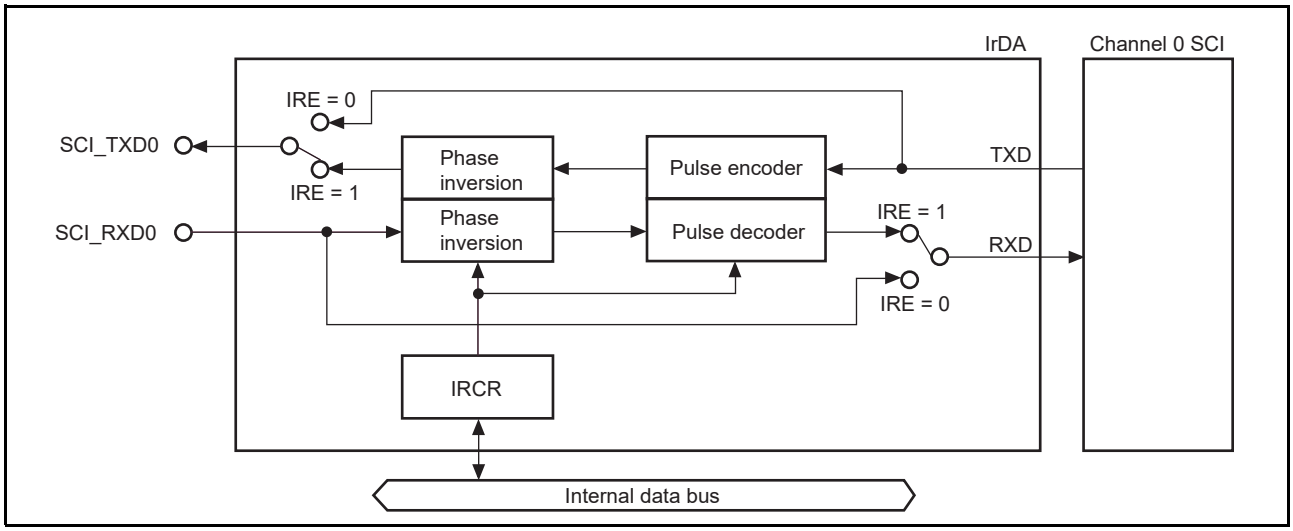


Figure 18.47 Block Diagram

Table 18.22 Pin Configuration

Pin Name	Symbol	I/O	Function
IrDA data transmission pin	SCI_TXD0	Output	IrDA data transmission output
IrDA data reception pin	SCI_RXD0	Input	IrDA data reception input

18.11 Description of the IrDA Register

Table 18.23 shows the register configuration.

Table 18.23 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
IrDA Control Register	IRCR	R/W	H'00	H'E8014000	8

18.11.1 IrDA Control Register (IRCR)

IRCR sets the operation of the IrDA module.

	b7	b6	b5	b4	b3	b2	b1	b0
	IRE	IRCKS[2:0]			IRTX INV	IRRX INV	—	—
Value after reset::	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2	IRRXINV	SCI_RXD0 Data Level Switching	Sets inversion of the logic level of inputs on SCI_RXD0. 0: The input on SCI_RXD0 is used without change as the received data. 1: The input on SCI_RXD0 is inverted before it is treated as the received data.	R/W
b3	IRTXINV	SCI_TXD0 Data Level Switching	Sets inversion of the logic level of outputs on SCI_TXD0. 0: Data for transmission is output without change to SCI_TXD0. The pulse-width setting in the IRCKS bits determines the period over which the pulses are at the high level. 1: Data for transmission is inverted before being output to SCI_TXD0. The pulse-width setting in the IRCKS bits determines the period over which the pulses are at the low level.	R/W
b6 to b4	IRCKS[2:0]	IrDA Clock Select	Selects the pulse width for use in encoding output pulses from SCI_TXD0 when the setting of the IRE bit is 1. 000: Bit rate × 3/16 001: P1φ/2 010: P1φ/4 011: P1φ/8 100: P1φ/16 101: P1φ/32 110: P1φ/64 111: P1φ/128	R/W
b7	IRE	IrDA Enable	Sets the function for use with the SCI_TXD0 and SCI_RXD0 pins as normal serial or IrDA. 0: Disables the IrDA function. Data from the TXD pin of the channel 0 serial communications interface is output without change to SCI_TXD0. Data on SCI_RXD0 is output without change to the RXD pin of the channel 0 serial communications interface. 1: Enables the IrDA function. Data from the TXD pin of the channel 0 serial communications interface is encoded before being output through SCI_TXD0. Data on SCI_RXD0 is decoded before being output to the RXD pin of the channel 0 serial communications interface.	R/W

18.12 IrDA Operation

18.12.1 Flow of Settings for IrDA Operation

Set up IrDA operation by following the procedure listed below.

1. Make the general-purpose I/O port-pin settings.
2. Set the IRCR register.
3. Set the registers related to the serial communications interface.

18.12.2 Transmission

In transmission with the IrDA function enabled, serial data (UART frames of data) from the TXD pin of the serial communications interface are converted to IR frames (see Figure 18.48). If the IRTXINV bit is set to 0, when the value of the serial data is 0, a high-level pulse with the width equivalent to three sixteenths of the bit period is output to the SCI_TXD0 pin by default. Note that the high-level pulse width is selectable by using IRCKS[2:0] bits in the IRCR register. In the IrDA protocol, the specifiable high-level pulse widths are from a minimum of 1.41 µs to a maximum of (3/16 + 2.5%) × bit period or (3/16 × bit period) + 1.08 µs. When the frequency of P1φ is 66 MHz, the specifiable high-level pulse widths are from a minimum of 1.41 µs to a maximum of 1.94 µs. On the other hand, if the value of the serial data is 1, no high-level pulse is output.

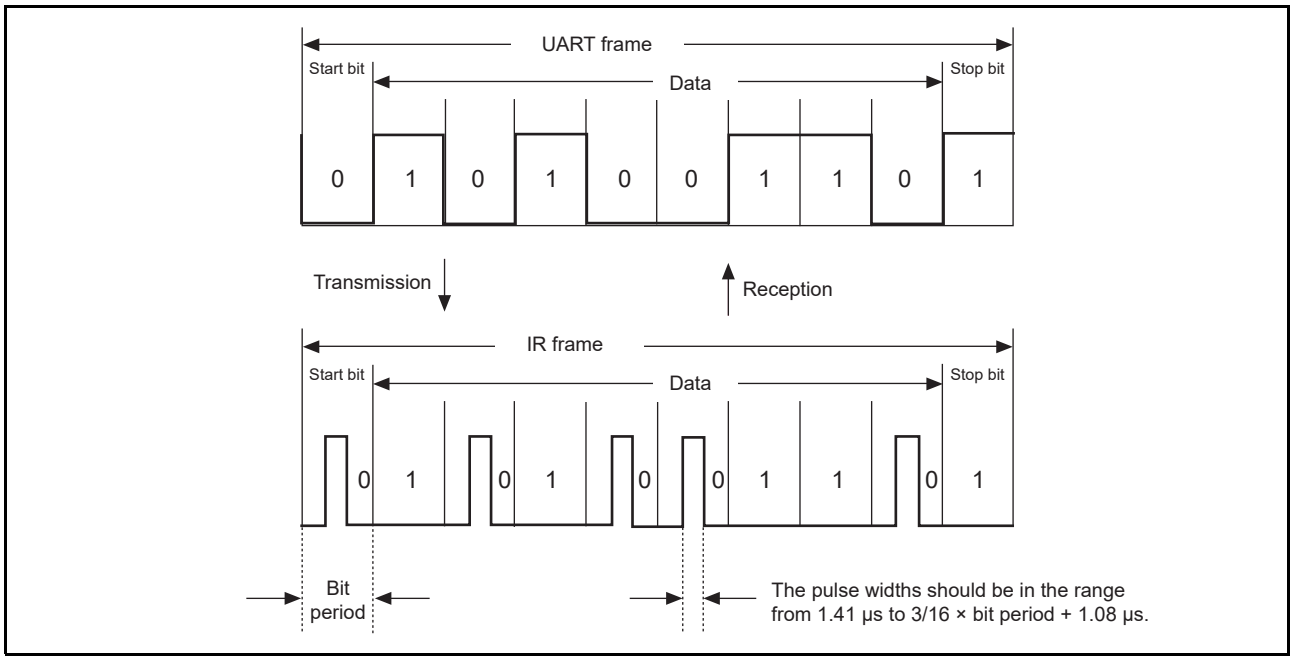


Figure 18.48 Example of the Operation in IrDA Transfer

18.12.3 Reception

In reception with the IrDA function enabled, IR frames of data from the SCI_RXD0 pin are converted to serial data and output to the RXD pin of the serial communications interface. If the setting of the IRRXINV bit is 0, the value 0 is output when a high-level pulse is detected and the value 1 is output when no high-level pulse is detected during each bit period. Note that pulses with a width shorter than the minimum pulse width of 1.41 µs cannot be recognized.

18.12.4 Selecting the High-Level Pulse Width

The correspondences between the specifiable value in the IRCKS[2:0] bits (the minimum pulse width), operating frequency of P1φ, and the bit rate for use in making the pulse width shorter than the bit period × 3/16 in transmission are given in Table 18.24.

Table 18.24 Settings of the IRCKS[2:0] Bits

	Upper Row: Bit Rate (bps); Lower Row: Bit Period × 3/16 (μs)					
	2400	9600	19200	38400	57600	115200
P1φ (MHz)	78.13	19.53	9.77	4.88	3.26	1.63
50	111	111	111	111	111	— *1
60	111	111	111	111	111	— *2
66	111	111	111	111	111	— *2

Note 1. This bit rate is not specifiable for the serial communications interface.
Note 2. No pulse width shorter than the bit period × 3/16 is specifiable.

18.13 Note on IrDA Usage

18.13.1 Minimum Pulse Width in Reception

Pulses with a width shorter than the minimum pulse width of 1.41 μs cannot be recognized.

18.13.2 Base Clock in Asynchronous Mode for the Serial Communications Interface

The IrDA module works with the serial communications interface by receiving the base clock with a frequency 16 times the bit rate for communications from the interface. For the serial communications interface, either 16 or 8 cycles of the base clock are selectable as the bit period. On the other hand, the IrDA module is only capable of operation when the setting for the bit period of the serial communications interface is 16 cycles of the base clock.

19. Renesas Serial Peripheral Interface

This LSI circuit includes three independent Renesas serial peripheral interfaces.
This module is capable of full-duplex synchronous serial communication.

19.1 Features

This module has the following features.

- SPI transfer functions
  Use of MOSI (master out/slave in), MISO (master in/slave out), SSL (slave select), and RSPCK (SPI clock) signals allows for serial communications through SPI operation (four-wire method).
  Capable of serial communications in master/slave mode
  Supports mode fault error detection (only in SPI slave mode)
  Supports overrun error detection (only in SPI slave mode)
  Switching of the polarity of the serial transfer clock
  Switching of the clock phase of serial transfer
- Data format
  MSB-first/LSB-first selectable
  Transfer bit-length is selectable as 8, 16, or 32 bits.
- Bit rate
  RSPCK can be divided by a maximum of 4096 in master mode
  RSPCK can be generated by dividing P1φ by the on-chip baud rate generator.
  An externally input clock can be used as a serial clock.
- Buffer configuration
  8 bytes for transmission and 32 bytes for reception
- SSL control function
  One SSL signal for each channel
  In master mode, outputs SSL signal.
  In slave mode, inputs SSL signal.
  Controllable delay from SSL output assertion to RSPCK operation (RSPCK delay)
  Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
  Controllable delay from RSPCK stoppage to SSL output negation (SSL negation delay)
  Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
  Controllable wait for next-access SSL output assertion (next-access delay)
  Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
  Function for changing SSL polarity
- Control in master transfer
  A transfer of up to four commands can be executed sequentially in looped execution.
  For each command, the following can be set:
  SSL signal value, bit rate, RSPCK polarity/phase, transfer data length, LSB/MSB first, burst, RSPCK delay, SSL negation delay, and next-access delay.
  A transfer can be initiated by writing to the transmit buffer.
  A transfer can be initiated by clearing the SPTEF bit.
  MOSI signal value specifiable in SSL negation
- Interrupt sources
  Maskable interrupt sources:
  Receive interrupt (receive buffer full)
  Transmit interrupt (transmit buffer empty)
  Error interrupt (mode fault, overrun)

- Others
  Provides loop back mode
  Provides a function for disabling (initializing) this module

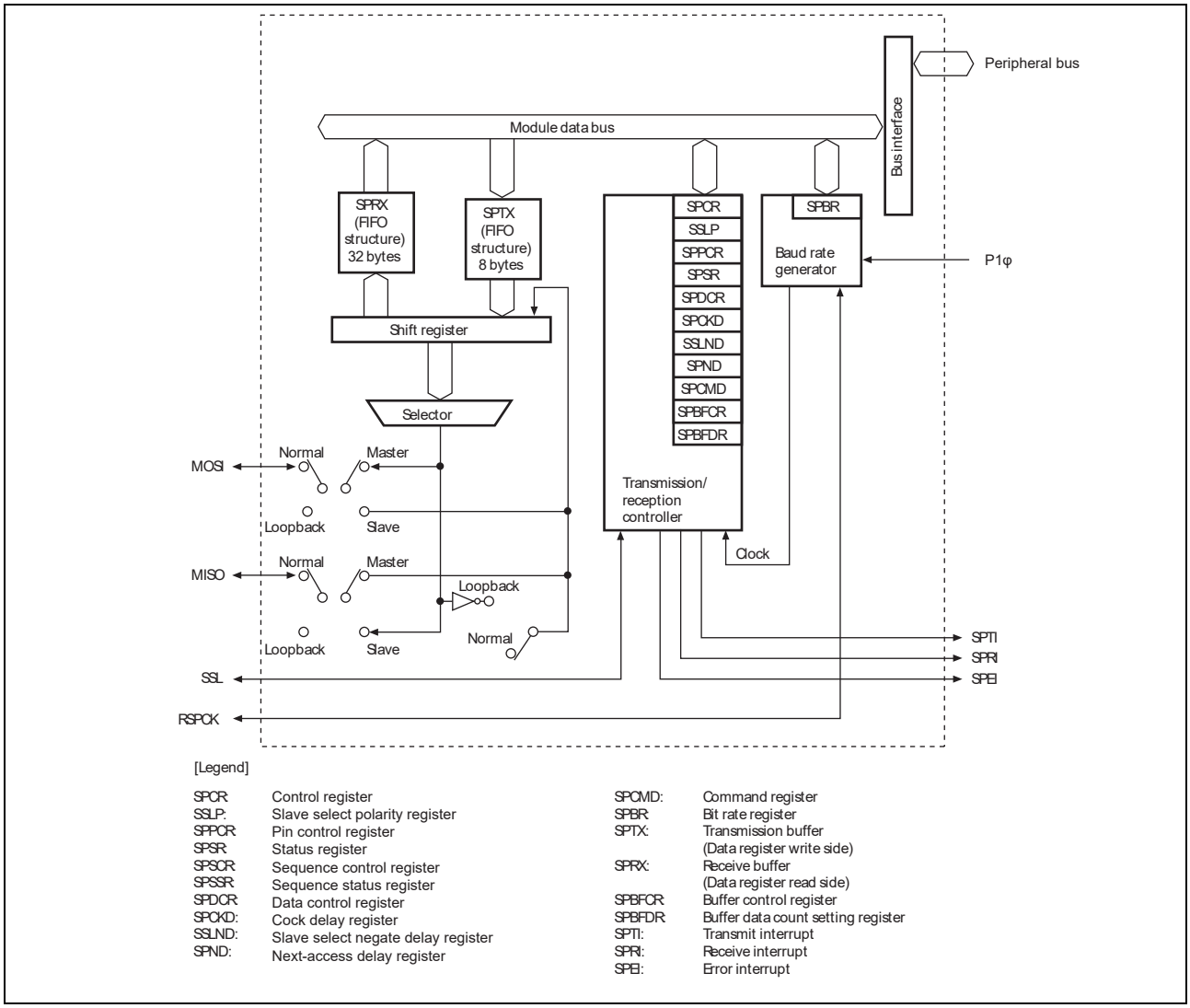


Figure 19.1 Block Diagram (for One Channel)

19.2 Input/Output Pins

Table 19.1 shows the pin configuration. This module automatically switches the input/output direction of the SSL pin. SSL is set as an output in master mode and as an input in slave mode. Pins RSPCK, MOSI, and MISO are automatically set as inputs or outputs according to the setting of master or slave and the level input on SSL (see section 19.4.2, Pin Control).

Table 19.1 Pin Configuration

Channel	Pin Name	Pin Name	I/O	Function
0	Clock pin	RSPCK0	I/O	Clock input/output
	Master transmit data pin	MOSI0	I/O	Master transmit data
	Slave transmit data pin	MISO0	I/O	Slave transmit data
	Slave select 0 pin	SSL00	I/O	Slave selection
1	Clock pin	RSPCK1	I/O	Clock input/output
	Master transmit data pin	MOSI1	I/O	Master transmit data
	Slave transmit data pin	MISO1	I/O	Slave transmit data
	Slave select 0 pin	SSL10	I/O	Slave selection
2	Clock pin	RSPCK2	I/O	Clock input/output
	Master transmit data pin	MOSI2	I/O	Master transmit data
	Slave transmit data pin	MISO2	I/O	Slave transmit data
	Slave select 0 pin	SSL20	I/O	Slave selection

Note: In the description of the pins, the channel is omitted and pin names are described as RSPCK, MOSI, MISO, and SSL.

19.3 Register Descriptions

Table 19.2 shows the register configuration. These registers enable this module to perform the following controls: specifying master/slave modes, specifying a transfer format, and controlling the transmitter and receiver.

Table 19.2 Register Configuration

Channel	Register Name	Abbreviation*1	R/W	Initial Value	Address	Access Size
0	Control register_0	SPCR_0	R/W	H'00	H'E800C800	8
	Slave select polarity register_0	SSLP_0	R/W	H'00	H'E800C801	8
	Pin control register_0	SPPCR_0	R/W	H'00	H'E800C802	8
	Status register_0	SPSR_0	R/(W)*2	H'60	H'E800C803	8
	Data register_0	SPDR_0	R/W	Undefined	H'E800C804	8, 16, 32
	Sequence control register_0	SPSCR_0	R/W	H'00	H'E800C808	8
	Sequence status register_0	SPSSR_0	R	H'00	H'E800C809	8
	Bit rate register_0	SPBR_0	R/W	H'FF	H'E800C80A	8
	Data control register_0	SPDCR_0	R/W	H'20	H'E800C80B	8
	Clock delay register_0	SPCKD_0	R/W	H'00	H'E800C80C	8
	Slave select negation delay register_0	SSLND_0	R/W	H'00	H'E800C80D	8
	Next-access delay register_0	SPND_0	R/W	H'00	H'E800C80E	8
	Command register0_0	SPCMD0_0	R/W	H'070D	H'E800C810	16
	Command register1_0	SPCMD1_0	R/W	H'070D	H'E800C812	16
	Command register2_0	SPCMD2_0	R/W	H'070D	H'E800C814	16
	Command register3_0	SPCMD3_0	R/W	H'070D	H'E800C816	16
	Buffer control register_0	SPBFCR_0	R/W	H'00	H'E800C820	8
	Buffer data count setting register_0	SPBFDR_0	R	H'0000	H'E800C822	16
1	Control register_1	SPCR_1	R/W	H'00	H'E800D000	8
	Slave select polarity register_1	SSLP_1	R/W	H'00	H'E800D001	8
	Pin control register_1	SPPCR_1	R/W	H'00	H'E800D002	8
	Status register_1	SPSR_1	R/(W)*2	H'60	H'E800D003	8
	Data register_1	SPDR_1	R/W	Undefined	H'E800D004	8, 16, 32
	Sequence control register_1	SPSCR_1	R/W	H'00	H'E800D008	8
	Sequence status register_1	SPSSR_1	R	H'00	H'E800D009	8
	Bit rate register_1	SPBR_1	R/W	H'FF	H'E800D00A	8
	Data control register_1	SPDCR_1	R/W	H'20	H'E800D00B	8
	Clock delay register_1	SPCKD_1	R/W	H'00	H'E800D00C	8
	Slave select negation delay register_1	SSLND_1	R/W	H'00	H'E800D00D	8
	Next-access delay register_1	SPND_1	R/W	H'00	H'E800D00E	8
	Command register0_1	SPCMD0_1	R/W	H'070D	H'E800D010	16
	Command register1_1	SPCMD1_1	R/W	H'070D	H'E800D012	16
	Command register2_1	SPCMD2_1	R/W	H'070D	H'E800D014	16
	Command register3_1	SPCMD3_1	R/W	H'070D	H'E800D016	16
	Buffer control register_1	SPBFCR_1	R/W	H'00	H'E800D020	8
	Buffer data count setting register_1	SPBFDR_1	R	H'0000	H'E800D022	16

Table 19.2 Register Configuration

Channel	Register Name	Abbreviation*1	R/W	Initial Value	Address	Access Size
2	Control register_2	SPCR_2	R/W	H'00	H'E800D800	8
	Slave select polarity register_2	SSLP_2	R/W	H'00	H'E800D801	8
	Pin control register_2	SPPCR_2	R/W	H'00	H'E800D802	8
	Status register_2	SPSR_2	R/(W)*2	H'60	H'E800D803	8
	Data register_2	SPDR_2	R/W	Undefined	H'E800D804	8, 16, 32
	Sequence control register_2	SPSCR_2	R/W	H'00	H'E800D808	8
	Sequence status register_2	SPSSR_2	R	H'00	H'E800D809	8
	Bit rate register_2	SPBR_2	R/W	H'FF	H'E800D80A	8
	Data control register_2	SPDCR_2	R/W	H'20	H'E800D80B	8
	Clock delay register_2	SPCKD_2	R/W	H'00	H'E800D80C	8
	Slave select negation delay register_2	SSLND_2	R/W	H'00	H'E800D80D	8
	Next-access delay register_2	SPND_2	R/W	H'00	H'E800D80E	8
	Command register0_2	SPCMD0_2	R/W	H'070D	H'E800D810	16
	Command register1_2	SPCMD1_2	R/W	H'070D	H'E800D812	16
	Command register2_2	SPCMD2_2	R/W	H'070D	H'E800D814	16
	Command register3_2	SPCMD3_2	R/W	H'070D	H'E800D816	16
	Buffer control register_2	SPBFCR_2	R/W	H'00	H'E800D820	8
	Buffer data count setting register_2	SPBFDR_2	R	H'0000	H'E800D822	16

Note 1. In the description of the register names, the channel is omitted.
Note 2. Only 0 can be written to clear the flag.

19.3.1 Control Register (SPCR)

SPCR sets the operating mode. If the MSTR and MODFEN bits are changed while the function of this module is enabled by setting the SPE bit to 1, subsequent operations cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	SPRIE	SPE	SPTIE	SPEIE	MSTR	MOD FEN	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Function
7	SPRIE	0	R/W	Receive Interrupt Enable Enables or disables generation of receive interrupt requests (SPRI) when the number of receive data units in the receive buffer (SPRX) is equal to or greater than the specified receive buffer data triggering number and the SPRF flag in SPSR is set to 1. 0: Disables the generation of receive interrupt requests. 1: Enables the generation of receive interrupt requests.
6	SPE	0	R/W	Function Enable Setting this bit to 1 enables the module function. When the MODF bit in the status register (SPSR) is 1, the SPE bit cannot be set to 1 (see 19.4.6 Error Detection). Setting the SPE bit to 0 disables the module function, and initializes a part of the module function (see 19.4.7 Initialization). 0: Disables the module function. 1: Enables the module function.
5	SPTIE	0	R/W	Transmit Interrupt Enable Enables or disables generation of transmit interrupt requests (SPTI) when the number of transmit data units in the transmit buffer (SPTX) is equal to or less than the specified transmit buffer data triggering number and the SPTEF flag in SPSR is set to 1. 0: Disables the generation of transmit interrupt requests. 1: Enables the generation of transmit interrupt requests.
4	SPEIE	0	R/W	Error Interrupt Enable Enables or disables the generation of error interrupt requests when this module detects a mode fault error and sets the MODF bit in the status register (SPSR) to 1, or when this module detects an overrun error and sets the OVRF bit in SPSR to 1 (see 19.4.6 Error Detection). 0: Disables the generation of error interrupt requests. 1: Enables the generation of error interrupt requests. Note: This bit is valid only in SPI slave mode.
3	MSTR	0	R/W	Master/Slave Mode Select Selects master/slave mode. According to MSTR bit settings, this module determines the direction of the RSPCK, MOSI, MISO, and SSL pins. 0: Slave mode 1: Master mode
2	MODFEN	0	R/W	Mode Fault Error Detection Enable Enables or disables the detection of a mode fault error (see 19.4.6 Error Detection). 0: Disables the detection of a mode fault error. 1: Enables the detection of a mode fault error. Note: This bit is valid only in SPI slave mode. When master mode is specified with the MSTR bit, this bit should always be cleared to 0.
1, 0	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

19.3.2 Slave Select Polarity Register (SSLP)

SSLP sets the polarity of the SSL signal. If the contents of SSL0P are changed while the function of this module is enabled by setting the SPE bit in the control register (SPCR) to 1, subsequent operations cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SSL0P
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Function
7 to 1	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	SSL0P	0	R/W	SSL Signal Polarity Setting Sets the polarity of the SSL signal. The value of SSL0P indicates the active polarity of the SSL signal. 0: SSL signal 0-active 1: SSL signal 1-active

19.3.3 Pin Control Register (SPPCR)

SPPCR sets the modes of the pins. If the contents of this register are changed while the function of this module is enabled by setting the SPE bit in the control register (SPCR) to 1, subsequent operations cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	MOIFE	MOIFV	—	—	—	SPLP
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Function
7, 6	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
5	MOIFE	0	R/W	MOSI Idle Value Fixing Enable Fixes the MOSI output value when this module in master mode is in an SSL negation period (including the SSL retention period during a burst transfer). When MOIFE is 0, this module outputs the last output value from the previous serial transfer during the SSL negation period to the MOSI pin. (The value is undefined when CPHA is 0). When MOIFE is 1, this module outputs the fixed value set in the MOIFV bit to the MOSI pin. 0: MOSI output value equals the last output value from previous transfer. (The value is undefined when CPHA is 0). 1: MOSI output value equals the value set in the MOIFV bit.
4	MOIFV	0	R/W	MOSI Idle Fixed Value If the MOIFE bit is 1 in master mode, this module, according to MOIFV bit settings, determines the MOSI signal value during the SSL negation period (including the SSL retention period during a burst transfer). 0: MOSI Idle fixed value equals 0. 1: MOSI Idle fixed value equals 1.
3 to 1	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	SPLP	0	R/W	Loopback When the SPLP bit is set to 1, this module shuts off the path between the MISO pin and the shift register, and between the MOSI pin and the shift register, and connects (reverses) the input path and the output path for the shift register. 0: Normal mode 1: Loopback mode

19.3.4 Status Register (SPSR)

SPSR indicates the operating status.

Bit:	7	6	5	4	3	2	1	0
	SPRF	TEND	SPTEF	—	—	MODF	—	OVRF
Initial value:	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/(W)*	R	R/(W)*

Note: *Only 0 can be written to clear the flag after reading 1.

Bit	Bit Name	Initial Value	R/W	Function
7	SPRF	0	R	Receive Buffer Full Flag Indicates that the number of receive data units in the receive buffer (SPRX) is equal to or greater than the receive buffer data triggering number specified in the buffer control register (SPBFCR). 0: The number of receive data units in the receive buffer is less than the receive buffer data triggering number. 1: The number of receive data units in the receive buffer is equal to or greater than the receive buffer data triggering number. [Clearing conditions] • The receive buffer data is read until the number of data units in the receive buffer becomes less than the specified receive buffer data triggering number. • Receive buffer data reset is enabled. • Power-on reset [Setting condition] • The number of data units in the receive buffer is equal to or greater than the specified receive buffer data triggering number.
6	TEND	1	R	Transmit End This bit is set to 1 when transmission is completed, and this bit is 0 when transmission is not completed. [Clearing condition] • When transmit data are transferred from the transmit register to the shift register. [Setting condition] • When the number of data units in the transmit buffer (SPTX) is zero when a serial transfer is completed. Note: This bit is valid only in SPI master mode.
5	SPTEF	1	R	Transmit Buffer Empty Flag Indicates that the number of transmit data units in the transmit buffer (SPTX) is equal to or less than the transmit buffer data triggering number specified in the buffer control register (SPBFCR). 0: The number of transmit data units in the transmit buffer is equal to or greater than the specified transmit buffer data triggering number. 1: The number of transmit data units in the transmit buffer is less than the specified transmit buffer data triggering number. [Clearing condition] • When data is written to the transmit buffer until the number of transmit data units in the transmit buffer exceeds the specified transmit buffer data triggering number. [Setting conditions] • When the number of transmit data units in the transmit buffer is less than the specified transmit buffer data triggering number. • When transmit buffer data reset is enabled. • Power-on reset
4, 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

Bit	Bit Name	Initial Value	R/W	Function
2	MODF	0	R/(W)*	Mode Fault Error Flag Indicates the occurrence of a mode fault error. If the MODFEN bit is set to 1 when this module is in slave mode and the SSL pin is negated before the RSPCK cycle necessary for data transfer ends, this module detects a mode fault error. The active level of the SSL signal is determined by the SSL0P bit in the slave select polarity register (SSLP). [Clearing conditions] • SPSR is read when the MODF bit is 1, and then 0 is written to the MODF bit. • Power-on reset 0: No mode fault error occurred 1: A mode fault error occurred Note: This bit is valid only in SPI slave mode.
1	—	0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	OVRF	0	R/(W)*	Overrun Error Flag Indicates the occurrence of an overrun error. If a serial transfer ends when there is not enough space for receiving the specified length of data in the receive buffer (SPRX), this module detects an overrun error, and sets the OVRF bit to 1. [Clearing conditions] • SPSR is read when the OVRF bit is 1, and then 0 is written to the OVRF bit. • Power-on reset 0: No overrun error occurred 1: An overrun error occurred Note: This bit is valid only in SPI slave mode.

Note: * Only 0 can be written to clear the flag after reading 1.

19.3.5 Data Register (SPDR)

SPDR is a buffer that holds data for transmission and reception.
The transmit buffer (SPTX) and receive buffer (SPRX) are independent and are mapped to SPDR.
SPDR should be read or written to in byte, word, or longword units according to the access width specification bit (SPLW) in the data control register (SPDCR).
The bit length to be used is determined by the data length specification bits (SPB3 to SPB0) in the command register (SPCMD).
The access width set by SPDCR must agree with the data length set by SPCMD.
When data is written to SPDR, the data will be written to the transmit buffer from SPDR if the transmit buffer has a space equal to or more than the SPDR access width. If there is not enough space, data will not be written to the transmit buffer. Even if an attempt is made to write data to the buffer, the data is ignored.
When data is read from SPDR, receive data in the receive buffer will be read. If SPDR is read when there is no receive data in the receive buffer, the read value is undefined.
When SPDR is written to with the longword-, word-, or byte-access width, the transmit data should be written to address 0 irrespective of the access width. If data is written to the other addresses, the data is not guaranteed.
When SPDR is read with the longword-, word-, or byte-access width, the receive data should be read from address 0. If data is read from the other addresses, the data is not guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.6 Sequence Control Register (SPSCR)

SPSCR sets the sequence control method when this module operates in master mode. If the contents of SPSCR are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPSLN1	SPSLN0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Function
7 to 2	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
1 0	SPSLN1 SPSLN0	0 0	R/W R/W	Sequence Length Specification These bits specify a sequence length when this module in master mode performs sequential operations. This module in master mode changes command registers 0 to 3 (SPCMD0 to SPCMD3) to be referenced and the order in which they are referenced according to the sequence length that is set in the SPSLN1 and SPSLN0 bits. The relationship among the setting of bits SPSLN1 and SPSLN0, sequence length, and SPCMD0 to SPCMD3 referenced by this module is shown below. In slave mode, SPCMD0 is always referenced. Sequence Length / Referenced SPCMD # 00: 1 0 →0 → … 01: 2 0 →1 → 0 → … 10: 3 0 →1 → 2 → 0 → … 11: 4 0 →1 → 2 → 3 → 0 → …

19.3.7 Sequence Status Register (SPSSR)

SPSSR indicates the sequence control status when this module operates in master mode.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPCP1	SPCP0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Function
7 to 2	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
1 0	SPCP1 SPCP0	0 0	R R	Command Pointer During sequence control, these bits indicate one of the command registers 0 to 3 (SPCMD0 to SPCMD3) that is currently pointed to by the pointer. The relationship between the setting of SPCP1 and SPCP0 and SPCMD0 to SPCMD3 is shown below. For the sequence control, see section 19.4.8, (1), (c) Sequence Control. 00: SPCMD0 01: SPCMD1 10: SPCMD2 11: SPCMD3

19.3.8 Bit Rate Register (SPBR)

SPBR sets the bit rate in master mode. If the contents of SPBR are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When this module is used in slave mode, the bit rate depends on the bit rate of the input clock regardless of the settings of SPBR and BRDV.

The bit rate is determined by combinations of SPBR settings and the bit settings in the BRDV1 and BRDV0 bits in the command registers (SPCMD0 to SPCMD3). The equation for calculating the bit rate is given below. In the equation, n denotes an SPBR setting (0, 1, 2, …, 255), and N denotes a BRDV1 and BRDV0 bit setting (0, 1, 2, 3).

$$\text{Bit rate} = \frac{f(P1\varphi)}{2 \times (n + 1) \times 2^N}$$

Table 19.3 shows examples of the relationship between the SPBR register and BRDV1 and BRDV0 bit settings.

Table 19.3 Relationship between SPBR and BRDV1 and BRDV0 Settings

			Bit Rate
SPBR (n)	BRDV[1:0] (N)	Division Ratio	P1φ = 66MHz
0	0	2*	33.00 Mbps
1	0	4	16.50 Mbps
2	0	6	11.00 Mbps
3	0	8	8.25 Mbps
4	0	10	6.60 Mbps
5	0	12	5.50 Mbps
5	1	24	2.75 Mbps
5	2	48	1.38 Mbps
5	3	96	687.50 Kbps
255	3	4096	16.11 Kbps

Note: * Decide the bit rate to be actually used in the system considering timing specifications.

19.3.9 Data Control Register (SPDCR)

SPDCR selects the width to access SPDR from longword-, word-, and byte-width, and enables or disables dummy data transmission for the master mode operation.

If the contents of SPDCR are changed while bit TEND in the status register (SPSR) indicates that transmission is not completed, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	TXDMY	SPLW1	SPLW0	—	—	—	—	—
Initial value:	0	0	1	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Function
7	TXDMY	0	R/W	Dummy Data Transmission Enable Enables or disables dummy data transmission. When communication is performed with this bit set to 1, dummy data is transmitted from the MOSI pin and a serial communication can be performed even if there is no transmit data in the transmit buffer. Specifically, if there is no transmit data in the transmit buffer and this bit is set to 1, dummy data is transferred to the shift register. Data previously transmitted from the pin is used as dummy data. If this bit is set to 1 after the initialization and a transfer is performed, the transmitted dummy data is undefined. 0: Disables dummy data transmission. 1: Enables dummy data transmission. Note: This bit is valid only in the master mode.
6 5	SPLW1 SPLW0	0 1	R/W R/W	Access Width Specification Specifies the width for accessing the data register (SPDR). If the length of data transferred to SPDR does not agree with these bit settings, operation is not guaranteed.* 00: Setting prohibited 01: SPDR is accessed in bytes (8 bits). 10: SPDR is accessed in words (16 bits). 11: SPDR is accessed in longwords (32 bits).
4 to 0	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

Note: * The data length is specified by the SPB3 to SPB0 bits in the command register (SPCMD). See section 19.3.5, Data Register (SPDR).

19.3.10 Clock Delay Register (SPCKD)

SPCKD sets a period from the beginning of SSL signal assertion to RSPCK oscillation (RSPCK delay) when the SCKDEN bit in the command register (SPCMD) is 1. If the contents of SPCKD are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.
When using this module in slave mode, set B'000 to SCKDL2 to SCKDL0.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCKDL2	SCKDL1	SCKDL0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Function
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2 1 0	SCKDL2 SCKDL1 SCKDL0	0 0 0	R/W R/W R/W	RSPCK Delay Setting These bits set an RSPCK delay value when the SCKDEN bit in SPCMD is 1. The relationship between the setting of SCKDL2 to SCKDL0 and the RSPCK delay value is shown below. 000: 1 RSPCK 001: 2 RSPCK 010: 3 RSPCK 011: 4 RSPCK 100: 5 RSPCK 101: 6 RSPCK 110: 7 RSPCK 111: 8 RSPCK

19.3.11 Slave Select Negation Delay Register (SSLND)

SSLND sets a period (SSL negation delay) from the transmission of a final RSPCK edge to the negation of the SSL signal during a serial transfer by this module in master mode. If the contents of SSLND are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set B'000 to SLNDL2 to SLNDL0.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLNDL2	SLNDL1	SLNDL0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Function
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2 1 0	SLNDL2 SLNDL1 SLNDL0	0 0 0	R/W R/W R/W	SSL Negation Delay Setting These bits set an SSL negation delay when the SLNDEN bit in SPCMD is 1. The relationship between the setting of SLNDL2 to SLNDL0 and the SSL negation delay value is shown below. 000: 1 RSPCK 001: 2 RSPCK 010: 3 RSPCK 011: 4 RSPCK 100: 5 RSPCK 101: 6 RSPCK 110: 7 RSPCK 111: 8 RSPCK

19.3.12 Next-Access Delay Register (SPND)

SPND sets a non-active period (next-access delay) after termination of a serial transfer when the SPNDEN bit in the command register (SPCMD) is 1. If the contents of SPND are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set B'000 to SPNDL2 to SPNDL0.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SPNDL2	SPNDL1	SPNDL0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Function
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SPNDL2	0	R/W	Next-Access Delay Setting These bits set a next-access delay when the SPNDEN bit in SPCMD is 1. The relationship between the setting of SPNDL2 to SPNDL0 and the next-access delay value is shown below. 000: 1 RSPCK + 2 P1φ 001: 2 RSPCK + 2 P1φ 010: 3 RSPCK + 2 P1φ 011: 4 RSPCK + 2 P1φ 100: 5 RSPCK + 2 P1φ 101: 6 RSPCK + 2 P1φ 110: 7 RSPCK + 2 P1φ 111: 8 RSPCK + 2 P1φ
1	SPNDL1	0	R/W	
0	SPNDL0	0	R/W	

19.3.13 Command Register (SPCMD)

Each channel has four command registers (SPCMD0 to SPCMD3). SPCMD0 to SPCMD3 are used to set a transfer format for master mode operation. Some of the bits in SPCMD0 are used to set a transfer mode for slave mode operation. In master mode, this module sequentially references SPCMD0 to SPCMD3 according to the settings in bits SPSLN1 and SPSLN0 in the sequence control register (SPSCR), and executes the serial transfer that is set in the referenced SPCMD. While bit TEND in the status register (SPSR) indicates that transmission is not completed, correct operation of this module cannot be guaranteed if SPCMD is changed that is referred by this module. SPCMD referenced by this module in master mode can be checked by means of bits SPCP1 and SPCP0 in the sequence status register (SPSSR). When the function of this module in slave mode is enabled, operation cannot be guaranteed if the value set in SPCMD0 is changed.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCK DEN	SLN DEN	SPN DEN	LSBF	SPB3	SPB2	SPB1	SPB0	SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
Initial value:	0	0	0	0	0	1	1	1	0	0	0	0	1	1	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Function
15	SCKDEN	0	R/W	RSPCK Delay Setting Enable Sets the period from the point this module in master mode activates the SSL signal until the RSPCK starts oscillation (RSPCK delay). If the SCKDEN bit is 0, this module sets the RSPCK delay to 1 RSPCK. If the SCKDEN bit is 1, this module starts the oscillation of RSPCK at an RSPCK delay in compliance with the clock delay register (SPCKD) settings. To use this module in slave mode, the SCKDEN bit should be set to 0. 0: An RSPCK delay of 1 RSPCK 1: An RSPCK delay equal to SPCKD settings.
14	SLNDEN	0	R/W	SSL Negation Delay Setting Enable Sets the period from the point this module in master mode stops RSPCK oscillation until this module sets the SSL signal inactive (SSL negation delay). If the SLNDEN bit is 0, this module sets the SSL negation delay to 1 RSPCK. If the SLNDEN bit is 1, this module negates the SSL signal at an SSL negation delay in compliance with the slave select negation delay register (SSLND) settings. To use this module in slave mode, the SLNDEN bit should be set to 0. 0: An SSL negation delay of 1 RSPCK 1: An SSL negation delay equal to SSLND settings.
13	SPNDEN	0	R/W	Next-Access Delay Enable Sets the period from the point this module in master mode terminates a serial transfer and sets the SSL signal inactive until this module enables the SSL signal assertion for the next access (next-access delay). If the SPNDEN bit is 0, this module sets the next-access delay to 1 RSPCK + 2P1φ. If the SPNDEN bit is 1, this module inserts a next-access delay in compliance with the next-access delay register (SPND) settings. To use this module in slave mode, the SPNDEN bit should be set to 0. 0: A next-access delay of 1 RSPCK + 2 P1φ 1: A next-access delay equal to SPND settings.
12	LSBF	0	R/W	LSB First Sets the data format in master mode or slave mode to MSB first or LSB first. 0: MSB first 1: LSB first
11 10 9 8	SPB3 SPB2 SPB1 SPB0	0 1 1 1	R/W R/W R/W R/W	Data Length Setting These bits set a transfer data length in master mode or slave mode. 0100 to 0111: 8 bits 1111: 16 bits 0010, 0011: 32 bits Others: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Function
7	SSLKP	0	R/W	SSL Signal Level Keeping When this module in master mode performs a serial transfer, this bit specifies whether the SSL signal level for the current command is to be kept or negated between the SSL negation timing associated with the current command and the SSL assertion timing associated with the next command. To use this module in slave mode, the SSLKP bit should be set to 0. 0: Negates the SSL signal upon completion of transfer. 1: Keeps the SSL signal level from the end of the transfer until the beginning of the next access.
6 to 4	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
3 2	BRDV1 BRDV0	1 1	R/W R/W	Bit Rate Division Setting These bits are used to determine the bit rate. A bit rate is determined by combinations of bits BRDV1 and BRDV 0 and the settings in the bit rate register (SPBR) (see section 19.3.8, Bit Rate Register (SPBR)). The settings in SPBR determine the base bit rate. The settings in bits BRDV1 and BRDV0 are used to select a bit rate which is obtained by dividing the base bit rate by 1, 2, 4, or 8. In the bits SPCMD0 to SPCMD3, different BRDV1 and BRDV0 settings can be specified. This permits the execution of serial transfers at a different bit rate for each command. 00: Select the base bit rate. 01: Select the base bit rate divided by 2. 10: Select the base bit rate divided by 4. 11: Select the base bit rate divided by 8.
1	CPOL	0	R/W	RSPCK Polarity Setting Sets an RSPCK polarity in master or slave mode. When data communication is performed between the Renesas serial peripheral interface module and the other modules, the same RSPCK polarity should be set for both modules. 0: RSPCK = 0 when idle 1: RSPCK = 1 when idle
0	CPHA	1	R/W	RSPCK Phase Setting Sets an RSPCK phase in master or slave mode. When data communication is performed between the Renesas serial peripheral interface module and the other modules, the same RSPCK phase should be set for both modules. 0: Data sampling on odd edge, data variation on even edge 1: Data variation on odd edge, data sampling on even edge

19.3.14 Buffer Control Register (SPBFCR)

SPBFCR resets the number of data units in the transmit buffer (SPTX) or receive buffer (SPRX) and sets the number of triggering data units.

Bit:	7	6	5	4	3	2	1	0
	TXRST	RXRST	TXTRG[1:0]		—	RXTRG[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Function
7	TXRST	0	R/W	Transmit Buffer Data Reset Resets the transmit buffer to an empty state. Transmit data in the transmit buffer becomes invalid when this bit is set to 1. 0: Disables the reset operation.* 1: Enables the reset operation Note: * The reset operation is performed after a power-on reset.
6	RXRST	0	R/W	Receive Buffer Data Reset Resets the receive buffer to an empty state. Receive data in the receive buffer becomes invalid when this bit is set to 1. 0: Disables the reset operation.* 1: Enables the reset operation Note: * The reset operation is performed after a power-on reset.
5, 4	TXTRG[1:0]	00	R/W	Transmit Buffer Data Triggering Number Specifies the timing at which the transmit buffer empty state is determined, that is when the SPTEF flag in the status register is set. When the number of bytes of data in the transmit buffer (SPTX) is equal to or less than the specified triggering number, the SPTEF flag is set to 1. 00: 7 bytes (1)* 01: 6 bytes (2)* 10: 4 bytes (4)* 11: 0 bytes (8)* Note: * The value in the parenthesis shows the number of available bytes in the transmit buffer (SPTX).
3	—	0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2 to 0	RXTRG[2:0]	000	R/W	Receive Buffer Data Triggering Number Specifies the timing at which the receive buffer full state is determined, that is when the SPRF flag in the status register is set. When the number of bytes of data in the receive buffer (SPRX) is equal to or greater than the specified triggering number, the SPRF flag is set to 1. 000: 1 byte (31)* 001: 2 bytes (30)* 010: 4 bytes (28)* 011: 8 bytes (24)* 100: 16 bytes (16)* 101: 24 bytes (8)* 110: 32 bytes (0)* 111: 5 bytes (27)* Note: * The value in the parenthesis shows the number of available bytes in the receive buffer (SPRX).

19.3.15 Buffer Data Count Setting Register (SPBFDR)

SPBFDR indicates the number of data units stored in the transmit buffer (SPTX) and receive buffer (SPRX). The upper eight bits indicate the number of transmit data units in SPTX and the lower eight bits indicate the number of receive data units in SPRX.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	T[3:0]				—	—	R[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Function
15 to 12	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
11 to 8	T[3:0]	0000	R	Indicates the number of bytes of data to be transmitted in SPTX. B'0000 indicates that SPTX is empty. B'1000 indicates that SPTX is full.
7, 6	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
5 to 0	R[5:0]	000000	R	Shows the number of bytes of received data in SPTX. B'000000 indicates that SPRX is empty. B'100000 indicates that SPRX is full.

19.4 Operation

In this section, the serial transfer period means a period from the beginning of driving valid data to the fetching of the final valid data.

19.4.1 Overview of Operations

This module is capable of serial transfers in slave mode and master mode. A particular mode of this module can be selected by using the MSTR bit in the control register (SPCR). Table 19.4 gives the relationship between the modes and SPCR settings, and a description of each mode.

Table 19.4 Relationship between Modes and SPCR and Description of Each Mode

Mode	Slave (SPI Operation)	Master (SPI Operation)
MSTR bit setting	0	1
MODFEN bit setting	0 or 1	0
RSPCK signal	Input	Output
MOSI signal	Input	Output
MISO signal	Output/Hi-Z	Input
SSL signal	Input	Output
SSL polarity modification function	Supported	Supported
Transfer rate	Up to P1φ/8	Up to P1φ/2
Clock source	RSPCK input	On-chip baud rate generator
Clock polarity	Two	Two
Clock phase	Two	Two
First transfer bit	MSB/LSB	MSB/LSB
Transfer data length	8, 16, or 32 bits	8, 16, or 32 bits
Burst transfer	Possible (CPHA = 1)	Possible (CPHA = 0,1)
RSPCK delay control	Not supported	Supported
SSL negation delay control	Not supported	Supported
Next-access delay control	Not supported	Supported
Transfer activation method	SSL input active or RSPCK oscillation	Transmit buffer is written when SPE = 1
Sequence control	Not supported	Supported
Transmit buffer empty detection	Supported	Supported
Receive buffer full detection	Supported	Supported
Overrun error detection	Supported	Not Supported
Mode fault error detection	Supported (MODFEN = 1)	Not supported

19.4.2 Pin Control

According to the MSTR bit in the control register (SPCR), this module can automatically switch pin directions and output modes. Table 19.5 shows the relationship between pin states and bit settings.

Table 19.5 Relationship between Pin States and Bit Settings

Mode	Pin	Pin State*
Master mode (SPI operation) (MSTR = 1)	RSPCK	CMOS output
	SSL	CMOS output
	MOSI	CMOS output
	MISO	Input
Slave mode (SPI operation) (MSTR = 0)	RSPCK	Input
	SSL	Input
	MOSI	Input
	MISO*	CMOS output/Hi-Z

Note: * When SSL is at the non-active level or the SPE bit in SPCR is cleared to 0, the pin state is Hi-Z.

This module in master mode (SPI operation) determines MOSI signal values during the SSL negation period (including the SSL retention period during a burst transfer) according to MOIFE and MOIFV bit settings in SPPCR, as shown in Table 19.6.

Table 19.6 MOSI Signal Value Determination during SSL Negation Period

MOIFE	MOIFV	MOSI Signal Value during SSL Negation Period
0	0, 1	Last output value from previous transfer (The value is undefined when CPHA is 0)
1	0	Always 0
1	1	Always 1

19.4.3 System Configuration Example

(1) Master/Slave (with This LSI Acting as Master)

Figure 19.2 shows a master/slave system configuration example when this LSI is used as a master. In master/slave configuration, the SSL output of this LSI (master) is not used. The SSL input of the slave is fixed to the low level, and the slave is always maintained in a selected state. In the transfer format corresponding to the case where the CPHA bit in the control register (SPCR) is 0, there are slave devices for which the SSL signal cannot be fixed to the active level. In situations where the SSL signal cannot be fixed, the SSL output of this LSI should be connected to the SSL input of the slave device.
This LSI (master) always drives the RSPCK and MOSI. The slave always drives the MISO.

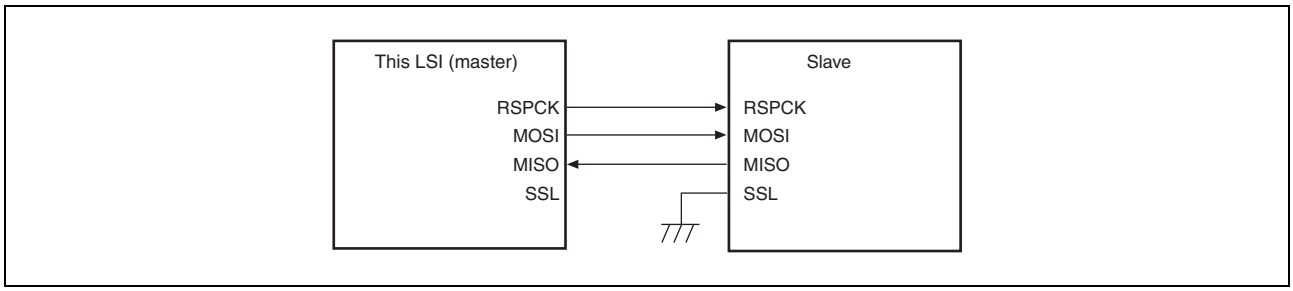


Figure 19.2 Master/Slave Configuration Example (This LSI = Master)

(2) Master/Slave (with This LSI Acting as Slave)

Figure 19.3 shows a master/slave system configuration example when this LSI is used as a slave. When this LSI is to operate as a slave, the SSL pin is used as SSL input. The master always drives the RSPCK and MOSI. This LSI (slave) always drives the MISO. When SSL is at the non-active level, the pin state is Hi-Z.
In the slave configuration in which the CPHA bit in the command register (SPCMD) is set to 1, the SSL input of this LSI (slave) is fixed to the 0 level, this LSI (slave) is always maintained in a selected state, and in this manner it is possible to execute serial transfer (Figure 19.4).

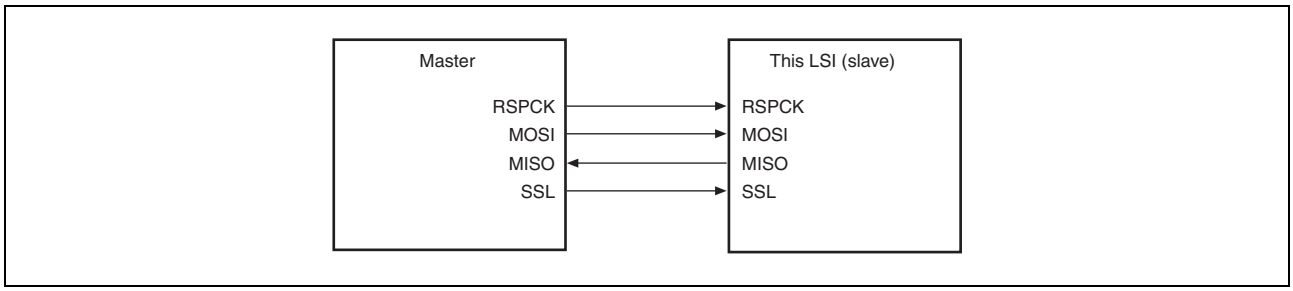


Figure 19.3 Master/Slave Configuration Example (This LSI = Slave)

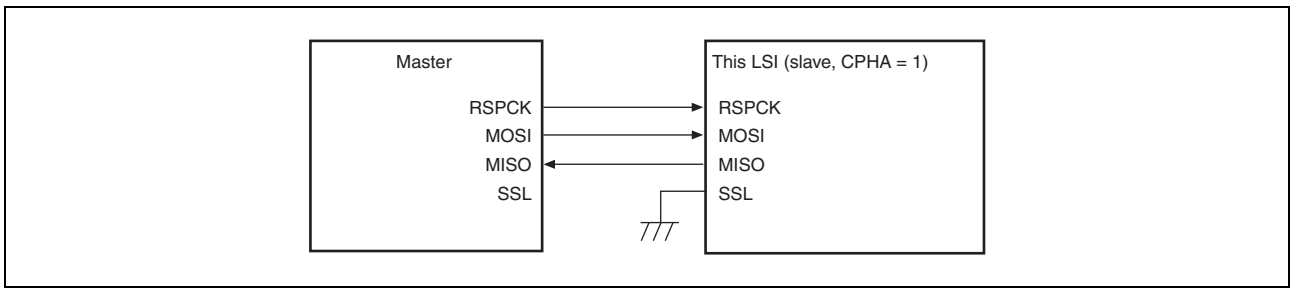


Figure 19.4 Master/Slave Configuration Example (This LSI = Slave, CPHA = 1)

(3) Master/Multi-Slave (with This LSI Acting as Slave)

Figure 19.5 shows a master/multi-slave system configuration example when this LSI is used as a slave. In the example of Figure 19.5, the system is comprised of a master and two LSIs (slave X and slave Y).
The RSPCK and MOSI outputs of the master are connected to the RSPCK and MOSI inputs of the LSIs (slave X and slave Y). The MISO outputs of the LSIs (slave X and slave Y) are all connected to the MISO input of the master. SSLX and SSLY outputs of the master are connected to the SSL inputs of the LSIs (slave X and slave Y), respectively.
The master always drives RSPCK, MOSI, SSLX, and SSLY. Of the LSIs (slave X and slave Y), the slave that receives low level input into the SSL0 input drives MISO.

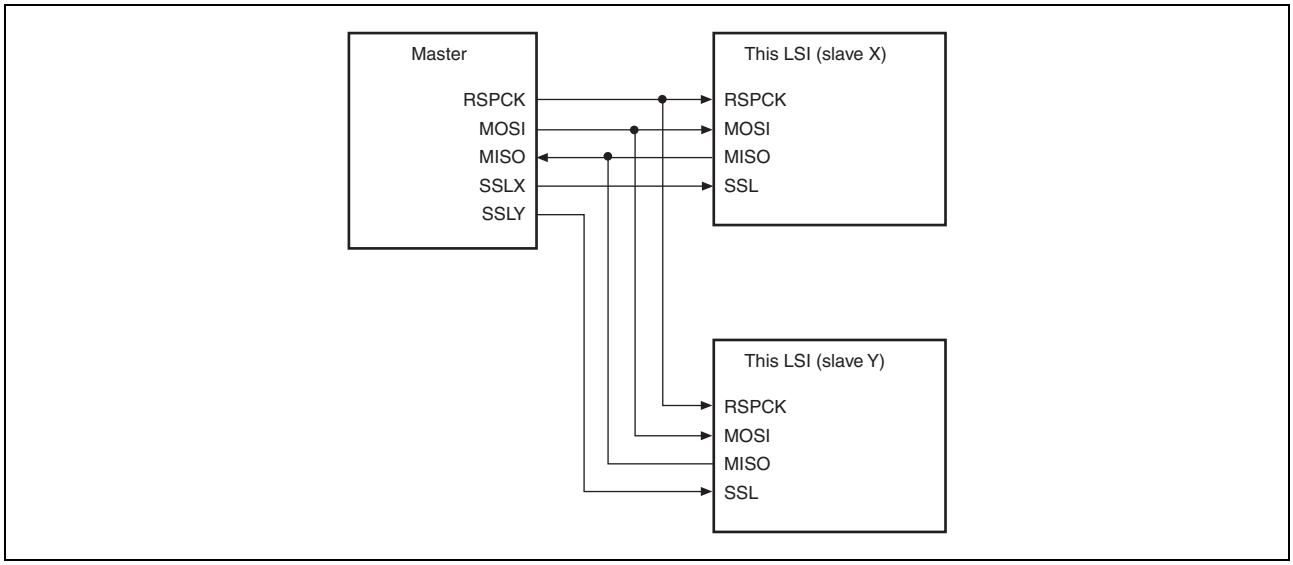


Figure 19.5 Master/Multi-Slave Configuration Example (This LSI = Slave)

19.4.4 Transfer Format

(1) CPHA = 0

Figure 19.6 shows a sample transfer format for the serial transfer of 8-bit data when the CPHA bit in the command register (SPCMD) is 0. In Figure 19.6, RSPCK (CPOL = 0) indicates the RSPCK signal waveform when the CPOL bit in SPCMD is 0; RSPCK (CPOL = 1) indicates the RSPCK signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which this module fetches serial transfer data into the shift register. The input/output directions of the signals depend on the settings of this module. For details, see section 19.4.2, Pin Control.

When the CPHA bit is 0, the driving of valid data to the MOSI and MISO signals commences at an SSL signal assertion timing. The first RSPCK signal change timing that occurs after the SSL signal assertion becomes the first transfer data fetching timing. After this timing, data is sampled at every 1 RSPCK cycle. The change timing for MOSI and MISO signals is always 1/2 RSPCK cycle after the transfer data fetch timing. The settings in the CPOL bit do not affect the RSPCK signal operation timing; they only affect the signal polarity.

t1 denotes a period from an SSL signal assertion to RSPCK oscillation (RSPCK delay). t2 denotes a period from the cessation of RSPCK oscillation to an SSL signal negation (SSL negation delay). t3 denotes a period in which SSL signal assertion is suppressed for the next transfer after the end of serial transfer (next-access delay). t1, t2, and t3 are controlled by a master device running on the system. For a description of t1, t2, and t3 when this module is in master mode, see section 19.4.3, (1) Master/Slave (with This LSI Acting as Master).

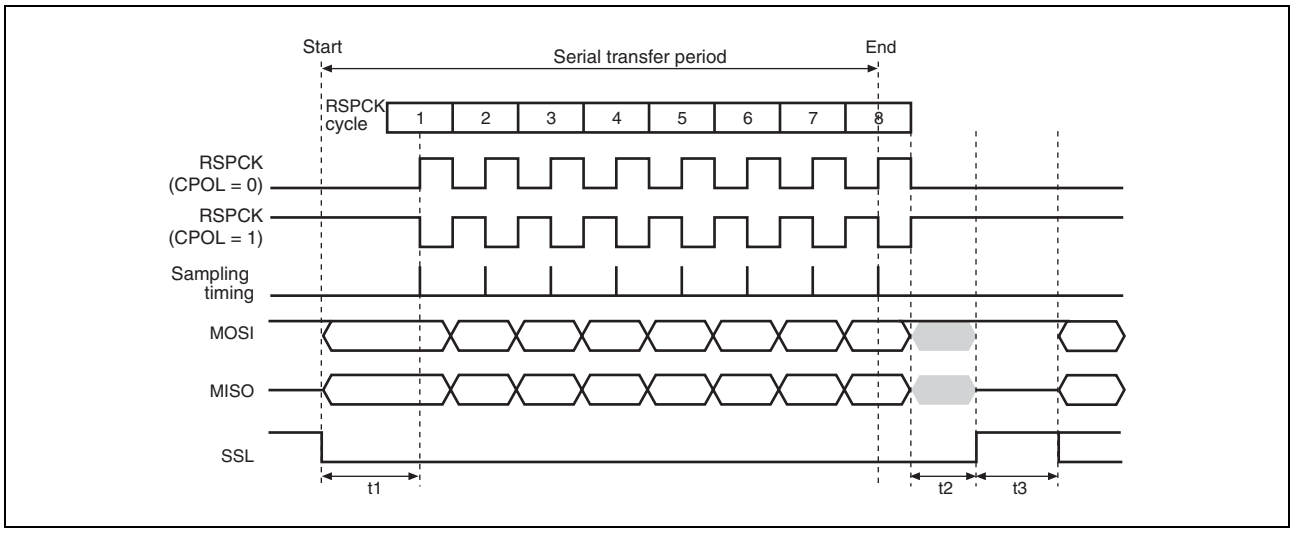


Figure 19.6 Transfer Format (CPHA = 0)

(2) CPHA = 1

Figure 19.7 shows a sample transfer format for the serial transfer of 8-bit data when the CPHA bit in the command register (SPCMD) is 1. In Figure 19.7, RSPCK (CPOL = 0) indicates the RSPCK signal waveform when the CPOL bit in SPCMD is 0; RSPCK (CPOL = 1) indicates the RSPCK signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which this module fetches serial transfer data into the shift register. The input/output directions of the signals depend on the modes (master or slave). For details, see 19.4.2 Pin Control.

When the CPHA bit is 1, the driving of invalid data to the MOSI and MISO signals commences at an SSL signal assertion timing. The driving of valid data to the MOSI and MISO signals commences at the first RSPCK signal change timing that occurs after the SSL signal assertion. After this timing, data is updated at every 1 RSPCK cycle. The transfer data fetch timing is always 1/2 RSPCK cycle after the data update timing. The settings in the CPOL bit do not affect the RSPCK signal operation timing; they only affect the signal polarity.

t1, t2, and t3 are the same as those in the case of CPHA = 0. For a description of t1, t2, and t3 when this module is in master mode, see section 19.4.3, (1) Master/Slave (with This LSI Acting as Master).

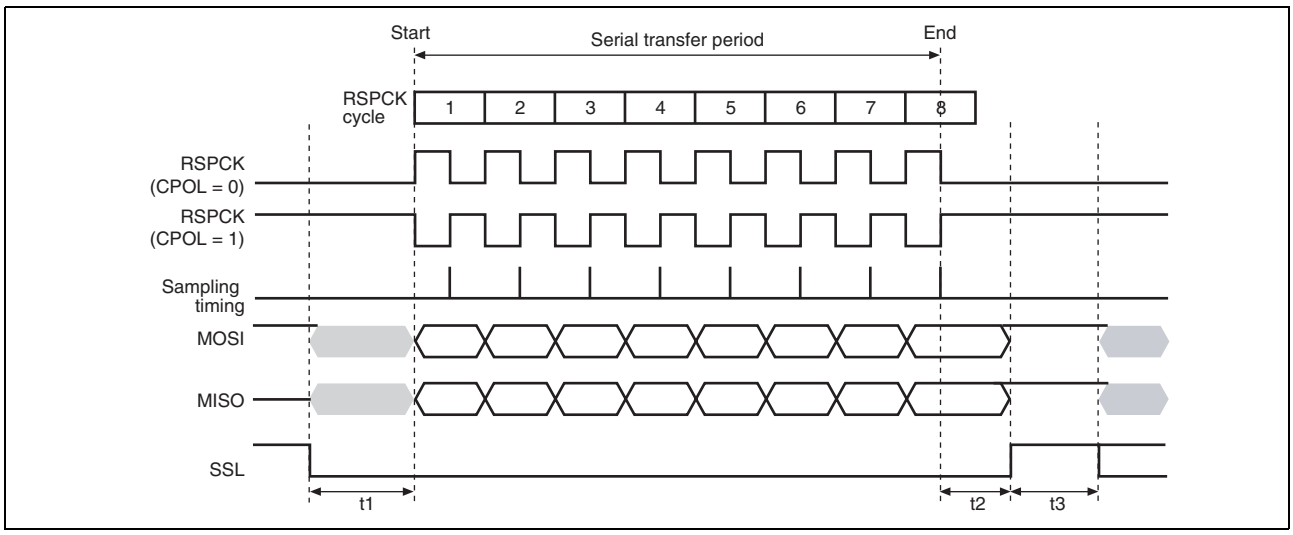


Figure 19.7 Transfer Format (CPHA = 1)

19.4.5 Data Format

The data format depends on the settings in the command register (SPCMD). Irrespective of MSB/LSB first, this module treats the range from the LSB of the data register (SPDR) to the assigned data length as transfer data.

(1) MSB First Transfer (32-Bit Data)

Figure 19.8 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 32-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T31 to T00 to the transmit buffer of SPDR. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R31 to R00 is stored in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after the receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R31 to R00 is shifted out from the shift register.

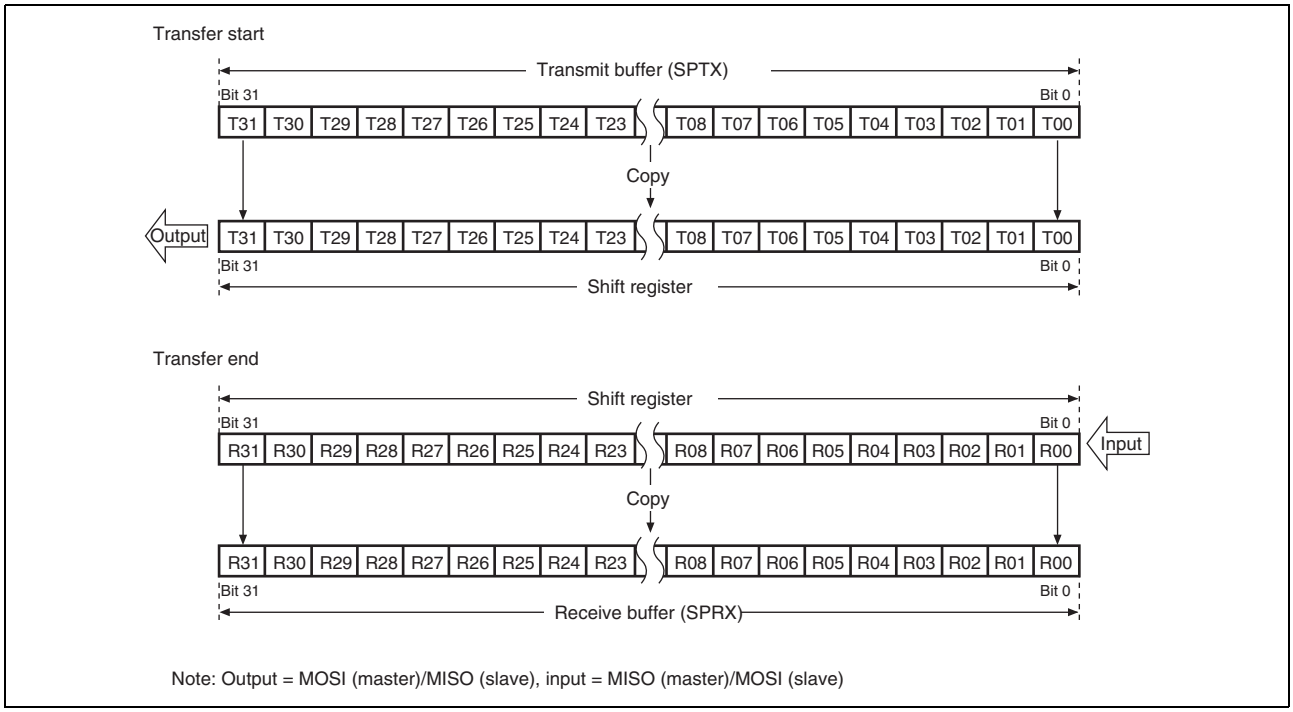


Figure 19.8 MSB First Transfer (32-Bit Data)

(2) MSB First Transfer (16-Bit Data)

Figure 19.9 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 16-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T15 to T00 to the transmit buffer. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from bit 15 in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 16 bits has passed, received data R15 to R00 is stored in bits 15 to 0 in the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 16 in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R15 to R00 is shifted out from the shift register.

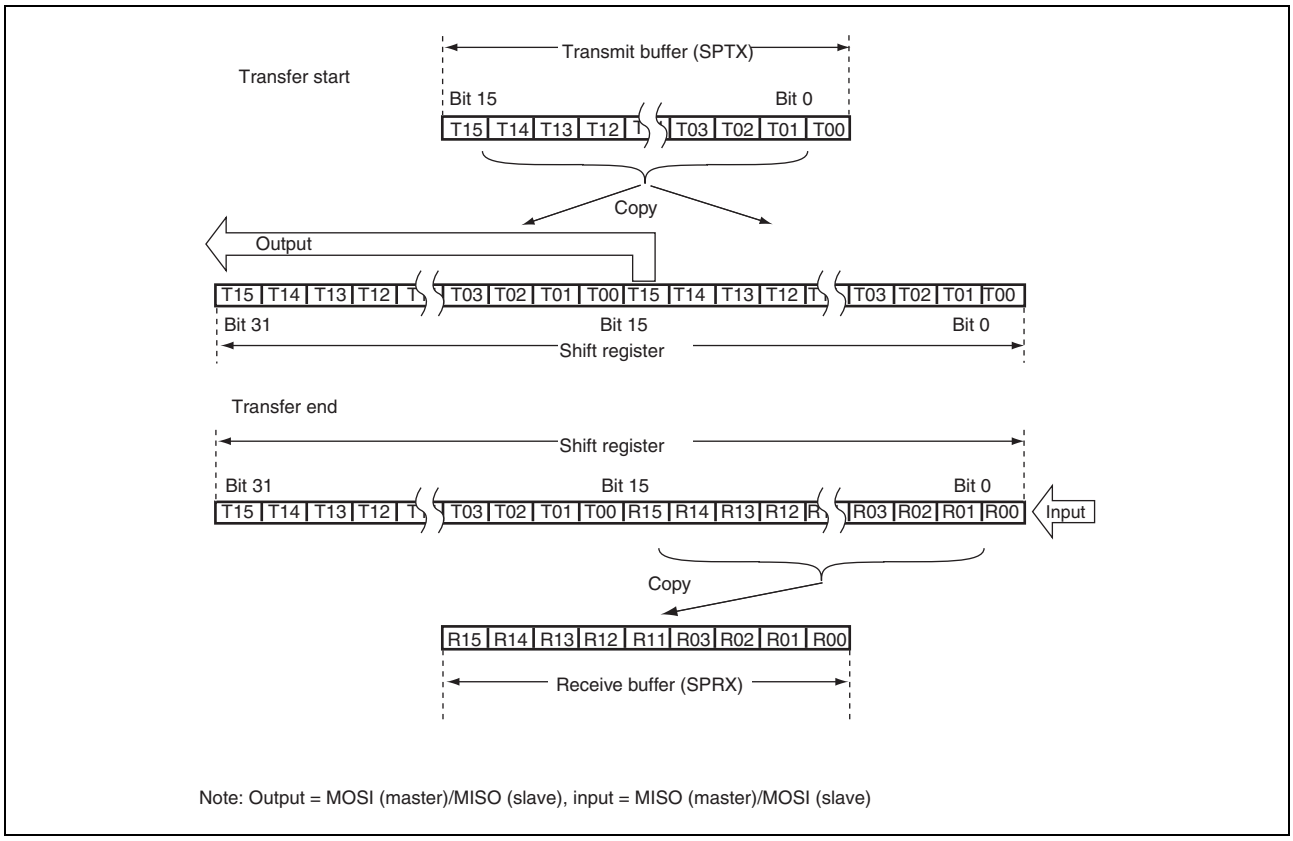


Figure 19.9 MSB First Transfer (16-Bit Data)

(3) MSB First Transfer (8-Bit Data)

Figure 19.10 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs an 8-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T07 to T00 to the transmit buffer. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from bit 7 in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 8 bits has passed, received data R07 to R00 is stored in bits 7 to 0 in the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 8 in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary area in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R07 to R00 is shifted out from the shift register.

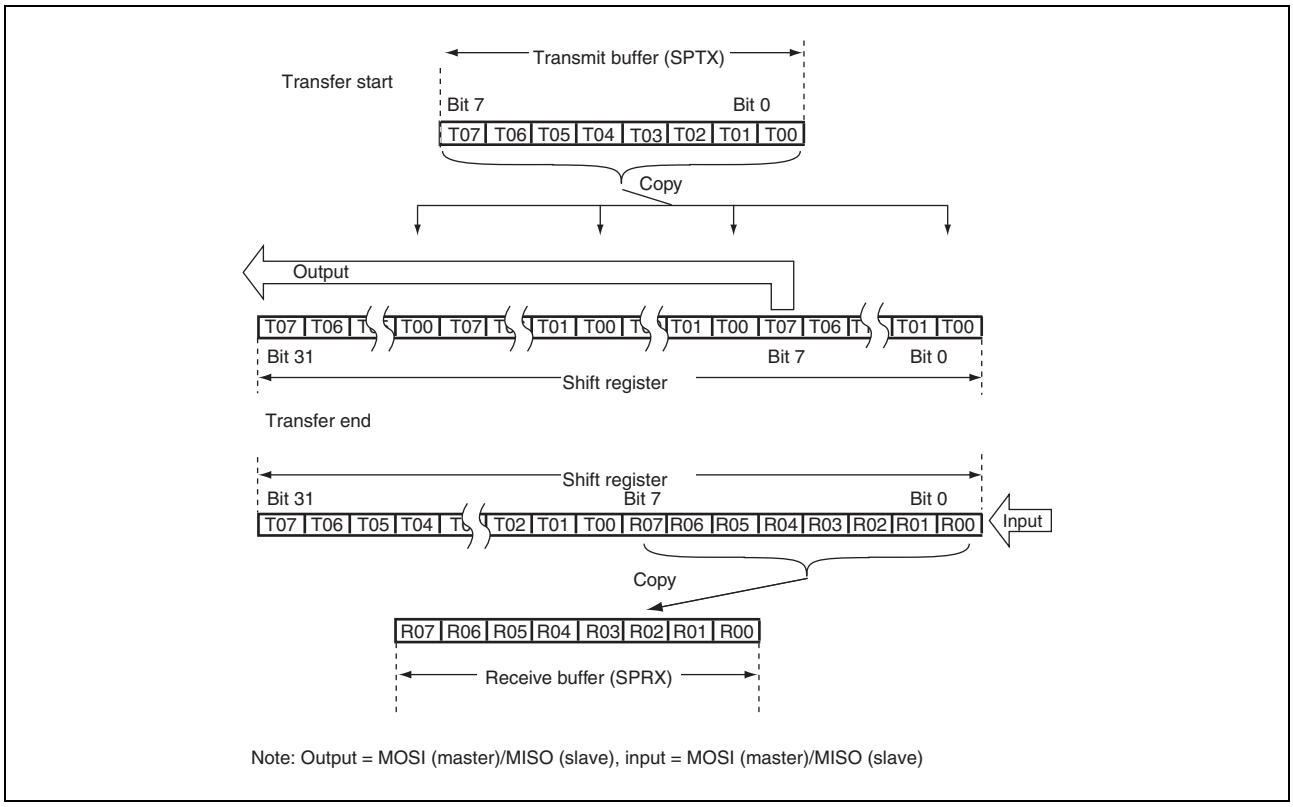


Figure 19.10 MSB First Transfer (8-Bit Data)

(4) LSB First Transfer (32-Bit Data)

Figure 19.11 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 32-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T31 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R00 to R31 is stored in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of the SPDR, received data R00 to R31 is shifted out from the shift register.

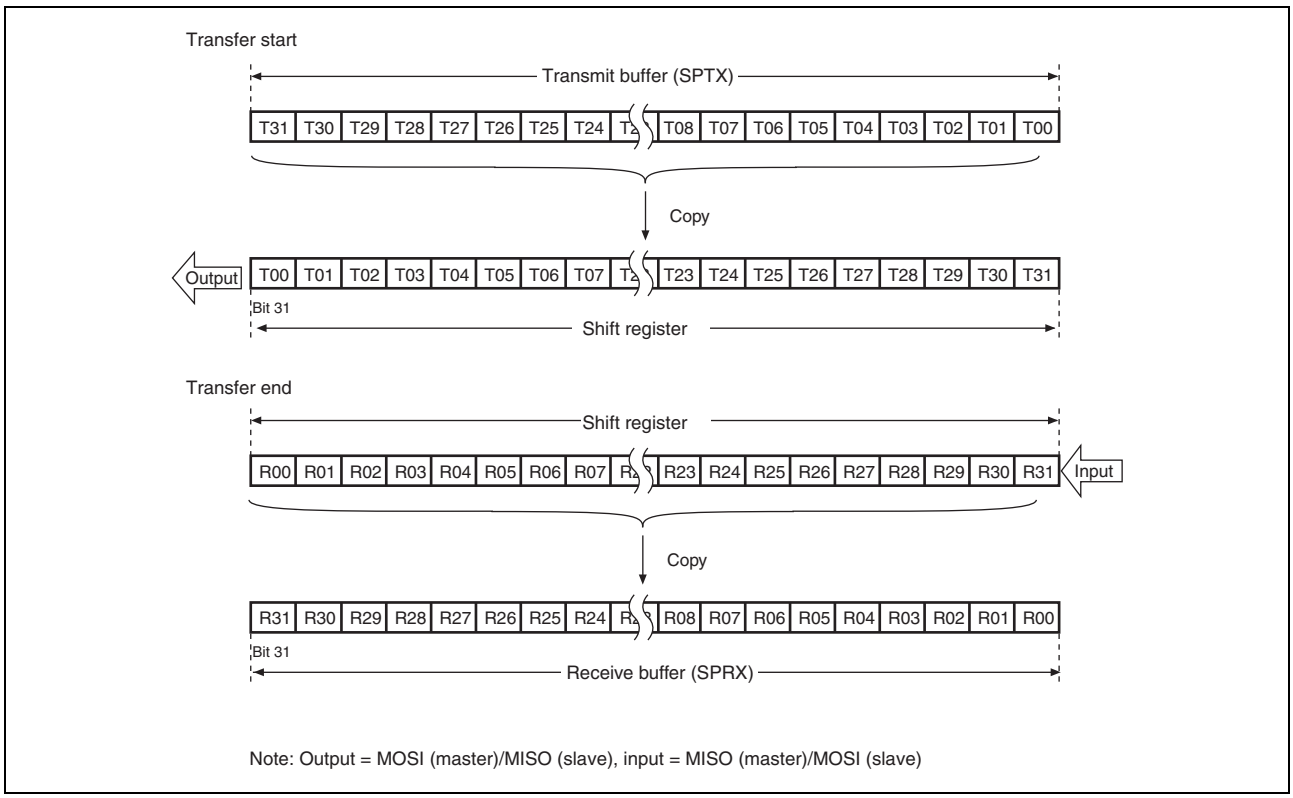


Figure 19.11 LSB First Transfer (32-Bit Data)

(5) LSB First Transfer (16-Bit Data)

Figure 19.12 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 16-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T15 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from bit 16 in the shift register. When the RSPCK cycle required for the serial transfer of 16 bits has passed, received data R00 to R15 is stored in bits 31 to 16 in the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 15 to 0 in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of SPDR, and empties the shift register.

If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of SPDR, received data R00 to R15 is shifted out from the shift register.

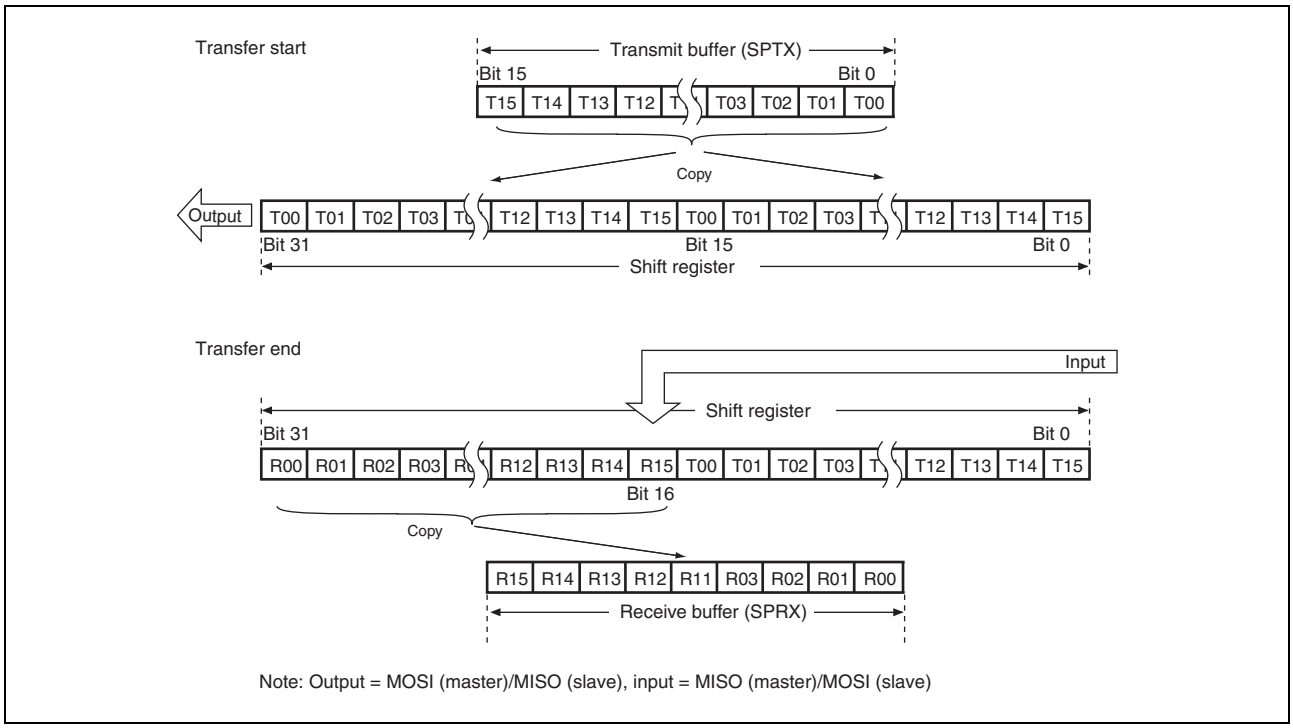


Figure 19.12 LSB First Transfer (16-Bit Data)

(6) LSB First Transfer (8-Bit Data)

Figure 19.13 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs an 8-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T07 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from bit 24 in the shift register. When the RSPCK cycle required for the serial transfer of 8 bits has passed, received data R00 to R07 is stored in bits 31 to 24 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 23 to 0 in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of SPDR, and empties the shift register. If the receive buffer does not have a space for the receive data length after the receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of SPDR, received data R00 to R07 is shifted out from the shift register.

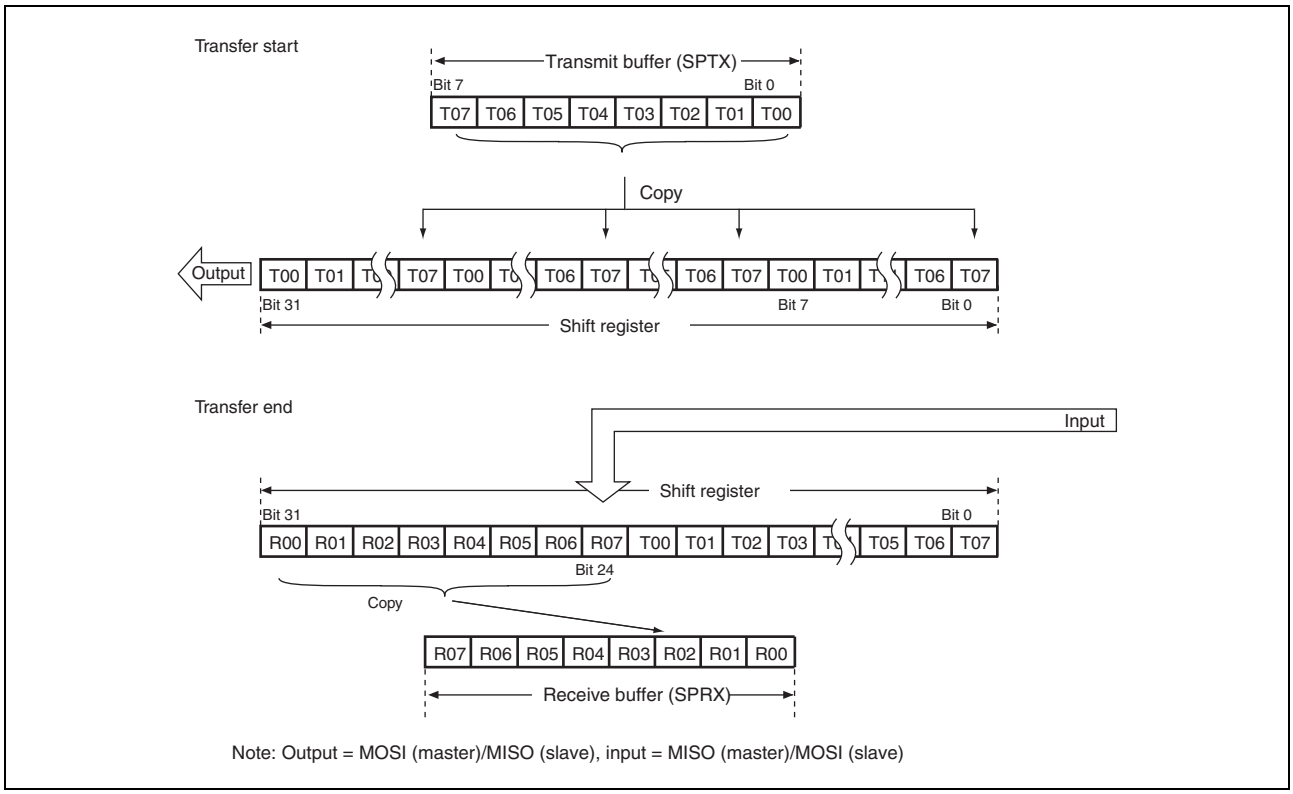


Figure 19.13 LSB First Transfer (8-Bit Data)

19.4.6 Error Detection

In the normal serial transfer, the data written to the transmit buffer of the data register (SPDR) is serially transmitted, and the serially received data can be read from the receive buffer of SPDR. If access is made to SPDR, depending on the status of the transmit buffer/receive buffer or the status at the beginning or end of serial transfer, in some cases non-normal transfers can be executed.
If a non-normal transfer operation occurs, this module detects the event as an overrun error or a mode fault error. Table 19.7 shows the relationship between non-normal transfer operations and the error detection function.

Table 19.7 Relationship between Non-Normal Transfer Operations and Error Detection Function

	Occurrence Condition	Operation	Error Detection
A	SPDR is written when the transmit buffer is full.	Missing write data.	None
B	Serial transfer is started in slave mode when transmit data is still not loaded on the shift register.	Data received in previous serial transfer is serially transmitted.	None
C	SPDR is read when the receive buffer is empty.	The output data is undefined.	None
D	Serial transfer terminates when the receive buffer is full.	Missing serial receive data.	Overrun error (only in slave mode)
E	The SSL input signal is negated during serial transfer in slave mode.	Serial transfer suspended. Missing send/receive data. Operation disabled.	Mode fault error

On operation A shown in Table 19.7, this module does not detect an error. Whether SPDR can be written to or not can be checked using the T[3:0] bits in the buffer data count setting register (SPBFDR).
Likewise, this module does not detect an error on operation B. In a serial transfer that was started before the shift register was updated, this module sends the data that was received in the previous serial transfer, and does not treat the operation indicated in B as an error. Note that the received data from the previous serial transfer is retained in the receive buffer of SPDR, thus it can be correctly read.
Similarly, this module does not detect an error on operation C. To prevent extraneous data from being read, the number of receive data units stored in the receive buffer should be read from the R[5:0] bits in the buffer data count setting register (SPBFDR).
An overrun error shown in D is described in section 19.4.6, (1) Overrun Errors. A mode fault error shown in E is described in section 19.4.6, (2) Mode Fault Error.

(1) Overrun Error

If serial transfer ends when the receive buffer of the data register (SPDR) is full, this module detects an overrun error, and sets the OVRF bit in SPSR to 1. When the OVRF bit is 1, this module does not copy data from the shift register to the receive buffer so that the data prior to the occurrence of the error is retained in the receive buffer. To reset the OVRF bit in SPSR to 0, either perform a power-on reset, or write a 0 to the OVRF bit after SPSR has been read with the OVRF bit set to 1.

Figure 19.14 shows an example of operation of the SPRF and OVRF bits in SPSR. The SPSR and SPDR accesses shown in Figure 19.14 indicate the condition of accesses to SPSR and SPDR, respectively, where I denotes an idle cycle, W a write cycle, and R a read cycle. In the example of Figure 19.14, this module performs an 8-bit serial transfer in which the CPHA bit in the command register (SPCMD) is 1, and CPOL is 0. The numbers given under the RSPCK waveform represent the number of RSPCK cycles (i.e., the number of transferred bits).

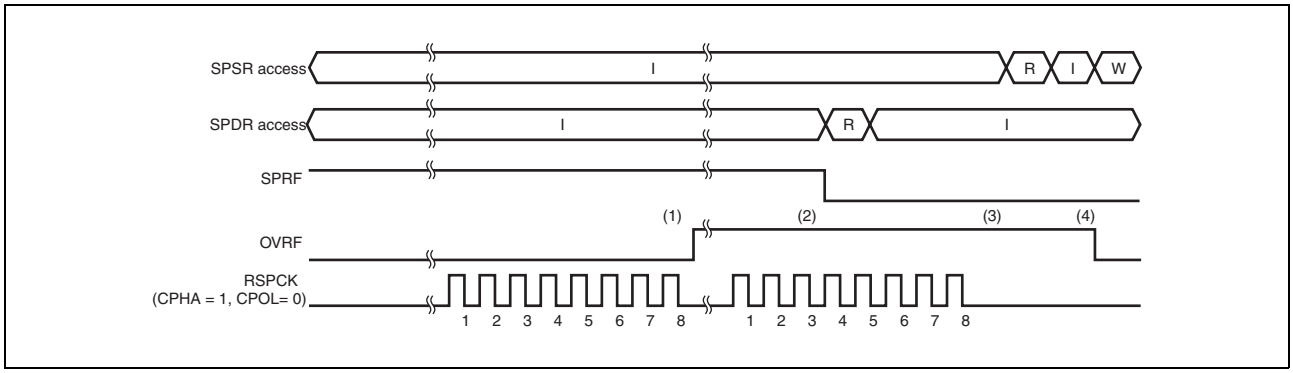


Figure 19.14 SPRF and OVRF Bit Operation Example

The operation of the flags at the timing shown in steps (1) to (4) in the figure is described below.

1. If a serial transfer terminates when the receive buffer does not have a space for the receive data length, this module detects an overrun error, and sets the OVRF bit to 1. This module does not copy the data in the shift register to the receive buffer.
2. The OVFR bit is not cleared even when SPDR is read and thus the number of data bytes in the receive buffer becomes less than the number of the receive buffer data triggering number specified by the RXTRG bits.
3. If the serial transfer terminates in an overrun error state, this module determines that the shift register is empty; in this manner, data transfer is enabled from the transmit buffer to the shift register.
4. If 0 is written to the OVRF bit after SPSR is read with OVRF = 1, this module clears the OVRF bit.

The occurrence of an overrun can be checked either by reading SPSR or by using an error interrupt and reading SPSR. When using an error interrupt, set the SPEIE bit in the control register (SPCR) to 1. When executing a serial transfer without using an error interrupt, measures should be taken to ensure the early detection of overrun errors, such as reading SPSR immediately after SPDR is read.

The OVRF bit is cleared to 0 under the following conditions:

- After SPSR is read in a condition in which the OVRF bit is set to 1, 0 is written to the OVRF bit.
- Power-on reset

Note: When the receive buffer has area enough to store receive data with an overrun error, this module receives receive data.

(2) Mode Fault Error

When the MSTR bit is 0, this module operates in slave mode. This module detects a mode fault error if the SSL input signal is negated during the serial transfer period (from the time the driving of valid data is started to the time the final valid data is fetched) when the MODFEN bit is 1 in slave mode.
Upon detecting a mode fault error, this module stops driving of the output signals and clears the SPE bit in SPCR to 0. When the SPE bit is cleared to 0, the function of this module is disabled and this module stops driving external signals. For details of disabling the function of this module by clearing the SPE bit to 0, see 19.4.7 Initialization.
The occurrence of a mode fault error can be checked either by reading SPSR or by using an error interrupt and reading SPSR. When using an error interrupt, set the SPEIE bit in the control register (SPCR) to 1. To detect a mode fault error without using an error interrupt, it is necessary to poll SPSR.
When the MODF bit is 1, writing 1 to the SPE bit is ignored. To enable the function of this module after the detection of a mode fault error, the MODF bit must be set to 0. The MODF bit is cleared to 0 under the following conditions:

- After SPSR is read in a condition where the MODF bit has turned 1, 0 is written to the MODF bit.
- Power-on reset

19.4.7 Initialization

If 0 is written to the SPE bit in the control register (SPCR) or this module clears the SPE bit to 0 because of the detection of a mode fault error, this module disables the module function, and initializes a part of the module function. When a power-on reset is generated, this module initializes all of the module function. An explanation follows of initialization by the clearing of the SPE bit.

(1) Initialization by Clearing SPE Bit

When the SPE bit in SPCR is cleared, this module performs the following initialization:

- Suspending any serial transfer that is being executed
- Stopping the driving of output signals (Hi-Z) in slave mode
- Initializing the internal state
- Initializing the TEND bit in SPSR

Initialization by the clearing of the SPE bit does not initialize the control bits of this module. For this reason, this module can be started in the same transfer mode as prior to the initialization if the SPE bit is re-set to 1.

19.4.8 SPI Operation

(1) Multi-Master Mode Operation

This section explains the operation in multi-master mode.

(a) Starting Serial Transfer

A serial transfer is started when transmit data is copied from the transmit buffer to the shift register, the shift register becomes full, and the receive buffer has a space for the receive data length. If transmit data has already been written to the shift register, data is not copied from the transmit buffer to the shift register.
For details of the transfer format, see section 19.4.4, Transfer Format.

(b) Terminating Serial Transfer

Irrespective of the CPHA bit in the command register (SPCMD), this module terminates the serial transfer after transmitting an RSPCK edge corresponding to the final sampling timing. After the serial transfer is completed, receive data is copied from the shift register to the receive buffer. If the receive buffer does not have a space for the receive data length after receive data is copied from the shift register to the receive buffer, another serial transfer will not be performed. In order to perform another serial transfer, data for the receive data length should be read from the receive buffer to secure the space for the receive data.
It should be noted that the final sampling timing varies depending on the bit length of transfer data. In master mode, the data length depends on the settings in bits SPB3 to SPB0 in SPCMD. For details on the transfer format, see section 19.4.4, Transfer Format.

(c) Sequence Control

The transfer format that is employed in master mode is determined by the sequence control register (SPSCR), command registers 0 to 3 (SPCMD0 to SPCMD3), the bit rate register (SPBR), the clock delay register (SPCKD), the slave select negation delay register (SSLND), and the next-access delay register (SPND).
SPSCR is a register used to determine the sequence configuration for serial transfers that are executed by this module in master mode. The following items are set in command registers SPCMD0 to SPCMD3: SSL output signal value, MSB/LSB first, data length, some of the bit rate settings, RSPCK polarity/phase, whether SPCKD is to be referenced, whether SSLND is to be referenced, and whether SPND is to be referenced. SPBR holds some of the bit rate settings; SPCKD, a clock delay value; SSLND, an SSL negation delay; and SPND, a next-access delay value.
According to the sequence length that is assigned to SPSCR, this module makes up a sequence comprised of a part or all of SPCMD0 to SPCMD3. This module contains a pointer to the SPCMD that makes up the sequence. The value of this pointer can be checked by reading bits SPCP1 and SPCP0 in the sequence status register (SPSSR). When the SPE bit in the control register (SPCR) is set to 1 and the function of this module is enabled, this module loads the pointer to the commands in SPCMD0, and incorporates the SPCMD0 settings into the transfer format at the beginning of serial transfer. This module increments the pointer each time the next-access delay period for a data transfer ends. Upon completion of the serial transfer that corresponds to the final command comprising the sequence, this module sets the pointer in SPCMD0, and in this manner the sequence is executed repeatedly.

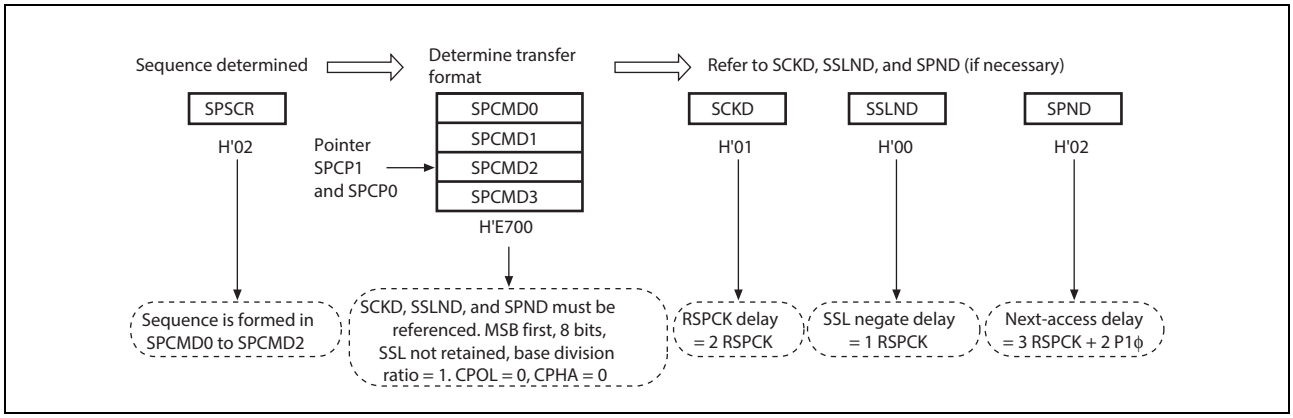


Figure 19.15 Determination Procedure of Serial Transfer Mode in Master Mode

(d) Burst Transfer

If the SSLKP bit in the command register (SPCMD) that this module references during the current serial transfer is 1, this module keeps the SSL signal level during the serial transfer until the beginning of the SSL signal assertion for the next serial transfer. If the SSL signal level for the next serial transfer is the same as the SSL signal level for the current serial transfer, this module can execute continuous serial transfers while keeping the SSL signal assertion status (burst transfer).

Figure 19.16 shows an example of an SSL signal operation for the case where a burst transfer is implemented using SPCMD0 and SPCMD1 settings. The text below explains operations (1) to (7) as depicted in Figure 19.16. It should be noted that the polarity of the SSL output signal depends on the settings in the slave select polarity register (SSLP).

1. Based on SPCMD0, this module asserts the SSL signal and inserts RSPCK delays.
2. Serial transfers are executed according to SPCMD0.
3. SSL negation delays are inserted.
4. Because the SSLKP bit in SPCMD0 is 1, this module keeps the SSL signal value on SPCMD0. This period is sustained, at the shortest, for a period equal to the next-access delay of SPCMD0. If the shift register is empty after the passage of a minimum period, this period is sustained until such time as the transmit data is stored in the shift register for another transfer.
5. Based on SPCMD1, this module asserts the SSL signal and inserts RSPCK delays.
6. Serial transfers are executed according to SPCMD1.
7. Because the SSLKP bit in SPCMD1 is 0, this module negates the SSL signal. In addition, a next-access delay is inserted according to SPCMD1.

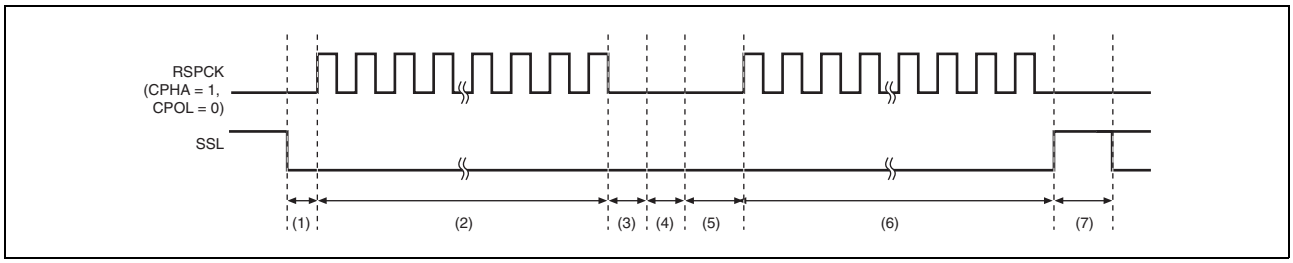


Figure 19.16 Example of Burst Transfer Operation using SSLKP Bit

If the SSL signal settings in the SPCMD in which 1 is assigned to the SSLKP bit are different from the SSL signal output settings in the SPCMD to be used in the next transfer, this module switches the SSL signal status to SSL signal assertion ((5) in Figure 19.16) corresponding to the command for the next transfer. Notice that if such an SSL signal switching occurs, the slaves that drive the MISO signal compete, and the possibility arises of the collision of signal levels.

This module in master mode references within the module the SSL signal operation for the case where the SSLKP bit is

not used. Even when the CPHA bit in SPCMD is 0, this module can accurately start serial transfers by asserting the SSL signal for the next transfer. For this reason, burst transfers in master mode can be executed irrespective of CPHA bit settings (see section 19.4.8, (2) Slave Mode Operation).

(e) RSPCK Delay (t1)

The RSPCK delay value in master mode depends on SCKDEN bit settings in the command register (SPCMD) and on clock delay register (SPCKD) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines an RSPCK delay value during serial transfer by using the SCKDEN bit in the selected SPCMD and SPCKD, as shown in Table 19.8. For a definition of RSPCK delay, see section 19.4.4, Transfer Format.

Table 19.8 Relationship among SCKDEN and SPCKD Settings and RSPCK Delay Values

SCKDEN	SPCKD	RSPCK Delay Value
0	000 to 111	1 RSPCK
1	000	1 RSPCK
	001	2 RSPCK
	010	3 RSPCK
	011	4 RSPCK
	100	5 RSPCK
	101	6 RSPCK
	110	7 RSPCK
	111	8 RSPCK

(f) SSL Negation Delay (t2)

The SSL negation delay value in master mode depends on SLNDEN bit settings in the command register (SPCMD) and on SSL negation delay register (SSLND) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines an SSL negation delay value during serial transfer by using the SLNDEN bit in the selected SPCMD and SSLND, as shown in Table 19.9. For a definition of SSL negation delay, see section 19.4.4, Transfer Format.

Table 19.9 Relationship among SLNDEN and SSLND Settings and SSL Negation Delay Values

SLNDEN	SSLND	SSL Negation Delay Value
0	000 to 111	1 RSPCK
1	000	1 RSPCK
	001	2 RSPCK
	010	3 RSPCK
	011	4 RSPCK
	100	5 RSPCK
	101	6 RSPCK
	110	7 RSPCK
	111	8 RSPCK

(g) Next-Access Delay (t3)

The next-access delay value in master mode depends on SPNDEN bit settings in the command register (SPCMD) and on next-access delay register (SPND) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines a next-access delay value during serial transfer by using the SPNDEN bit in the selected SPCMD and SPND, as shown in Table 19.10. For a definition of next-access delay, see section 19.4.4, Transfer Format.

Table 19.10 Relationship among SPNDEN and SPND Settings and Next-Access Delay Values

SPNDEN	SPND	Next-Access Delay Value
0	000 to 111	1 RSPCK + 2 P1φ
1	000	1 RSPCK + 2 P1φ
	001	2 RSPCK + 2 P1φ
	010	3 RSPCK + 2 P1φ
	011	4 RSPCK + 2 P1φ
	100	5 RSPCK + 2 P1φ
	101	6 RSPCK + 2 P1φ
	110	7 RSPCK + 2 P1φ
	111	8 RSPCK + 2 P1φ

(h) Initialization Flowchart

Figure 19.17 is a flowchart illustrating an example of initialization in SPI operation when this module is used in master mode. For a description of how to set up the interrupt controller, direct memory access controller, and input/output ports, see the descriptions given in the individual blocks.

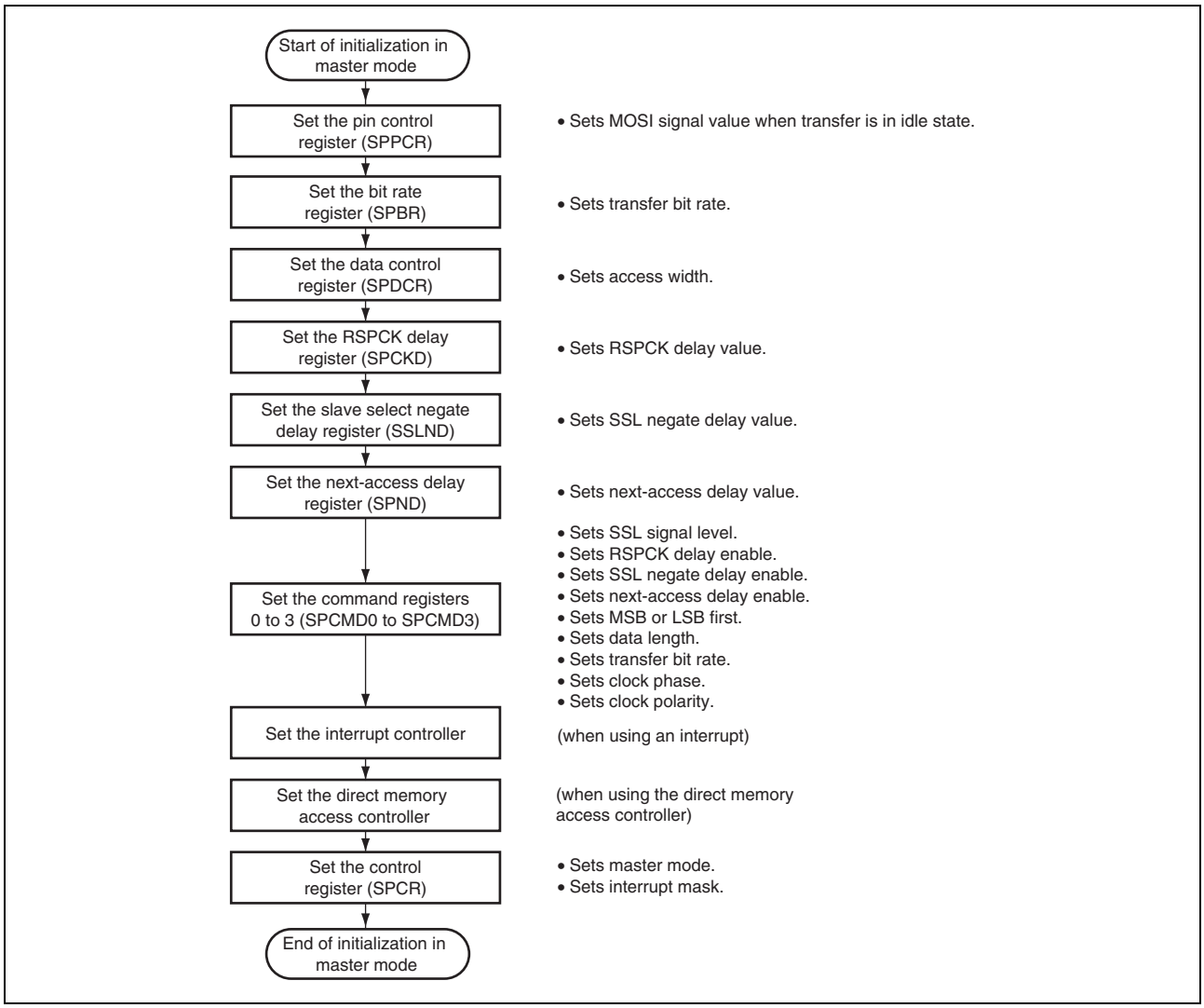


Figure 19.17 Example of Initialization Flowchart in Master Mode

(i) Transfer Operation Flowchart

Figure 19.18 is a flowchart illustrating a transfer in SPI operation when this module is used in master mode.

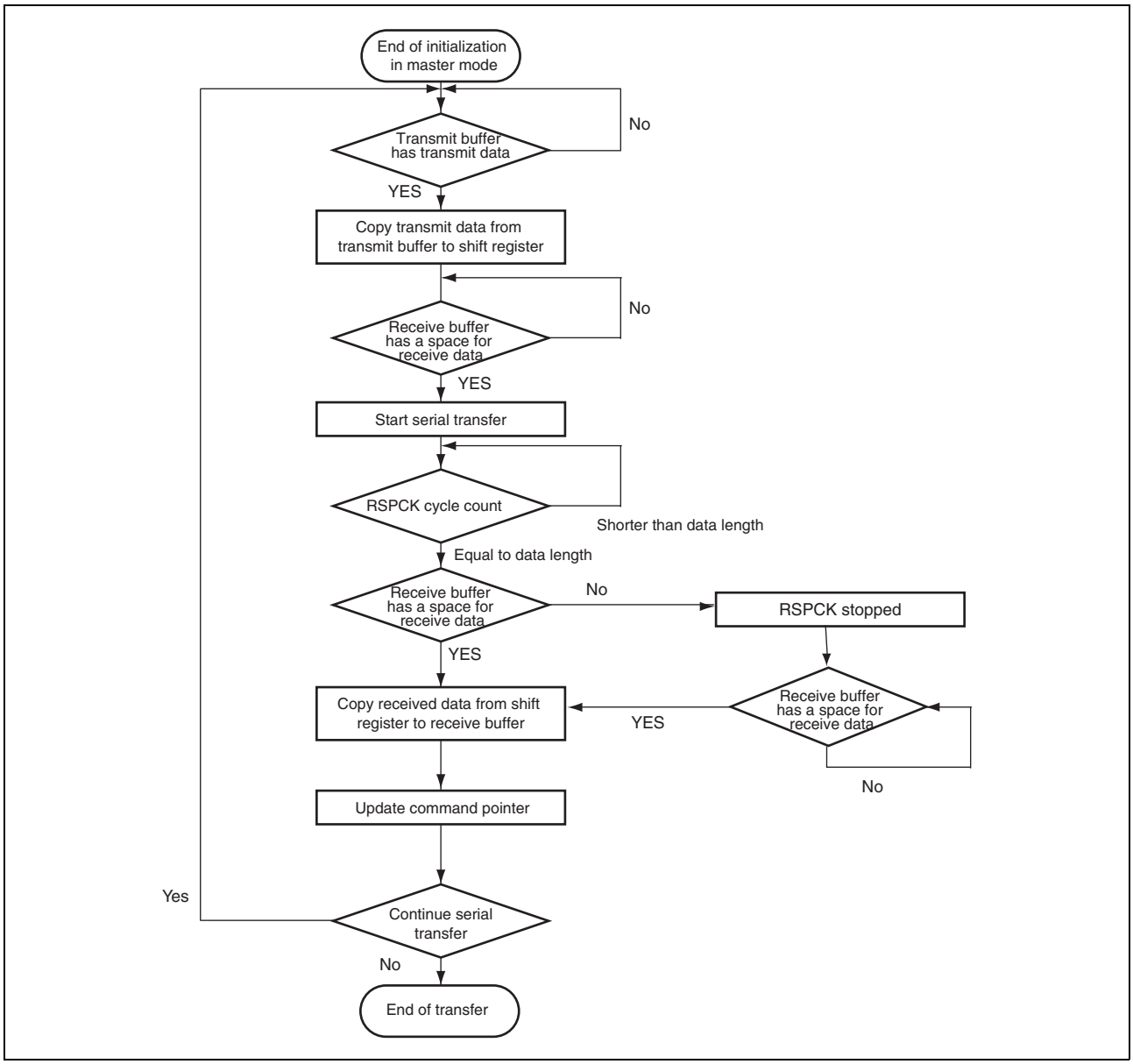


Figure 19.18 Transfer Operation Flowchart in Master Mode

(2) Slave Mode Operation

(a) Starting Serial Transfer

If this module detects an SSL input signal assertion when the CPHA bit in the command register 0 (SPCMD0) is 0, this module is required to start driving valid data to the MISO output signal. For this reason, when the CPHA bit is 0, the asserting of the SSL input signal triggers the start of a serial transfer.
If this module detects the first RSPCK edge in an SSL signal asserted condition when the CPHA bit is 1, this module is required to start driving valid data to the MISO output signal. For this reason, when the CPHA bit is 1, the first RSPCK edge in an SSL signal asserted condition triggers the start of a serial transfer.
When detecting the start of a serial transfer in a condition in which the shift register is empty, this module changes the status of the shift register to "full", so that data cannot be copied from the transmit buffer to the shift register when serial transfer is in progress. If the shift register was full before the serial transfer started, this module leaves the status of the shift register intact, in the full state.
Irrespective of CPHA bit settings, this module starts driving MISO output signals at the SSL signal assertion timing. Whether the data output from this module is valid or invalid differs depending on CPHA bit settings.
For details on the transfer format, see section 19.4.4, Transfer Format. The polarity of the SSL input signal depends on the setting of the SSL0P bit in the slave select polarity register (SSLP).

(b) Terminating Serial Transfer

Irrespective of the CPHA bit in the command register 0 (SPCMD0), this module terminates the serial transfer after detecting an RSPCK edge corresponding to the final sampling timing. When the receive buffer has an enough space for receive data, this module copies received data from the shift register to the receive buffer of the data register (SPDR) upon termination of the serial transfer. Irrespective of the value of the SPRF bit, this module changes the status of the shift register to "empty" upon termination of the serial transfer. If this module detects an SSL input signal negation from the beginning of serial transfer to the end of serial transfer, a mode fault error occurs (see section 19.4.6, Error Detection).
The final sampling timing changes depending on the bit length of the transfer data. In slave mode, the data length depends on the settings in bits SPB3 to SPB0 bits in SPCMD0. The polarity of the SSL input signal depends on the setting in the SSL0P bit in the slave select polarity register (SSLP). For details on the transfer format, see section 19.4.4, Transfer Format.

(c) Notes on Slave Operations

If the CPHA bit in the command register 0(SPCMD0) is 0, this module starts serial transfers when it detects the assertion edge for an SSL input signal. In the type of configuration shown in Figure 19.4 as an example, if this module is used in single-slave mode, the SSL signal is always fixed at active state. Therefore, when the CPHA bit is set to 0, this module cannot correctly start a serial transfer. To correctly execute send/receive operation in a configuration in which the SSL input signal is fixed at active state, the CPHA bit should be set to 1. When it is necessary to set the CPHA bit to 0, the SSL input signal should not be fixed.

(d) Burst Transfer

If the CPHA bit in the command register 0 (SPCMD0) is 1, continuous serial transfer (burst transfer) can be executed while retaining the assertion state for the SSL input signal. If the CPHA bit is 1, the period from the first RSPCK edge to the sampling timing for the reception of the final bit in an SSL signal active state corresponds to a serial transfer period. Even when the SSL input signal remains at the active level, this module can accommodate burst transfers because it can detect the start of access.
If the CPHA bit is 0, for the reason given in section 19.4.8, (2), (c) Notes on Slave Operations, second and subsequent serial transfers during the burst transfer cannot be executed correctly.

(e) Initialization Flowchart

Figure 19.19 is a flowchart illustrating an example of initialization in SPI operation when this module is used in slave mode. For a description of how to set up the interrupt controller, direct memory access controller, and input/output ports, see the descriptions given in the individual blocks.

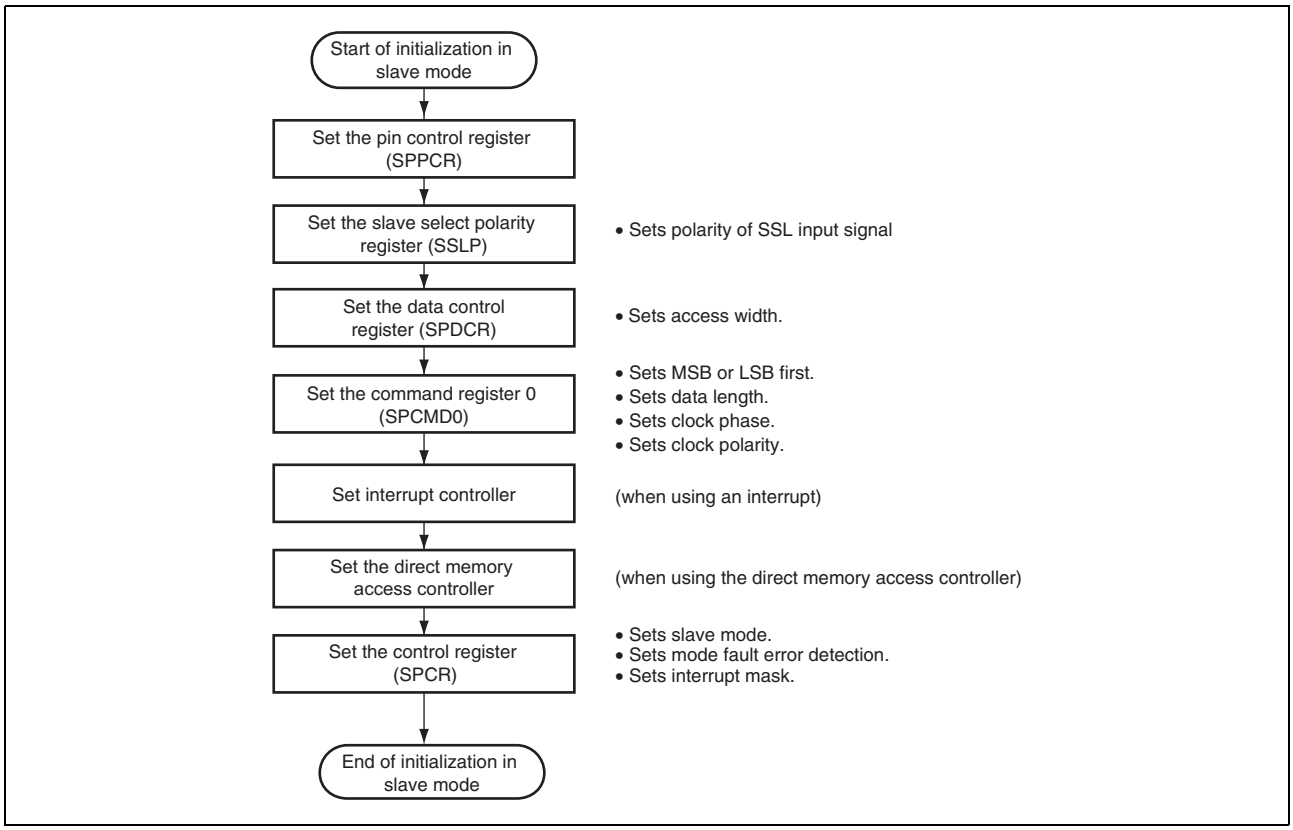


Figure 19.19 Example of Initialization Flowchart in Slave Mode

(f) Transfer Operation Flowchart (CPHA = 0)

Figure 19.20 is a flowchart illustrating a transfer in SPI operation when this module is used in slave mode with the CPHA bit in the command register 0 (SPCMD0) set to 0.

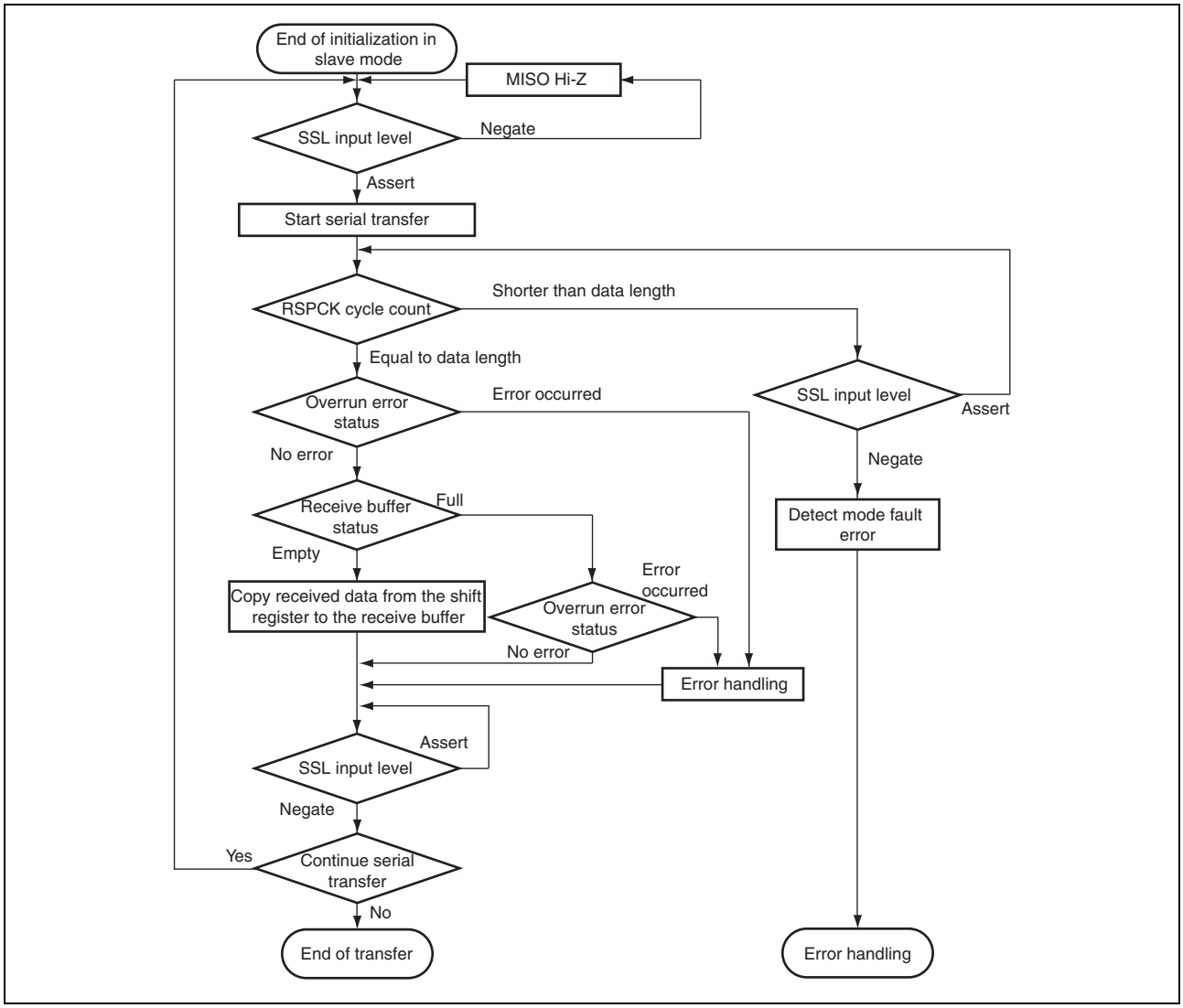


Figure 19.20 Transfer Operation Flowchart in Slave (CPHA = 0)

(g) Transfer Operation Flowchart (CPHA = 1)

Figure 19.21 is a flowchart illustrating a transfer in SPI operation when this module is used in slave mode with the CPHA bit in the command register 0 (SPCMD0) and the MODFEN bit in the control register (SPCR) set to 1, respectively. The subsequent operation is not guaranteed when the serial transfer is started with the MODFEN bit set to 0 and the SSL input level is negated with the number of RSPCK cycles shorter than the data length.

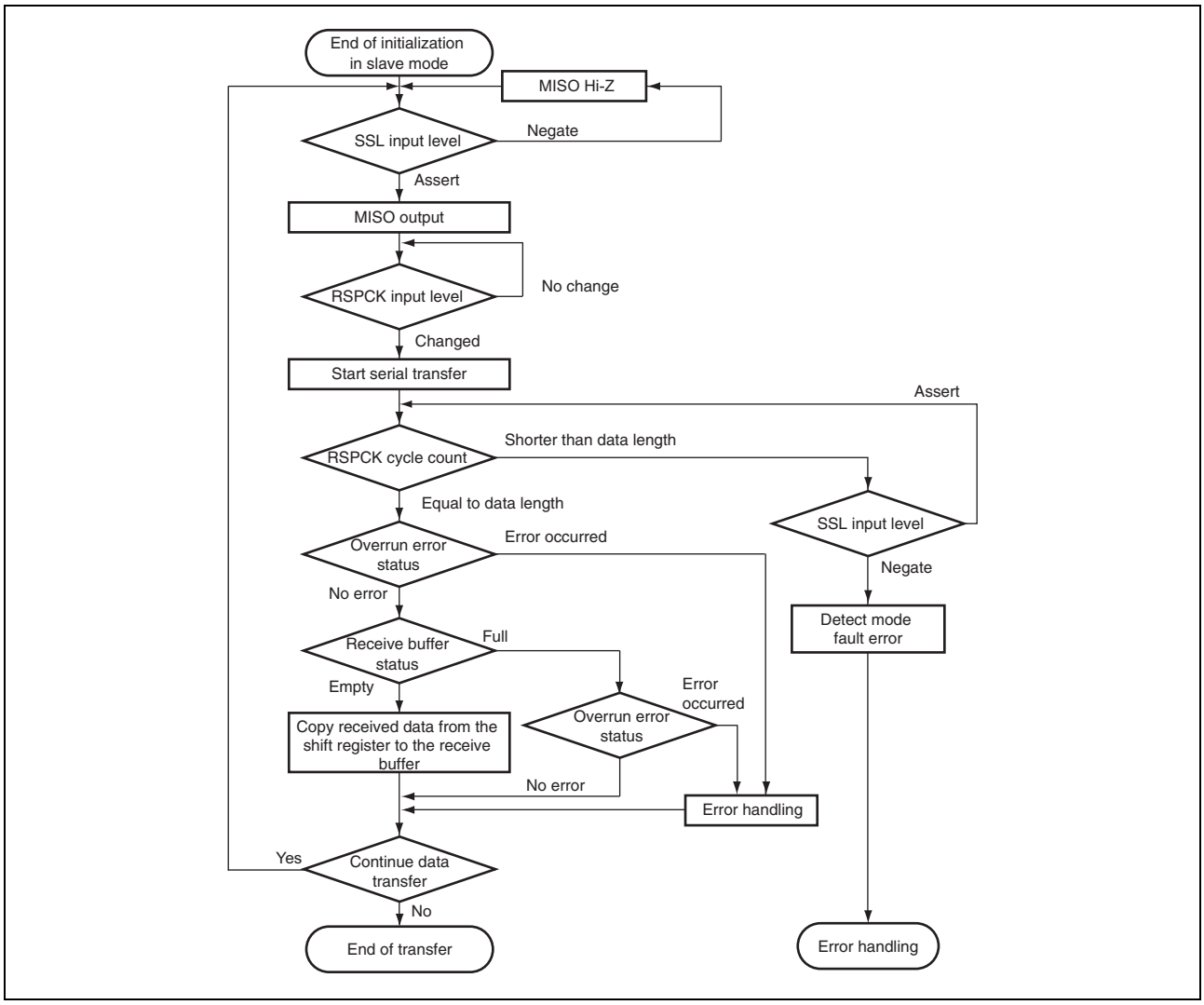


Figure 19.21 Transfer Operation Flowchart in Slave Mode (CPHA = 1)

19.4.9 Error Handling

Figure 19.22 and Figure 19.23 show the error handling. The following error handling is used to return from the error state after an error in master or slave mode.

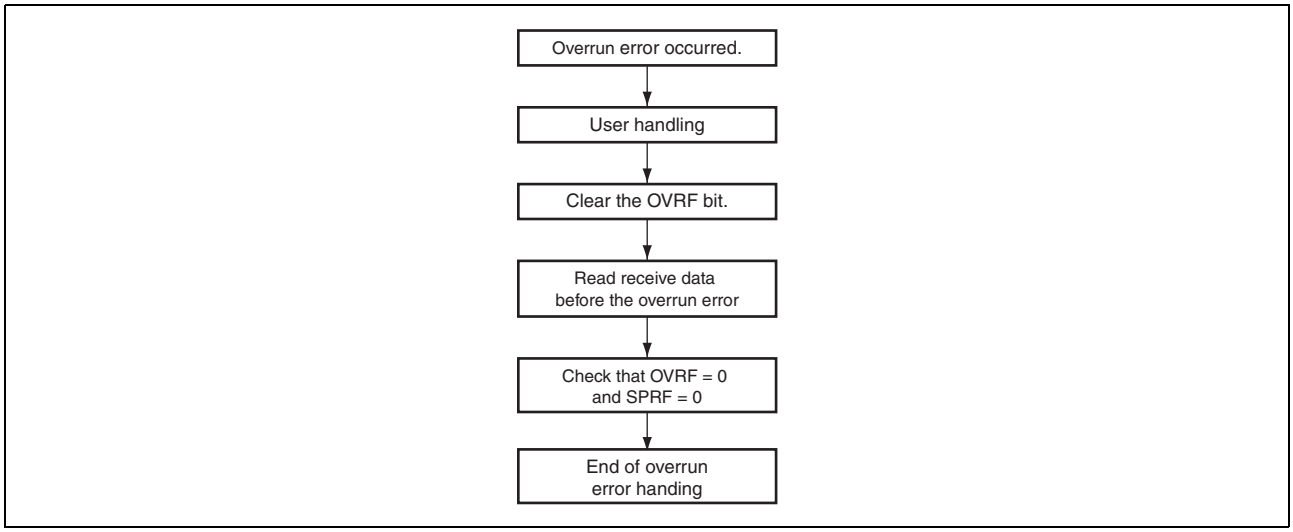


Figure 19.22 Error Handling (Overrun Error)

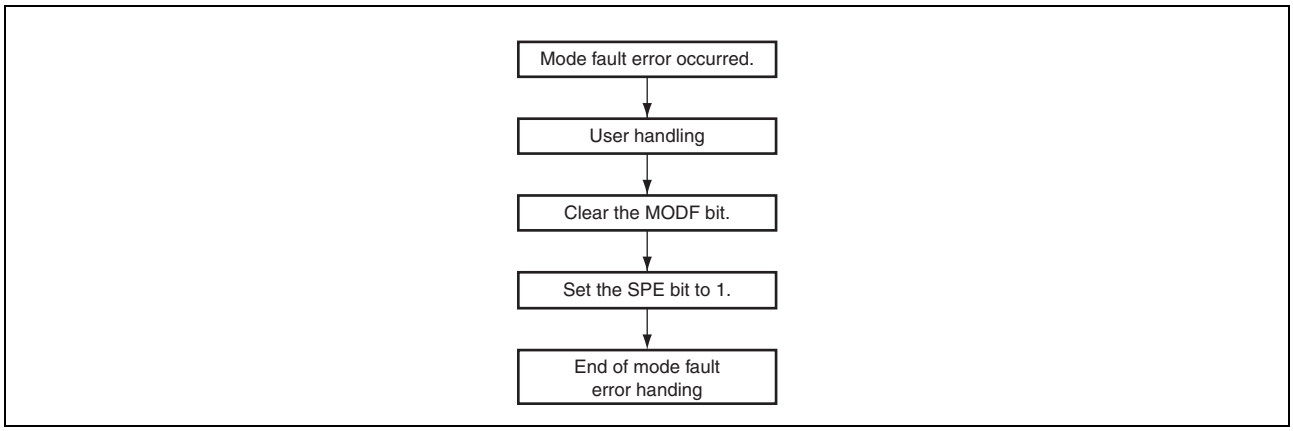


Figure 19.23 Error Handling (Mode Fault Error)

19.4.10 Loopback Mode

When 1 is written to the SPLP bit in the pin control register (SPPCR), this module shuts off the path between the MISO pin and the shift register, and between the MOSI pin and the shift register, and connects the input path and the output path (reversed) of the shift register. This is called loopback mode. When a serial transfer is executed in loopback mode, the transmit data becomes the received data. Figure 19.24 shows the configuration of the shift register input/output paths for the case where this module in master mode is set in loopback mode.

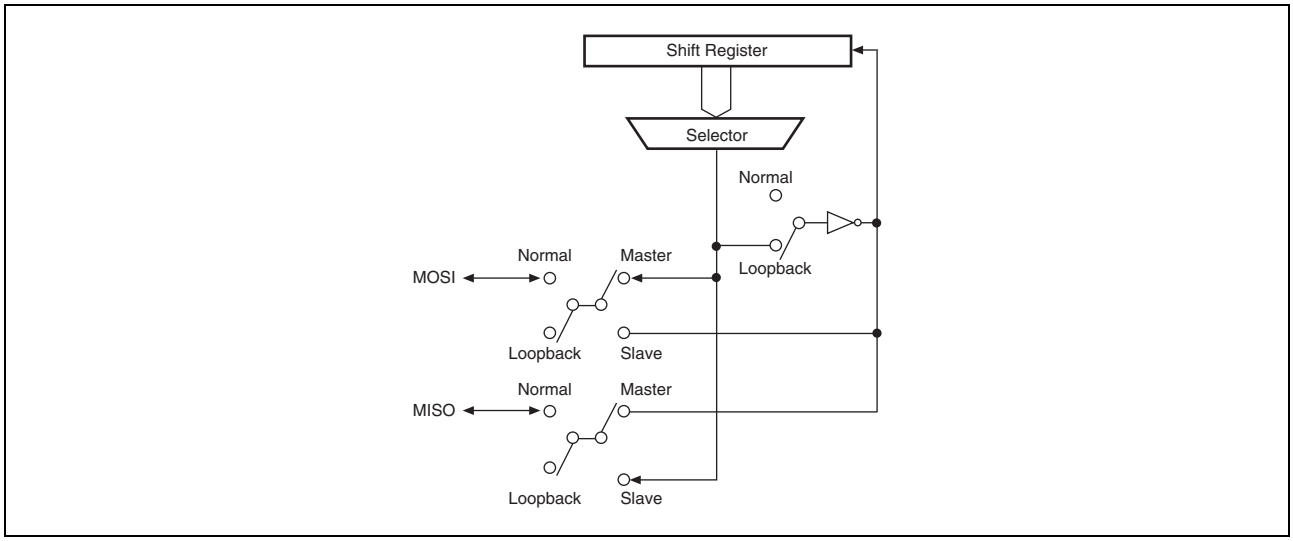


Figure 19.24 Configuration of Shift Register Input/Output Paths in Loopback Mode (Master Mode)

19.4.11 Interrupt Sources

This module has interrupt sources of receive buffer full, transmit buffer empty, mode fault, and overrun. In addition, the direct memory access controller can be activated by the receive buffer full or transmit buffer empty interrupt for data transfer.

Table 19.11 shows the interrupt sources.

When any of the interrupt conditions in Table 19.11 is met, an interrupt is generated. The interrupt sources should be cleared with data transfer by the CPU or direct memory access controller.

Table 19.11 Interrupt Sources

Name	Interrupt Source	Abbreviation	Interrupt Condition	Activation of Direct Memory Access Controller
SPRI	Receive buffer full	RXI	(SPRIE = 1) • (SPRF = 1)	Possible
SPTI	Transmit buffer empty	TXI	(SPTIE = 1) • (SPTEF = 1)	Possible
SPEI	Mode fault	MOI	(SPEIE = 1) • (MODF = 1)	—
	Overrun	OVI	(SPEIE = 1) • (OVRF = 1)	—

20. SPI Multi I/O Bus Controller

The SPI multi I/O bus controller enables the direct connection of serial flash, OctaFlash™, Xccela™ flash, or HyperFlash™ memory devices to this LSI chip.

20.1 Features

This module allows the connected serial flash, OctaFlash™, Xccela™ flash, or HyperFlash™ memory devices to be accessed by reading the external address space, or using Manual mode to transmit and receive data.

20.1.1 Serial Flash Memory Interface

- PVcc_SPI: 3.3V
- QSPI0_SPCLK: 66MHz
- Up to two serial flash memory devices are connectable per channel.
  Note: Two serial flash memory devices are connected in parallel to implement an 8-bit bus interface.
- A data bus size of 1 bit or 4 bits can be selected for one serial flash memory device.
- Both single data rate (SDR) and double data rate (DDR) transfers are supported.
- Supports master operation. (Slave operation is not supported.)

20.1.2 OctaFlash™, Xccela™ Flash Memory Interface

- PVcc_SPI: 1.8V
- QSPI0_SPCLK: 132MHz
- One Octal-SPI flash memory device is connectable.

Note: OctaFlash™ is a trademark of Macronix International Co., Ltd.
Note: Xccela™ flash memory is a trademark of Micron Technology, Inc.
Note: In this manual, an 8-bit SPI flash memory with 1 chip select, 1 clock source, and 1 data strobe configuration is called "Octal-SPI flash memory".
Note: In this manual, transfer with an 8-bit data bus width is referred to as Octal-SPI flash memory protocol mode. Transfer with a 1-bit data bus width is handled in the same way as for standard serial flash memory.

20.1.3 HyperFlash™ Interface

- PVcc_SPI: 1.8V
- QSPI0_SPCLK/QSPI1_SPCLK: 132MHz
- One HyperFlash memory device is connectable.
- The data bus is fixed to 8-bit width.
- Wrapped burst operation is not supported.

Note: HyperFlash is a trademark of Cypress Semiconductor Corporation.

20.1.4 External Address Space Read Mode

- Read access from the bus master to the SPI multi I/O space is automatically converted to a read command, and the read data is returned to the bus master.
- Normal read operation and burst read operation
- Efficient data reception due to built-in read cache (64-bit line × 32 entries)

20.1.5 Manual Mode

- Read and write commands are available for serial flash memory, Octal-SPI flash memory, or HyperFlash memory.
- A write buffer is provided to improve the efficiency of data write operations.
- The cache is not usable for reading.

20.2 Block Diagram

Figure 20.1 shows a block diagram of this module.

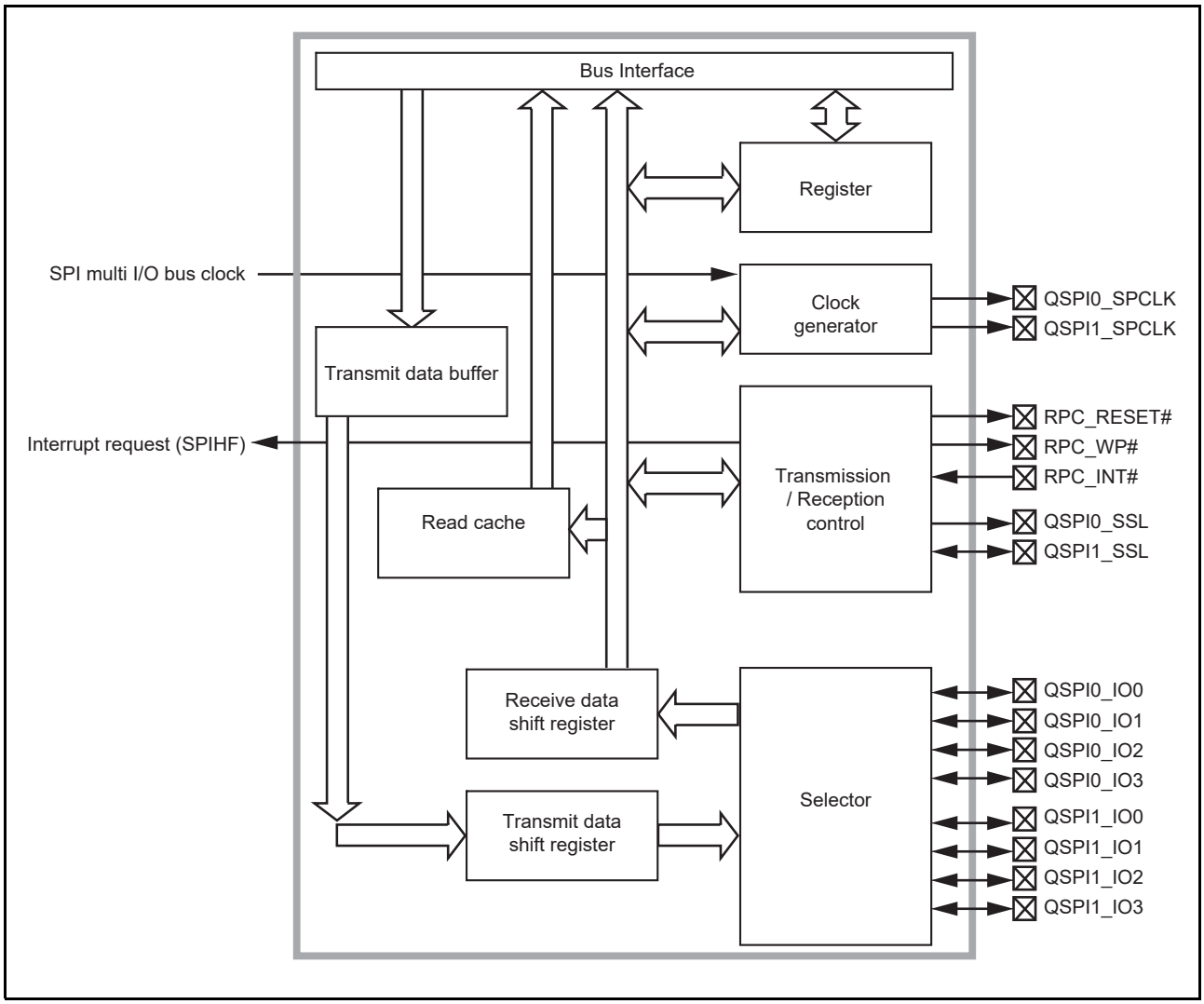


Figure 20.1 Block Diagram

20.3 Input/Output Pins

Table 20.1 shows the pin configuration for one channel.
The operating voltage on pins of the SPI multi I/O bus controller is 3.3 V or 1.8 V.
Before using the pins of the SPI multi I/O bus controller, be sure to set up the dedicated pin POC control register (PPOC) and SPI multi I/O bus controller dedicated pin driving ability control register (PSPIBSC).
For details, see section 51.3.30, Dedicated Pin POC Control Register (PPOC) and section 51.3.34, SPI Multi I/O Bus Controller Dedicated Pin Driving Ability Control Register (PSPIBSC).

Table 20.1 Pin Configuration

Pin Name	Symbol	I/O	One 4-bit Serial Flash connected. CMNCR.BSZ = 00	Two 4-bit Serial Flash connected. CMNCR.BSZ = 01	Connection in Octal-SPI flash memory protocol mode CMNCR.BSZ = 01	HyperFlash connected. CMNCR.BSZ = 01
Clock*1	QSPI0_SPCLK	Output	SCK	SCK	SCLK	CK
Data 0*1	QSPI0_IO0	I/O	SI/IO0	SI/IO0	SI/SIO0	DQ0
Data 1*1	QSPI0_IO1	I/O	SO/IO1	SO/IO1	SO/SIO1	DQ1
Data 2*1	QSPI0_IO2	I/O	WP#/IO2	WP#/IO2	SIO2	DQ2
Data 3*1	QSPI0_IO3	I/O	HOLD#/IO3	HOLD#/IO3	SIO3	DQ3
Slave select*1	QSPI0_SSL	Output	CS#	CS#	CS#	CS#
Clock	QSPI1_SPCLK	Output	-	SCK	-	CK#
Data 4	QSPI1_IO0	I/O	-	SI/IO0	SIO4	DQ4
Data 5	QSPI1_IO1	I/O	-	SO/IO1	SIO5	DQ5
Data 6	QSPI1_IO2	I/O	-	WP#/IO2	SIO6	DQ6
Data 7	QSPI1_IO3	I/O	-	HOLD#/IO3	SIO7	DQ7
Slave select	QSPI1_SSL	I/O	-	CS#	DQS (Data Strobe)	RWDS (Read Data Strobe)
Reset*2	RPC_RESET#	Output	RESET#	RESET#	RESET#	RESET#
Write protect*2	RPC_WP#	Output	For one 1-bit serial flash memory: WP#/IO2	-	-	-
Interrupt*2	RPC_INT#	Input	-	-	-	INT#

Note 1. When connecting a serial flash memory device with the BSZ[1:0] bits set to 00, connect the serial flash memory to the QSPI0 pin group of this LSI.
Note 2. These pin connections must meet to the flash specifications.

20.4 Register Descriptions

Table 20.2 shows the register configuration.

Table 20.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Common control register	CMNCR	R/W	H'01557301	H'1F800000	32
SSL delay register	SSLDR	R/W	H'00000000	H'1F800004	32
Data read control register	DRCR	R/W	H'001F0100	H'1F80000C	32
Data read command setting register	DRCMR	R/W	H'00A00000	H'1F800010	32
Data read extended address setting register	DREAR	R/W	H'00000000	H'1F800014	32
Data read option setting register	DROPR	R/W	H'00000000	H'1F800018	32
Data read enable setting register	DRENR	R/W	H'A222D400	H'1F80001C	32
Manual mode control register	SMCR	R/W	H'00000000	H'1F800020	32
Manual mode command setting register	SMCMR	R/W	H'00000000	H'1F800024	32
Manual mode address setting register	SMADR	R/W	H'00000000	H'1F800028	32
Manual mode option setting register	SMOPR	R/W	H'00000000	H'1F80002C	32
Manual mode enable setting register	SMENR	R/W	H'00004000	H'1F800030	32
Manual mode read data register 0	SMRDR0	R	Undefined	H'1F800038	8, 16, 32
Manual mode read data register 1	SMRDR1	R	Undefined	H'1F80003C	8, 16, 32
Manual mode write data register 0	SMWDR0	R/W	H'00000000	H'1F800040	8, 16, 32
Manual mode write data register 1	SMWDR1	R/W	H'00000000	H'1F800044	8, 16, 32
Common status register	CMNSR	R	H'00000001	H'1F800048	32
Data read dummy cycle setting register	DRDMCR	R/W	H'0000000B	H'1F800058	32
Data read DDR enable register	DRDRENR	R/W	H'00005101	H'1F80005C	32
Manual mode dummy cycle setting register	SMDMCR	R/W	H'00000000	H'1F800060	32
Manual mode DDR enable register	SMDRENR	R/W	H'00000000	H'1F800064	32
PHY control register	PHYCNT	R/W	H'00000263	H'1F80007C	32
PHY offset register 1	PHYOFFSET1	R/W	H'21511144	H'1F800080	32
PHY offset register 2	PHYOFFSET2	R/W	H'00000431	H'1F800084	32
PHY interrupt register	PHYINT	R/W	H'07070002	H'1F800088	32
PHY adjustment register 1	PHYADJ1	R/W	H'00000000	H'1F800070	32
PHY adjustment register 2	PHYADJ2	R/W	H'00000000	H'1F800074	32

Note: Do not write to any address those listed in the table. Otherwise, correct operation cannot be guaranteed. Reading from a non-listed address returns an undefined value.

20.4.1 Common Control Register (CMNCR)

CMNCR is a 32-bit register that controls the SPI multi I/O bus controller. The settings of this register are reflected both in external address space read mode and manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MD	—	—	—	—	—	—	—	MOIIO3[1:0]		MOIIO2[1:0]		MOIIO1[1:0]		MOIIO0[1:0]	
Initial Value:	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1
R/W:	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IO3FV[1:0]		IO2FV[1:0]		—	—	IO0FV[1:0]		—	—	—	—	—	—	BSZ[1:0]	
Initial Value:	0	1	1	1	0	0	1	1	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MD	0	R/W	Operating Mode Switch Switches the operating modes. 0: External address space read mode 1: Manual mode
30 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
23, 22	MOIIO3[1:0]	01	R/W	QSPIn_SSL Output Idle Value Fix QSPIn_IO3 Fixes the output value on QSPIn_IO3 while QSPIn_SSL is negated. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state. Note 1. When a connection is to be made in Octal-SPI flash memory protocol mode, set these bits to 01. Note 2. If a HyperFlash memory device is connected, be sure to set these bits to 10 (the output value being that of the last bit in the previous transfer, with the pin placed in the Hi-Z state if that was the case in the previous transfer).
21, 20	MOIIO2[1:0]	01	R/W	QSPIn_SSL Output Idle Value Fix QSPIn_IO2 Fixes the output value on QSPIn_IO2 while QSPIn_SSL is negated. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state. Note 1. When a connection is to be made in Octal-SPI flash memory protocol mode, set these bits to 01. Note 2. If a HyperFlash memory device is connected, be sure to set these bits to 10 (the output value being that of the last bit in the previous transfer, with the pin placed in the Hi-Z state if that was the case in the previous transfer).

Bit	Bit Name	Initial Value	R/W	Description
19, 18	MOIIO1[1:0]	01	R/W	QSPIn_SSL Output Idle Value Fix QSPIn_IO1 Fixes the output value on QSPIn_IO1 while QSPIn_SSL is negated. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state. Note 1. When a connection is to be made in Octal-SPI flash memory protocol mode, set these bits to 01. Note 2. If a HyperFlash memory device is connected, be sure to set these bits to 10 (the output value being that of the last bit in the previous transfer, with the pin placed in the Hi-Z state if that was the case in the previous transfer).
17, 16	MOIIO0[1:0]	01	R/W	QSPIn_SSL Output Idle Value Fix QSPIn_IO0 Fixes the output value on QSPIn_IO0 while QSPIn_SSL is negated. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state. Note 1. When a connection is to be made in Octal-SPI flash memory protocol mode, set these bits to 01. Note 2. If a HyperFlash memory device is connected, be sure to set these bits to 10 (the output value being that of the last bit in the previous transfer, with the pin placed in the Hi-Z state if that was the case in the previous transfer).
15, 14	IO3FV[1:0]	01	R/W	QSPIn_IO3 Fixed Value for 1-bit Size Fixes the output value of QSPIn_IO3 pin for 1-bit size. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state.
13, 12	IO2FV[1:0]	11	R/W	QSPIn_IO2 Fixed Value for 1-bit Size Fixes the output value of QSPIn_IO2 pin for 1-bit size. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state.
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	IO0FV[1:0]	11	R/W	QSPIn_IO0 Fixed Value for 1-bit Size Fixes the output value of QSPIn_IO0 pin for 1-bit size. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state. Note: Set these bits to 11 (the pin is placed in the Hi-Z state) when data read transfer size is not 1-bit.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	BSZ[1:0]	01	R/W	Data Bus Size Specifies the number of serial flash memories to be connected. When HyperFlash connected, set to 01. 00: one serial flash memory connected 01: Two serial flash memory devices are connected, a single HyperFlash memory device is connected, or a single device is connected in Octal-SPI flash memory protocol mode. 1X: Setting prohibited Note: After changing (the value of) this bit field, all the entries in the read cache must be cleared by setting the RCF bit in DRCR to 1.

Note: The settings of the MOIIO3, MOIIO2, MOIIO1, and MOIIO0 bits for control over fixing the output values are reflected at the time QSPIn_SSL is negated on completion of data transfer.

20.4.2 SSL Delay Register (SSLDR)

SSLDR is a 32-bit register that adjusts the timing between the QSPIn_SSL signal and the QSPIn_SPCLK signal.
The settings of this register are reflected both in external address space read mode and manual mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	SPNDL[2:0]		
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLNDL[2:0]			—	—	—	—	—	SCKDL[2:0]		
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	SPNDL[2:0]	000	R/W	Next Access Delay Sets the period from transfer end to next transfer start (next access). 000: 1 QSPIn_SPCLK (3 QSPIn_SPCLK when SSLE = 0) 001: 2 QSPIn_SPCLK 010: 3 QSPIn_SPCLK 011: 4 QSPIn_SPCLK 100: 5 QSPIn_SPCLK 101: 6 QSPIn_SPCLK 110: 7 QSPIn_SPCLK 111: 8 QSPIn_SPCLK
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	SLNDL[2:0]	000	R/W	QSPIn_SSL Negation Delay Sets the period from the time the last QSPIn_SPCLK edge is sent of a transfer to QSPIn_SSL pin negation (QSPIn_SSL negation delay). 000: 1 QSPIn_SPCLK 001: 2 QSPIn_SPCLK 010: 3 QSPIn_SPCLK 011: 4 QSPIn_SPCLK 100: 5 QSPIn_SPCLK 101: 6 QSPIn_SPCLK 110: 7 QSPIn_SPCLK 111: 8 QSPIn_SPCLK
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	SCKDL[2:0]	000	R/W	Clock Delay Sets the period from QSPIn_SSL pin assertion to QSPIn_SPCLK oscillation (clock delay). 000: 1.5 QSPIn_SPCLK 001: 2.5 QSPIn_SPCLK 010: 3.5 QSPIn_SPCLK 011: 4.5 QSPIn_SPCLK 100: 5.5 QSPIn_SPCLK 101: 6.5 QSPIn_SPCLK 110: 7.5 QSPIn_SPCLK 111: 8.5 QSPIn_SPCLK

20.4.3 Data Read Control Register (DRCR)

DRCR is a 32-bit register that sets the operation in external address space read mode.
The bits except the SSLN bit should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	SSLN	—	—	—	RBURST[4:0]				
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RCF	RBE	—	—	—	—	—	—	—	SSLE
Initial Value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	SSLN	0	W	QSPIn_SSL Negation Asserted QSPIn_SSL can be negated by writing 1 to this bit when both the RBE and SSLE bits are 1. This bit is always read as 0. Note: To start next access after QSPIn_SSL negation using this bit, read SSLF in CMNSR = 0 to confirm that the QSPIn_SSL has been negated.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	RBURST [4:0]	11111	R/W	Read Data Burst Length Sets the burst length (data unit count) when reading. This bit field is enabled when the RBE bit is set to 1. 00000: 1 data unit 00001: 2 continuous data units : 11110: 31 continuous data units 11111: 32 continuous data units One data unit is 64 bits long.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	RCF	0	W	Read Cache Flush When 1 is written to this bit, all the entries in the read cache are cleared. This bit is always read as 0. After using the cache area as the write buffer, the cache area must be cleared by writing 1 to the RCF bit. Note: After flushing the read cache by writing 1 to the RCF bit, read the DRCR register before proceeding to read from the external address space.
8	RBE	1	R/W	Read Burst Turns burst ON or OFF when reading. 0: Data is read according to the access size. 1: Read cache is enabled, and as many data units as the burst count specified in RBURST[4:0] bits is read. Note: When the cache is enabled and read access reaches the last address of the flash memory, the access address does not match the address held in the cache. Control the read address and access size so that read access does not reach the last address of the flash memory.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	SSLE	0	R/W	QSPIn_SSL Negation Sets the conditions for QSPIn_SSL negation during read burst. QSPIn_SSL is negated for each access during normal read. 0: QSPIn_SSL is negated after transfer of data set in burst length. 1: QSPIn_SSL is negated when the accessed address is not continuous with the previously transferred address. Note 1. Set up this bit when a serial flash memory device is connected. Note 2. When this bit is set to 1, setting the RBURST[4:0] bits to 5'h01 is prohibited.

20.4.4 Data Read Command Setting Register (DRCMR)

DRCMR is a 32-bit register that sets the commands issued in external address space read mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CMD[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	OCMD[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	CMD[7:0]	H'A0	R/W	Command Sets the command.
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	OCMD[7:0]	'H00	R/W	Optional Command Sets the optional command.

20.4.5 Data Read Extended Address Setting Register (DREAR)

DREAR is a 32-bit register that is set when addresses for serial flash memory or according to the Octal-SPI flash memory protocol are to be output in 32-bit mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

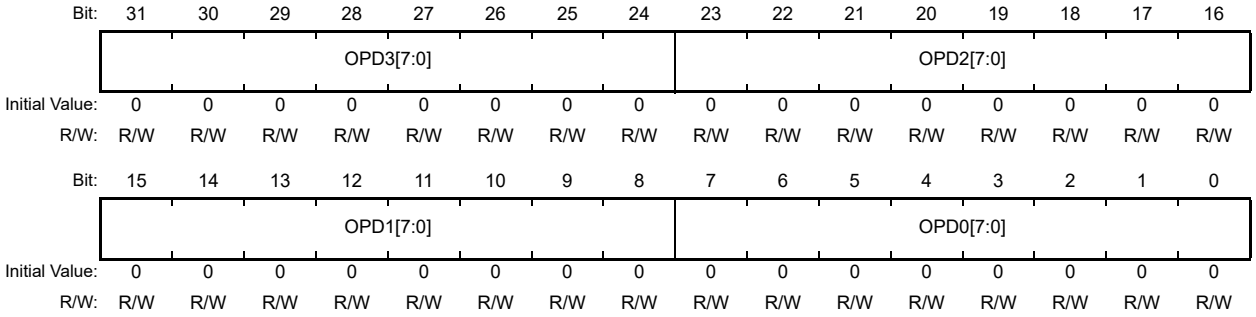
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	EAV[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	EAC[2:0]		
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	EAV[7:0]	H'00	R/W	32-Bit Extended Upper Address Fixed Value Sets the higher-order address bit values for external addresses as specified by the EAC[2:0] bits when addresses for serial flash memory or according to the Octal-SPI flash memory protocol are to be output in 32-bit mode. Bit 0 corresponds to address bit 25 of the serial flash memory or Octal-SPI flash memory, and bit 7 corresponds to bit 32. This setting is valid when the ADE[3] bit in DRENR is 1. When EAC[2:0] are 000, set a fixed value for address bits 32 to 25 of the serial flash memory or Octal-SPI flash memory in EAV[7:0]. When EAC[2:0] are 001, set a fixed value for address bits 32 to 26 of the serial flash memory or Octal-SPI flash memory in EAV[7:1]. When EAC[2:0] are 010, set a fixed value for address bits 32 to 27 of the serial flash memory or Octal-SPI flash memory in EAV[7:2]. When EAC[2:0] are 011, set a fixed value for address bits 32 to 28 of the serial flash memory or Octal-SPI flash memory in EAV[7:3]. (1) When a single serial flash memory device is connected or the Octal-SPI flash memory protocol is in use Address bits 31 to 0 of the serial flash memory are used in access. (2) When two serial flash memory devices are connected Address bits 32 to 1 of the serial flash memory are used in access.
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	EAC[2:0]	000	R/W	32-Bit Extended External Address Valid Range Sets the range of external addresses to be used as addresses of the serial flash memory or Octal-SPI flash memory when addresses for serial flash memory or according to the Octal-SPI flash memory protocol are to be output in 32-bit mode. This setting is valid when the ADE[3] bit in DRENR is 1. 000: External address bits [24:0] enabled 001: External address bits [25:0] enabled 010: External address bits [26:0] enabled 011: External address bits [27:0] enabled Other than above: Setting prohibited

20.4.6 Data Read Option Setting Register (DROPR)

DROPR is a 32-bit register that sets the option data in external address space read mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	OPD3[7:0]	H'00	R/W	Option Data 3 Sets the option data 3.
23 to 16	OPD2[7:0]	H'00	R/W	Option Data 2 Sets the option data 2.
15 to 8	OPD1[7:0]	H'00	R/W	Option Data 1 Sets the option data 1.
7 to 0	OPD0[7:0]	H'00	R/W	Option Data 0 Sets the option data 0.

Note: OPD3, OPD2, OPD1, and OPD0 are output in this order.

20.4.7 Data Read Enable Setting Register (DRENR)

DRENR is a 32-bit register that sets the bit size of the command, optional command, address, option data, and read data in external address space read mode and enables outputting them other than read data.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

When HyperFlash memory is connected or a connection is made in Octal-SPI flash memory protocol mode, refer to Table 20.6, Enable Register (HyperFlash), or Table 20.7, Enable Register (Octal-SPI Flash Memory Protocol Mode).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDB[1:0]		OCDB[1:0]		—		ADB[1:0]		—		OPDB[1:0]		—		DRDB[1:0]	
Initial Value:	1	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DME	CDE	—	OCDE	ADE[3:0]				OPDE[3:0]				—	—	—	—
Initial Value:	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	CDB[1:0]	10	R/W	Command Bit Size Sets the command size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
29, 28	OCDB[1:0]	10	R/W	Optional Command Bit Size Sets the optional command size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
27, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25, 24	ADB[1:0]	10	R/W	Address Bit Size Sets the address size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	OPDB[1:0]	10	R/W	Option Data Bit Size Sets the option data size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	DRDB[1:0]	10	R/W	Data Read Bit Size Sets the data read size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
15	DME	1	R/W	Dummy Cycle Enable Enables insertion of the dummy cycle before the read data. Note: A setting is prohibited for a transfer starting with a dummy cycle. 0: Dummy cycle insertion disabled 1: Dummy cycle insertion enabled
14	CDE	1	R/W	Command Enable Sets whether or not the command is to be output. 0: Command output disabled 1: Command output enabled Note: When an Octal-SPI flash memory device is connected and this bit set to 1, set the CDE bit in SMENR register to 0.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	OCDE	1	R/W	Optional Command Enable Sets whether or not the optional command is to be output. 0: Optional command output disabled 1: Optional command output enabled Note: When an Octal-SPI flash memory device is connected, set this bit to 0.
11 to 8	ADE[3:0]	0100	R/W	Address Enable Sets the address to be output. Be sure to use the following setting; otherwise, the operation is not guaranteed. (1) When a serial flash memory device is connected 0000: Output disabled 0111: Address[23:0] 1111: Address[31:0] Other than above: Setting prohibited (2) When two serial flash memory devices are connected or a connection is made in Octal-SPI flash memory protocol mode 0000: Output disabled 0111: Address[24:1] 1111: Address[32:1] 1100: Addresses in Octal-SPI flash memory protocol mode (operation is in accord with the 8-8-8 protocol in Table 20.15) Other than above: Setting prohibited (3) When a HyperFlash memory device is connected 0100: HyperFlash addresses Other than above: Setting prohibited
7 to 4	OPDE[3:0]	0000	R/W	Option Data Enable Sets the option data to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. 0000: Output disabled 1000: OPD3 1100: OPD3, OPD2 1110: OPD3, OPD2, OPD1 1111: OPD3, OPD2, OPD1, OPD0 Other than above: Setting prohibited
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

20.4.8 Manual Mode Control Register (SMCR)

SMCR is a 32-bit register that sets the operation in Manual mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SSLKP	—	—	—	—	—	SPIRE	SPIWE	SPIE
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R/W	R/W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SSLKP	0	R/W	QSPIn_SSL Signal Level Determines the QSPIn_SSL status after the end of transfer. 0: QSPIn_SSL signal is negated at the end of transfer. 1: QSPIn_SSL signal level is maintained from the end of transfer to the start of next access. Note: The setting to start data transfer for reading is prohibited while the setting of SSLKP is 1.
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	SPIRE	0	R/W	Data Read Enable Enables reading in manual mode. 0: Data reading disabled 1: Data reading enabled Note: When the transfer data bit size is set to 4 bits with the SPIDB[1:0] bits, the SPIRE and SPIWE bits should not be set to 1 at the same time.
1	SPIWE	0	R/W	Data Write Enable Enables writing in manual mode. 0: Data writing disabled 1: Data writing enabled Note: When the transfer data bit size is set to 4 bits with the SPIDB[1:0] bits, the SPIRE and SPIWE bits should not be set to 1 at the same time.
0	SPIE	0	W	SPI Data Transfer Enable Data is transferred by setting this bit to 1. This bit is enabled only when the TEND bit in CMNSR is set to 1. The operation cannot be guaranteed when this bit is set to 1 with the TEND bit set to 0. This bit is always read as 0. Note: When the QSPIn_SSL pin is de-asserted, the command, optional command, address, and option data that are output enabled are output even if the SPIRE and SPIWE bits are set to 0. When the QSPIn_SSL pin is asserted, follow the notes described in section 20.6.2, Notes on Starting Transfer from the QSPIn_SSL Retained State in Manual Mode.

20.4.9 Manual Mode Command Setting Register (SMCMR)

SMCMR is a 32-bit register that sets the commands issued in Manual mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CMD[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	OCMD[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	CMD[7:0]	H'00	R/W	Command Sets the command.
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	OCMD[7:0]	H'00	R/W	Optional Command Sets the optional command.

20.4.10 Manual Mode Address Setting Register (SMADR)

SMADR is a 32-bit register that sets the addresses in Manual mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADR[31:24]								ADR[23:16]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR[15:0]															
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ADR[31:24]	H'00	R/W	Address Sets the value of bits 31 to 24 when the serial flash address is output in 32-bit units. If a serial flash memory or Octal-SPI flash memory is connected, the setting of these bits is only valid when the ADE[3] bit in the SMENR register is 1. If HyperFlash memory is connected, the setting of these bits is valid regardless of the setting of the ADE[3] bit.
23 to 0	ADR[23:0]	H'000000	R/W	Address Sets the address.

20.4.11 Manual Mode Option Setting Register (SMOPR)

SMOPR is a 32-bit register that sets the option data in Manual mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OPD3[7:0]								OPD2[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OPD1[7:0]								OPD0[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	OPD3[7:0]	H'00	R/W	Option Data 3 Sets the option data 3.
23 to 16	OPD2[7:0]	H'00	R/W	Option Data 2 Sets the option data 2.
15 to 8	OPD1[7:0]	H'00	R/W	Option Data 1 Sets the option data 1.
7 to 0	OPD0[7:0]	H'00	R/W	Option Data 0 Sets the option data 0.

Note: OPD3, OPD2, OPD1, and OPD0 are output in this order.

20.4.12 Manual Mode Enable Setting Register (SMENR)

SMENR is a 32-bit register that sets the bit size of the command, optional command, address, option data, and transfer data in Manual mode and enables their output. SMENR also enables dummy cycle insertion. Disabling all of the command, optional command, address, option data, dummy cycle, and transfer data is prohibited. At least one of them except dummy cycle must be enabled.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

When HyperFlash memory is connected or a connection is made in Octal-SPI flash memory protocol mode, refer to Table 20.6, Enable Register (HyperFlash), or Table 20.7, Enable Register (Octal-SPI Flash Memory Protocol Mode).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDB[1:0]		OCDB[1:0]		—	—	ADB[1:0]		—	—	OPDB[1:0]		—	—	SPIDB[1:0]	
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DME	CDE	—	OCDE	ADE[3:0]				OPDE[3:0]				SPIDE[3:0]			
Initial Value:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

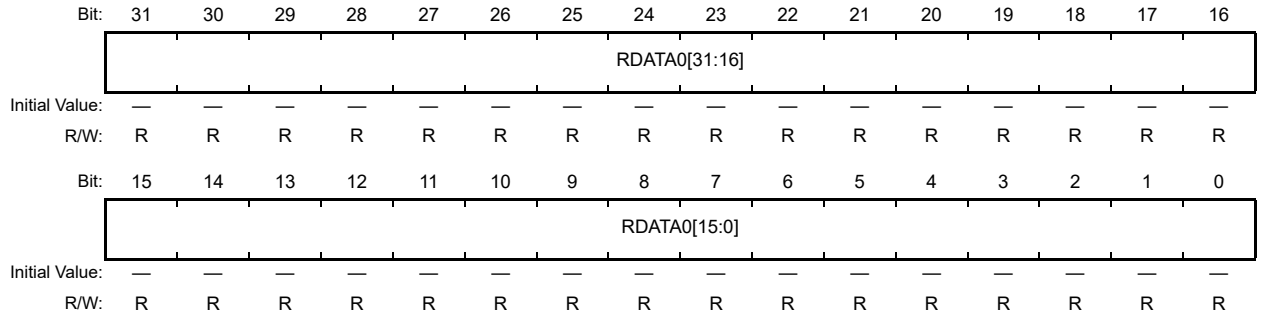
Bit	Bit Name	Initial Value	R/W	Description
31, 30	CDB[1:0]	00	R/W	Command Bit Size Sets the command size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
29, 28	OCDB[1:0]	00	R/W	Optional Command Bit Size Sets the optional command size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
27, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25, 24	ADB[1:0]	00	R/W	Address Bit Size Sets the address size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	OPDB[1:0]	00	R/W	Option Data Bit Size Sets the option data size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	SPIDB[1:0]	00	R/W	Transfer Data Bit Size Sets the transfer data size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
15	DME	0	R/W	Dummy Cycle Enable Enables insertion of the dummy cycle before the read data. Note: Dummy cycle insertion is prohibited for write in Manual mode including the case in which a transfer ends with a dummy cycle. Note: A setting is prohibited for a transfer starting with a dummy cycle. 0: Dummy cycle insertion disabled 1: Dummy cycle insertion enabled
14	CDE	1	R/W	Command Enable Sets the command to be output. 0: Command output disabled 1: Command output enabled Note: When an Octal-SPI flash memory device is connected and the CDE bit in DRENR register set to 1, set this bit to 0.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	OCDE	0	R/W	Optional Command Enable Sets the optional command to be output. 0: Optional command output disabled 1: Optional command output enabled Note: When an Octal-SPI flash memory device is connected, set this bit to 0.
11 to 8	ADE[3:0]	0000	R/W	Address Enable Sets the address to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. (1) When a serial flash memory is connected or a connection is made in Octal-SPI flash memory protocol mode 0000: Output disabled 0100: ADR[23:16] 0110: ADR[23:8] 0111: ADR[23:0] 1111: ADR[31:0] 1100: Addresses in Octal-SPI flash memory protocol mode (operation is in accord with the 8-8-X protocol in Table 20.15) Other than above: Setting prohibited (2) When a HyperFlash memory device is connected 0100: HyperFlash memory Note: The setting of these bits is required to be 0111 or 1111 when the write buffer is in use.
7 to 4	OPDE[3:0]	0000	R/W	Option Data Enable Sets the option data to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. 0000: Output disabled 1000: OPD3 1100: OPD3, OPD2 1110: OPD3, OPD2, OPD1 1111: OPD3, OPD2, OPD1, OPD0 Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	SPIDE[3:0]	0000	R/W	Transfer Data Enable Sets valid transfer data. Valid data differs depending on the BSZ[1:0] bit setting in CMNCR. The following settings must be used. Otherwise, the operation is not guaranteed. (1) BSZ[1:0] bits in CMNCR = 00 (one serial flash memory connected) 0000: Not transferred 1000: 8 bits transferred (enables data at address 0 of the Manual mode read/write data registers 0) 1100: 16 bits transferred (enables data at addresses 0 and 1 of the Manual mode read/write data registers 0) 1111: 32 bits transferred (enables data at addresses 0 to 3 of the Manual mode read/write data registers 0) Other than above: Setting prohibited (2) BSZ[1:0] bits in CMNCR = 01 (two serial flash memories connected) 0000: Not transferred 1000: 16 bits transferred (enables data at addresses 0 and 1 of the Manual mode read/write data registers 0) 1100: 32 bits transferred (enables data at addresses 0 to 3 of the Manual mode read/write data registers 0) 1111: 64 bits transferred (enables data at addresses 0 to 3 of the Manual mode read/write data registers 0 and data at addresses 0 to 3 of the Manual mode read/write data registers 1) Other than above: Setting prohibited

20.4.13 Manual Mode Read Data Register 0 (SMRDR0)

SMRDR0 is a 32-bit register that stores the read data in Manual mode.
Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the Manual mode enable setting register (SMENR). Be sure to access from LSB.
The settings of this register should be read when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

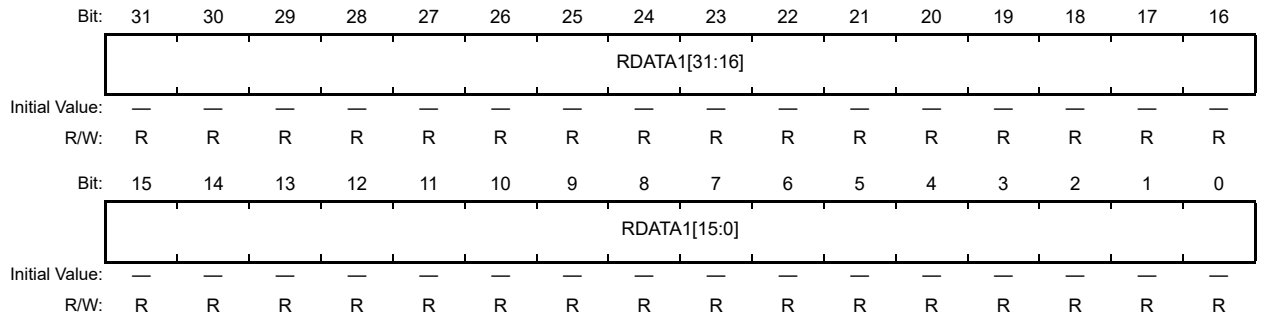


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDATA0[31:0]	Undefined	R	Read Data Holds the data read in Manual mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = 00: Read data[31:0]. BSZ[1:0] = 01: Read data[63:32].

Note: The contents of this register and SMRDR1 are modified upon completion of reception in manual mode. Be sure to read data when reception in manual mode is completed.

20.4.14 Manual Mode Read Data Register 1 (SMRDR1)

SMRDR1 is a 32-bit register that stores the read data in Manual mode.
Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the Manual mode enable setting register (SMENR). Be sure to access from LSB.
The settings of this register should be read when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDATA1[31:0]	Undefined	R	Read Data Holds the data read in Manual mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = 00: Bits in this register are disabled. BSZ[1:0] = 01: Read data[31:0]

20.4.15 Manual Mode Write Data Register 0 (SMWDR0)

SMWDR0 is a 32-bit register that sets the write data in Manual mode.
Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the Manual mode enable setting register (SMENR). Be sure to access from address 0.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WDATA0[31:16]															
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WDATA0[15:0]															
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	WDATA0 [31:0]	All 0	R/W	Write Data Holds the data to be written in Manual mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = 00: Write data[31:0]. BSZ[1:0] = 01: Write data[63:32]. Note: In Octal-SPI flash memory protocol mode (OPI), the write data should be arranged and written as follows. • To write data D[31:0] in the STR mode WDATA0[31:0]={D[23:20], D[31:28], D[19:16], D[27:24], D[7: 4], D[15:12], D[3: 0], D[11: 8] } • To write data D[31:0] in the DTR mode WDATA0[31:0]={D[31:28], D[23:20], D[27:24], D[19:16], D[15:12], D[7:4], D[11:8], D[3:0]}

20.4.16 Manual Mode Write Data Register 1 (SMWDR1)

SMWDR1 is a 32-bit register that sets the write data in Manual mode.
This register is enabled when the BSZ[1:0] bits in CMNCR are set to 01 and disabled when the BSZ[1:0] bits in CMNCR are set to 00.
Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the Manual mode enable setting register (SMENR). Be sure to access from address 0.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WDATA1[31:16]															
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit: :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WDATA1[15:0]															
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	WDATA1 [31:0]	All 0	R/W	Write Data Holds the data to be written in Manual mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = 00: Bits in this register are disabled. BSZ[1:0] = 01: Write data[31:0]. Note: In Octal-SPI flash memory protocol mode (OPI), the write data should be arranged and written as follows. •To write data D[31:0] in the STR mode WDATA1[31:0]={D[23:20], D[31:28], D[19:16], D[27:24], D[7: 4], D[15:12], D[3: 0], D[11: 8]} •To write data D[31:0] in the DTR mode WDATA1[31:0]={D[31:28], D[23:20], D[27:24], D[19:16], D[15:12], D[7:4], D[11:8], D[3:0]}

20.4.17 Common Status Register (CMNSR)

CMNSR is a 32-bit register that holds flags indicating the operating state.
The settings of this register are reflected both in external address space read mode and Manual mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSLF	TEND
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	SSLF	0	R	QSPIn_SSL Pin Monitor 0: QSPIn_SSL pin is negated 1: QSPIn_SSL pin is asserted
0	TEND	1	R	Transfer End Flag Indicates whether the data transfer has ended. 0: Indicates that data transfer is in progress 1: Indicates that data transfer has ended

20.4.18 Data Read Dummy Cycle Setting Register (DRDMCR)

DRDMCR is a 32-bit register that sets the number of dummy cycles to be inserted in external address space read mode. The settings of this register are enabled when the DME bit in the data read enable setting register (DRENR) is 1.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	DMCYC[4:0]				
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	DMCYC[4:0]	H'0B	R/W	Number of Dummy Cycles Sets the number of dummy cycles to be inserted when the DME bit in the data read enable setting register (DRENR) is 1. 00001: 2 cycles 00010: 3 cycles … 10010: 19 cycles 10011: 20 cycles Other than above: Setting prohibited

20.4.19 Data Read DDR Enable Register (DRDRENR)

DRDRENR is a 32-bit register that sets SDR or DDR transfer of the address, option data, and read data in external address space read mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	HYPE			—	—	—	ADDRE	—	—	—	OPDRE	—	—	—	DRDRE
Initial Value:	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	1
R/W:	R	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14 to 12	HYPE	101	R/W	HyperFlash, Octal-SPI flash memory DDR mode enable 101: When HyperFlash memory is connected, not only commands but also addresses and data are transferred according to the Octal-SPI flash memory protocol for DDR mode. 100: Only commands are transferred according to the Octal-SPI flash memory protocol for DDR mode. 000: SPI flash mode Other than above: Setting prohibited
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	ADDRE	1	R/W	Address DDR Enable Sets SDR or DDR transfer of the address. 0: SDR transfer 1: DDR transfer Note: Set this bit to 1 when the HyperFlash memory is connected.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OPDRE	0	R/W	Option Data DDR Enable Sets SDR or DDR transfer of the option data. 0: SDR transfer 1: DDR transfer
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	DRDRE	1	R/W	Data Read DDR Enable Sets SDR or DDR transfer of the read data. 0: SDR transfer 1: DDR transfer

20.4.20 Manual Mode Dummy Cycle Setting Register (SMDMCR)

SMDMCR is a 32-bit register that sets the number of dummy cycles to be inserted in Manual mode.
The settings of this register are enabled when the DME bit in the Manual mode enable setting register (SMENR) is 1.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	DMCYC[4:0]				
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	DMCYC[4:0]	00000	R/W	Number of Dummy Cycles Sets the number of dummy cycles to be inserted when the DME bit in the Manual mode enable setting register (SMENR) is 1. 00001: 2 cycles 00010: 3 cycles .... 10010: 19 cycles 10011: 20 cycles Other than above: Setting prohibited

20.4.21 Manual Mode DDR Enable Register (SMDRENR)

SMDRENR is a 32-bit register that sets SDR or DDR transfer of the address, option data, and transfer data in Manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	HYPE[2:0]			—	—	—	ADDRE	—	—	—	OPDRE	—	—	—	SPIDRE
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14 to 12	HYPE[2:0]	000	R/W	HyperFlash, Octal-SPI Flash Memory Protocol Mode DDR mode enable 101: When HyperFlash memory is connected, not only commands but also addresses and data are transferred according to the Octal-SPI flash memory protocol for DDR mode. 100: Only commands are transferred according to the Octal-SPI flash memory protocol for DDR mode. 000: SPI flash mode Other than above: Setting prohibited
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	ADDRE	0	R/W	Address DDR Enable Sets SDR or DDR transfer of the address. 0: SDR transfer 1: DDR transfer Note: Set this bit to 1 when the HyperFlash memory is connected.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OPDRE	0	R/W	Option Data DDR Enable Sets SDR or DDR transfer of the option data. 0: SDR transfer 1: DDR transfer
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SPIDRE	0	R/W	Transfer Data DDR Enable Sets SDR or DDR transfer of the transfer data. 0: SDR transfer 1: DDR transfer Note: Set this bit to 1 when the HyperFlash memory is connected.

20.4.22 PHY Control Register (PHYCNT)

PHYCNT is a 32-bit register that sets the PHY operation mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAL	ALT_ ALIGN	—	—	—	—	—	—	OCTA[1:0]		EXDS	OCT	—	HS	CKSEL[1:0]	
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	WBUF2	—	WBUF	PHYMEM[1:0]	
Initial Value:	0	0	0	0	0	0	1	0	0	1	1	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CAL	0	W	PHY Calibration Specifies whether to perform PHY calibration. Be sure to perform calibration before access in Manual mode. 0: Calibration is not performed. 1: Calibration is performed.
30	ALT_ ALIGN	0	R/W	Alternative Alignment for Octal-SPI Flash Memory Protocol Mode 0: Alternative alignment for a connection in Octal-SPI flash memory protocol mode is not supported. 1: Alternative alignment for a connection in Octal-SPI flash memory protocol mode is supported.
29 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23, 22	OCTA [1:0]	00	R/W	Alignment in Octal-SPI Flash Memory Protocol Mode Specify the data alignment for DDR transfer in Octal-SPI flash memory protocol mode. For details, refer to section 20.5.14, Data Alignment in Octal-SPI Flash Memory Protocol Mode. When the connected memory is HyperFlash or serial flash memory, set these bits to 00. 00: Specify this value when a HyperFlash or serial flash memory device is connected or a connection is made in Octal-SPI flash memory protocol mode (SDR mode). 01: Alternative alignment is supported. 10: Sequential alignment is supported. 11: Setting prohibited. Note 1. When the alternative alignment of Octal-SPI flash memory protocol mode is to be used, also set the ALT_ALIGN bit to 1. Note 2. When the sequential alignment of Octal-SPI flash memory protocol mode is to be used, the setting of the ALT_ALIGN bit is treated as “don’t care”.
21	EXDS	0	R/W	External Data Strobe When the data strobe line is connected to the serial flash memory, the external data strobe signal sent from the serial flash memory is used. When a connection is made in Octal-SPI flash memory protocol mode, set this bit to 1. Note: When the HyperFlash memory is connected, set this bit to 0. 0: The external data strobe signal is not used. 1: The external data strobe signal is used.
20	OCT	0	R/W	Octal-SPI flash memory Protocol Mode 0: A mode other than the Octa protocol mode is used. 1: Octal-SPI flash memory protocol mode is used.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
18	HS	0	R/W	High-Speed Response Mode Specifies the high-speed response mode. 0: Data is read for the number of data units specified in the RBURST bits of the DRCR register and then output to the bus master. 1: The read data is output to bus master in parallel of device access when DRCR.RBE = 1. Note 1. When this bit is set to 1, use DMA transfer. The transfer size in the RBURST[4:0] bits of the DRCR register should be fixed to 5'h1F. Note 2. Do not access the register area during DMA transfer. Note 3. When this bit is set to 1, access address alignment to should be 256Byte align and clear the cache entry by setting the RCF bit in DRCR to 1 before starting DMA transfer.
17, 16	CKSEL [1:0]	00	R/W	Clock Timing Switching Adjusts the timing of the clock when serial flash is connected. Always follow the procedure for setting shown in section 20.5.17, Timing Adjustment. Be sure to set the CKSEL[1:0] bits to b'11 when a connection is made in Octal-SPI flash memory protocol mode or HyperFlash memory is connected.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
8, 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6, 5	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
4	WBUF2	0	R/W	Write Buffer Enable 2 Specifies whether to use the write buffer to write data to the flash memory. For the usage of the write buffer, refer to section 20.5.13, Write Buffer Operation. 0: The write buffer is not used. 1: The write buffer is used to write data to the flash memory.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	WBUF	0	R/W	Write Buffer Enable Specifies whether to use the write buffer to write data to the flash memory. For the usage of the write buffer, refer to section 20.5.13, Write Buffer Operation. 0: The write buffer is not used. 1: The write buffer is used to write data to the flash memory.
1, 0	PHYMEM [1:0]	11	R/W	Device Selection Selects the device to be connected. 00: Serial flash or Octal-SPI flash memory in SDR mode 01: Serial flash or Octal-SPI flash memory in DDR mode 11: HyperFlash memory Other than above: Setting prohibited

20.4.23 PHY Offset Register 1 (PHYOFFSET1)

PHYOFFSET1 is a 32-bit register with two effective bits. These are for selecting DDR or SDR operation.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	DDRTMG		—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	1	0	0	0	0	1	0	1	0	1	0	0	0	1
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	1	0	0	0	1	0	1	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29, 28	DDRTMG	10	R/W	SDR or DDR Operation Register 10: Set these bits when data are to be read by DDR transfer in the external address space read mode or manual mode. 11: Set these bits when data are to be read by SDR transfer in the external address space read mode or manual mode. Other than above: Setting prohibited
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
5 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

20.4.24 PHY Offset Register 2 (PHYOFFSET2)

PHYOFFSET2 is a 32-bit register that sets the timing adjustment in the DDR operation.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OCTTMG			—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	OCTTMG	100	R/W	Octal-SPI Flash Memory Protocol Mode Operation Timing Adjusts the timing of writing in Octal-SPI flash memory protocol mode. 000: Specify this value when using the serial flash memory with write buffer operation of the bit width 1-1-4 or 1-4-4 in Table 20.14. 100: Specify this value when the serial flash or HyperFlash memory is used. 011: Specify this value when a connection is made in Octal-SPI flash memory protocol mode. Other than above: Setting prohibited Note: In Octal-SPI flash memory protocol mode and the serial flash compatible mode, set these bits to 100.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

20.4.25 PHY Interrupt Register (PHYINT)

PHYINT is a 32-bit register for making settings to do with interrupt signals and pins.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RSTEN	WPEN	INTEN	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RST VAL	WPVAL	INT
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	RSTEN	1	R/W	RPC_RESET# Pin Enable Enables or disables the RPC_RESET# pin. 0: The RPC_RESET# pin is disabled. 1: The RPC_RESET# pin is enabled and the value specified in the RSTVAL bit is output.
25	WPEN	1	R/W	RPC_WP# Pin Enable Enables or disables the RPC_WP# pin. 0: The RPC_WP# pin is disabled. 1: The RPC_WP# pin is enabled and the value specified in the WPVAL bit is output.
24	INTEN	1	R/W	RPC_INT# Pin Enable Enables or disables the RPC_INT# pin. 0: The RPC_INT# pin is disabled. 1: The RPC_INT# pin is enabled and the interrupt signal sent from the HyperFlash memory is enabled.
23 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	RSTVAL	0	R/W	Value Output through RPC_RESET# Pin Specifies the value to be output through the RPC_RESET# pin. This setting takes effect when the RSTEN bit is set to 1. 0: RPC_RESET# = H 1: RPC_RESET# = L
1	WPVAL	1	R/W	Value Output through RPC_WP# Pin Specifies the value to be output through the RPC_WP# pin. This setting takes effect when the WPEN bit is set to 1. 0: RPC_WP# = H 1: RPC_WP# = L
0	INT	0	R	Interrupt Status When the RPC_INT# signal goes low, this bit is set to 1 to indicate the occurrence of an interrupt in the connected device.

20.4.26 PHY Adjustment Register 1 (PHYADJ1)

PHYADJ1 is a 32-bit register used to adjust the timing when serial flash memory is connected.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADJ1[31:16]															
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADJ1[15:0]															
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ADJ1[31:0]	All 0	R/W	Serial Flash Memory Operation Timing Adjustment Set these bits by following the procedure shown in Figure 20.29, (1) Flow of Timing Adjustment when the Serial Flash Memory is Connected (SDR Mode Except in the Case of Octal-SPI Flash Memory) and Figure 20.29, (2) Flow of Timing Adjustment when the Serial Flash Memory is Connected (DDR Mode). Note: Write the setting values described in the order indicated for the flow of adjustment.

Note: Bits 31 to 6 are always read as 0.

20.4.27 PHY Adjustment Register 2 (PHYADJ2)

PHYADJ2 is a 32-bit register used to adjust the timing when serial flash memory is connected.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADJ2[31:16]															
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADJ2[15:0]															
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ADJ2[31:0]	All 0	R/W	Serial Flash Memory Operation Timing Adjustment Set these bits by following the procedure shown in Figure 20.29, (1) Flow of Timing Adjustment when the Serial Flash Memory is Connected (SDR Mode Except in the Case of Octal-SPI Flash Memory) and Figure 20.29, (2) Flow of Timing Adjustment when the Serial Flash Memory is Connected (DDR Mode). Note: Write the setting values described in the order indicated for the flow of adjustment.

20.5 Operation

20.5.1 System Configuration

This module can connect one or two serial flash memory devices, an Octal-SPI flash memory device, or a HyperFlash memory device. The number of connected memories can be selected using the BSZ[1:0] bits in CMNCR.

Examples of system configuration are shown in Figure 20.2, Figure 20.3, Figure 20.4, and Figure 20.5 respectively.

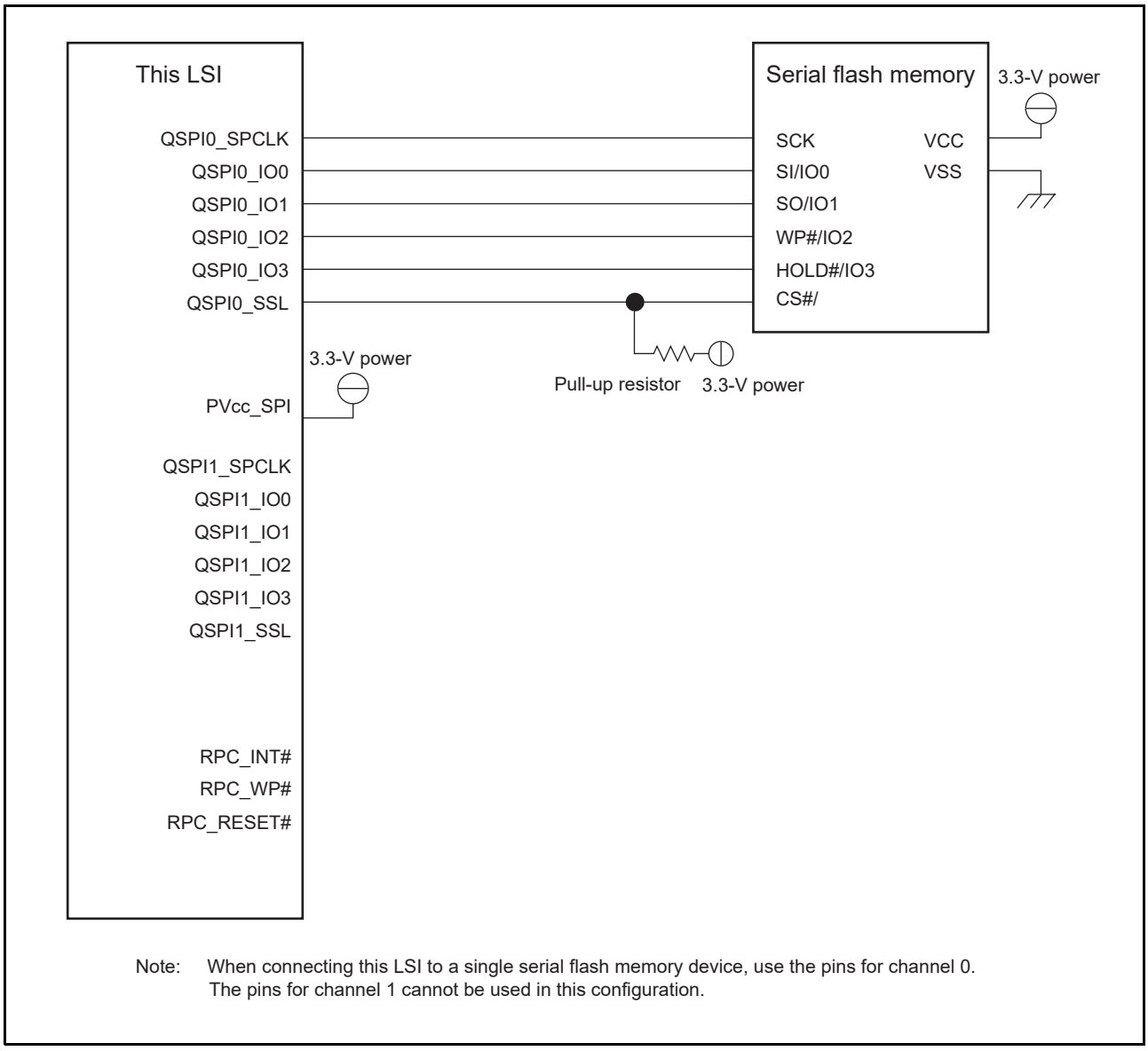


Figure 20.2 System Configuration Example with 4-Bit Data Size and One Serial Flash Memory Connected

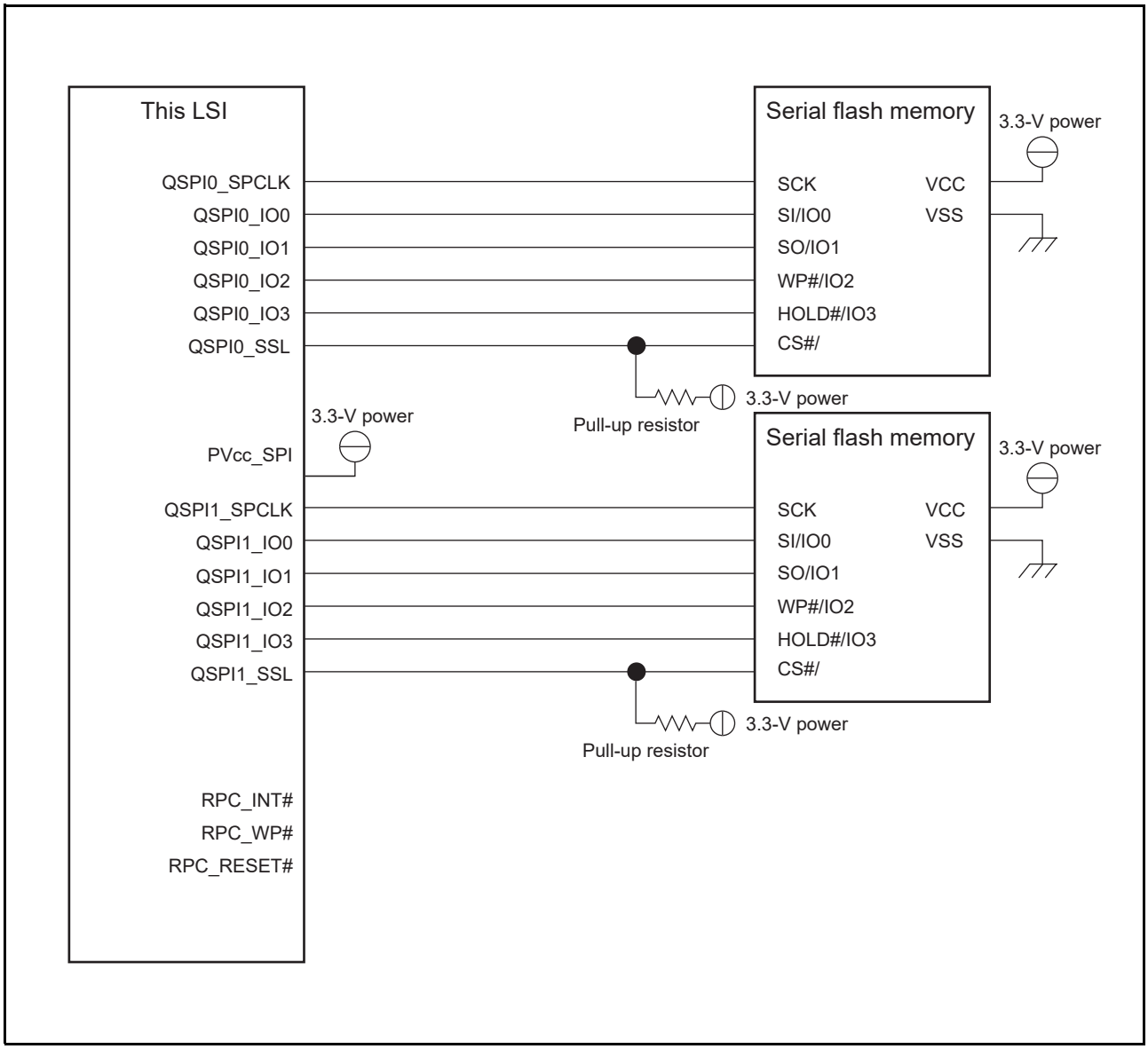


Figure 20.3 System Configuration Example with 4-Bit Data Size and Two Serial Flash Memories Connected (BSZ[1:0] Bits in CMNCR = 01)

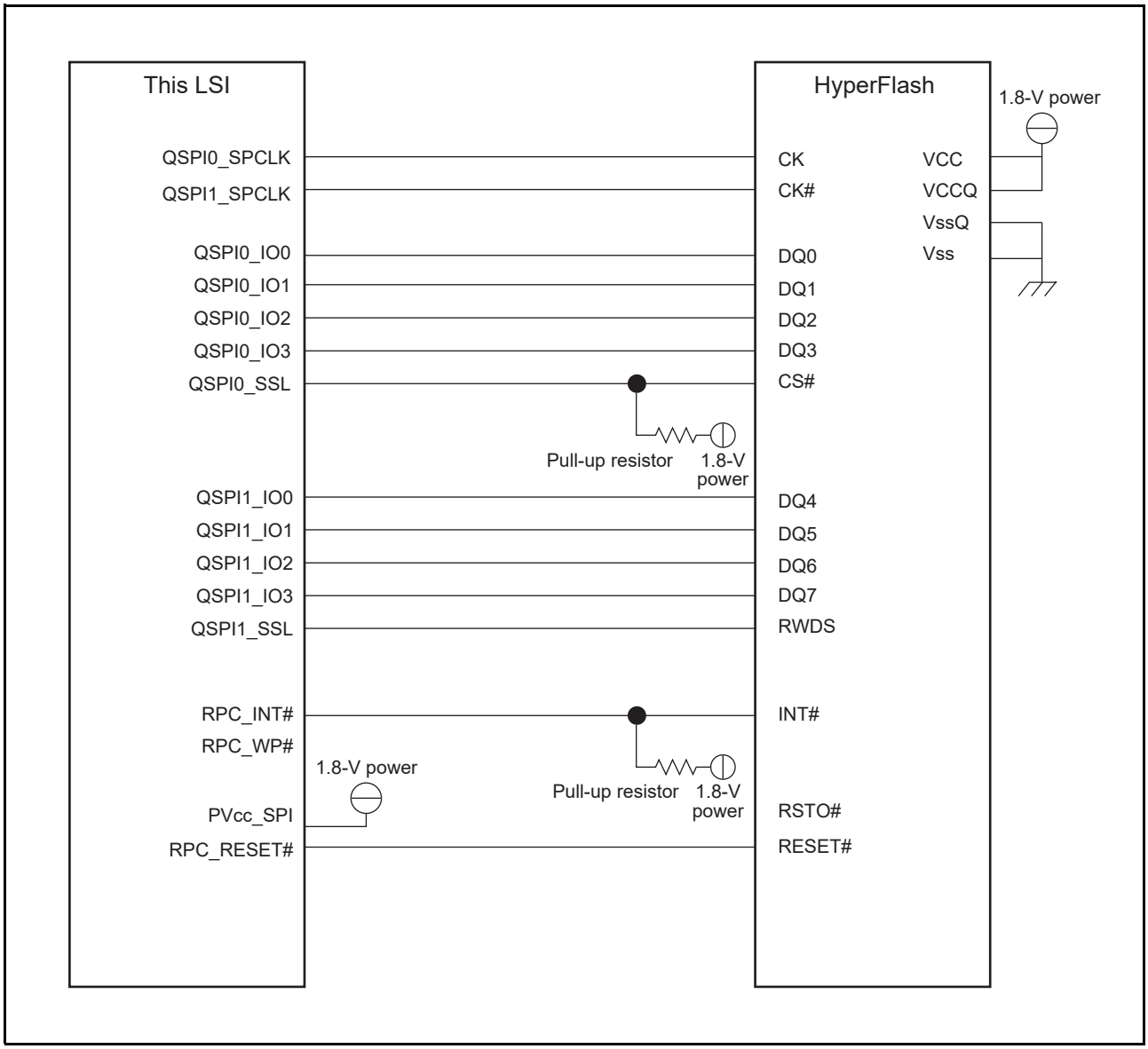


Figure 20.4 System Configuration Example with HyperFlash Connected

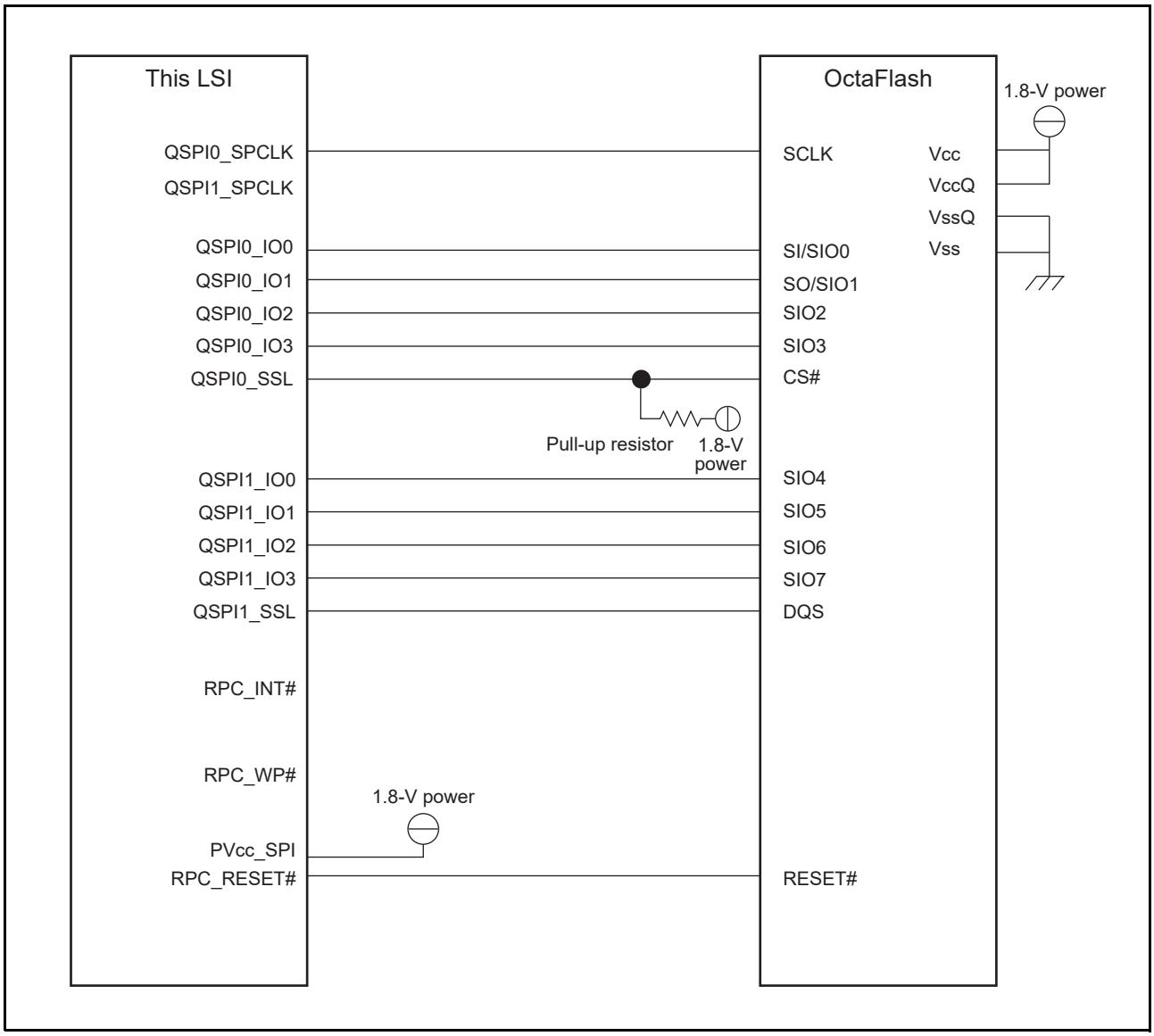


Figure 20.5 System Configuration Example with OctaFlash Connected

20.5.2 Address Map

In external address space read mode, the serial, HyperFlash, or Octal-SPI flash memory to be connected are allocated to the SPI multi I/O bus space. The internal address space is 256 Mbytes. However, specifying offset addresses in the DREAR register allows access to a maximum of 4 Gbytes when a single serial flash memory or Octal-SPI flash memory device is connected, and to a maximum of 8 Gbytes when two serial flash memory devices are connected or a single HyperFlash memory device is connected. When HyperFlash memory is connected, address expansion cannot be performed in the external address space read mode.
Use manual mode when HyperFlash memory is connected and an address range greater than 256 Mbytes is to be used.

Table 20.3 Address Map

Internal Address	Max. Access Area
H'20000000 to H'2FFFFFFF	A serial flash memory or Octal-SPI flash memory device is connected: 4 Gbytes Two serial flash memory devices are connected or a single HyperFlash memory device is connected in manual mode: 8 Gbytes

Note: When access protection is set, not only this address space, but also the register area is protected.

20.5.3 32-bit Serial Flash Addresses

Since the SPI multi I/O bus space is 256 Mbytes, only a part of the 32-bit serial flash address area can be directly accessed. Here, the fixed value set in the pertinent register is used as the upper bit value of a 32-bit address.
To output serial flash addresses in 32 bits, set the ADE[3] bit in DRENR to 1, set the range of the external addresses used as the serial flash addresses to the EAC[2:0] bits in DREAR, and set the upper bit value of the 32-bit address as the fixed value to the EAV[7:0] bits in DREAR.

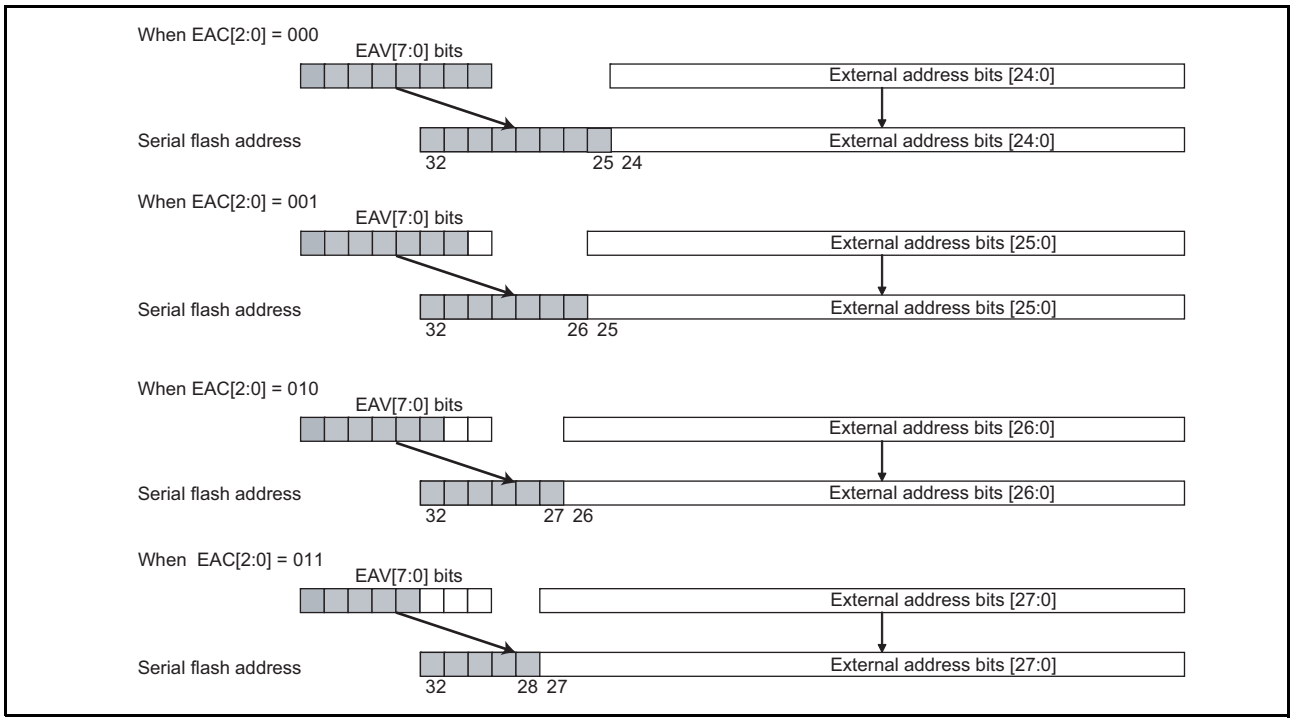


Figure 20.6 32-Bit Address Setting

Setting the ADE[3] bit in DRENR to 1 allows the serial flash address to be output using [31:0] bits.

When EAC[2:0] = 000, external address bits [24:0] are valid; set the value for [32:25] bits to EAV[7:0].
When EAC[2:0] = 001, external address bits [25:0] are valid; set the value for [32:26] bits to EAV[7:1].
When EAC[2:0] = 010, external address bits [26:0] are valid; set the value for [32:27] bits to EAV[7:2].
When EAC[2:0] = 011, external address bits [27:0] are valid; set the value for [32:28] bits to EAV[7:3].

The address bits actually used for access depend on the number of serial flash memories connected.
When a single serial flash memory device is connected or the Octal-SPI flash memory protocol is in use, address bits [31:0] are used. When two serial flash memory devices are connected, address bits [32:1] are used.

20.5.4 Operating Modes

This module has two operating modes: external address space read mode and Manual mode.
In external address space read mode, a read access to the SPI multi I/O bus space is converted into Data read protocol and data is received. After data acquisition, data is returned to the bus master that is the issuing source. For details, see section 20.5.5, External Address Space Read Mode.
In Manual mode, arbitrary protocol is carried out using register settings. For details, see section 20.5.7, Manual mode.

20.5.5 External Address Space Read Mode

A read access to the SPI multi I/O bus space can be converted into Data read protocol in external address space read mode. Further, the commands, optional commands, option data, and dummy cycle issued for reading can be modified using registers.
In external address space read mode, either normal read operation or burst read operation can be selected. The transfer format is determined based on the common control register (CMNCR), SSL delay register (SSLDR), data read control register (DRCR), data read command setting register (DRCMR), data read extended address setting register (DREAR), data read option setting register (DROPR), data read enable setting register (DRENR), data read dummy cycle setting register (DRDMCR), and data read DDR enable register (DRDRENR).

(1) Normal Read Operation

When the RBE bit in the DRCR register is set to 0, normal read operation is performed. In the normal read operation, data is read from the external device when the bus master executes read access. After data is read, the QSPIn_SSL pins are negated.
The timing of normal read operation for the serial flash memory is shown in Figure 20.7.
t1 is the time period from QSPIn_SSL pin assertion to QSPIn_SPCLK oscillation (clock delay), t2 is the time period from transmission of the last QSPIn_CLK edge of a transfer to QSPIn_SSL pin negation (QSPIn_SSL negation delay), and t3 is the time period from one transfer end to the next transfer start (next access).

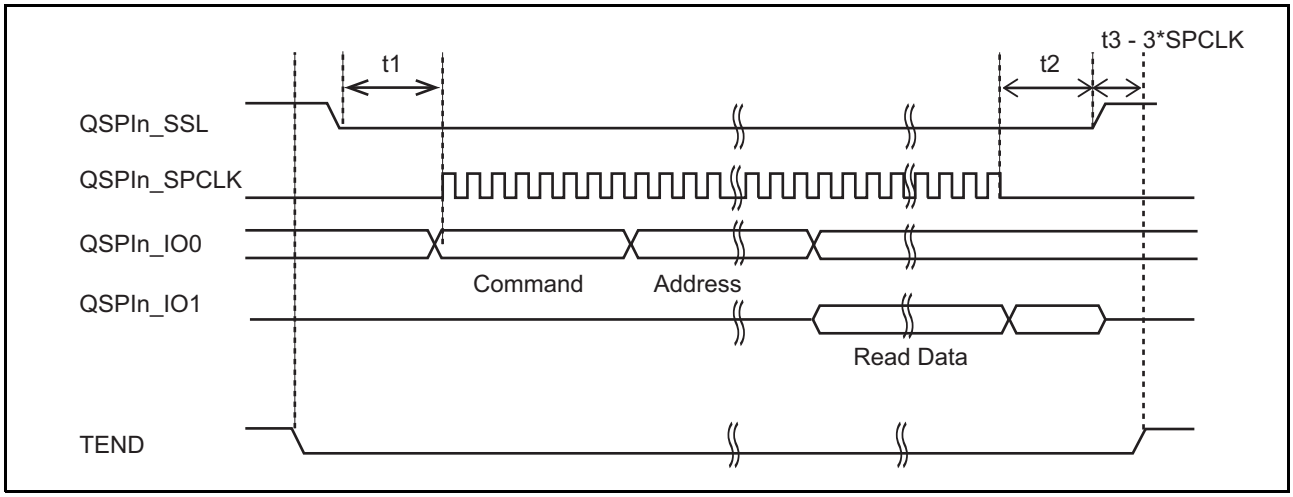


Figure 20.7 Normal Read Operation Timing

Note 1. T1 to T3 are specified with SSLDR.SCKDL, SSLDR.SLNDL and SSLDR.SPNDL bits, respectively.

(2) Burst Read Operation

When the RBE bit in DRCR is set to 1, burst read operation is performed.
Read cache is enabled in the burst read operation. For read cache operation, see section 20.5.6, Read Cache.
When the bus master executes read access, the read cache is first searched for the target data. When the read cache contains the data, it is read from the cache instead of accessing the external device. When the read cache does not contain the data, the external device is read in the burst mode and the read data is stored in the read cache and returned to the bus master. In this operation, the data transfer length is 64 bits × the value in the RBURST[4:0] bits, and data is always read from a 64-bit boundary address.
The QSPIn_SSL pin status after data transfer can be selected by using the SSLE bit in DRCR. When the SSLE bit is set to 0, the QSPIn_SSL pin is negated after data transfer. For an operation performed when the SSLE bit is set to 1, see section 20.5.5, (3), Burst Read Operation with Automatic QSPIn_SSL Negation, just below.
A pattern diagram of this operation and a burst read operation timing diagram when SSLE bit is set to 0 are shown in Figure 20.8 and Figure 20.9.

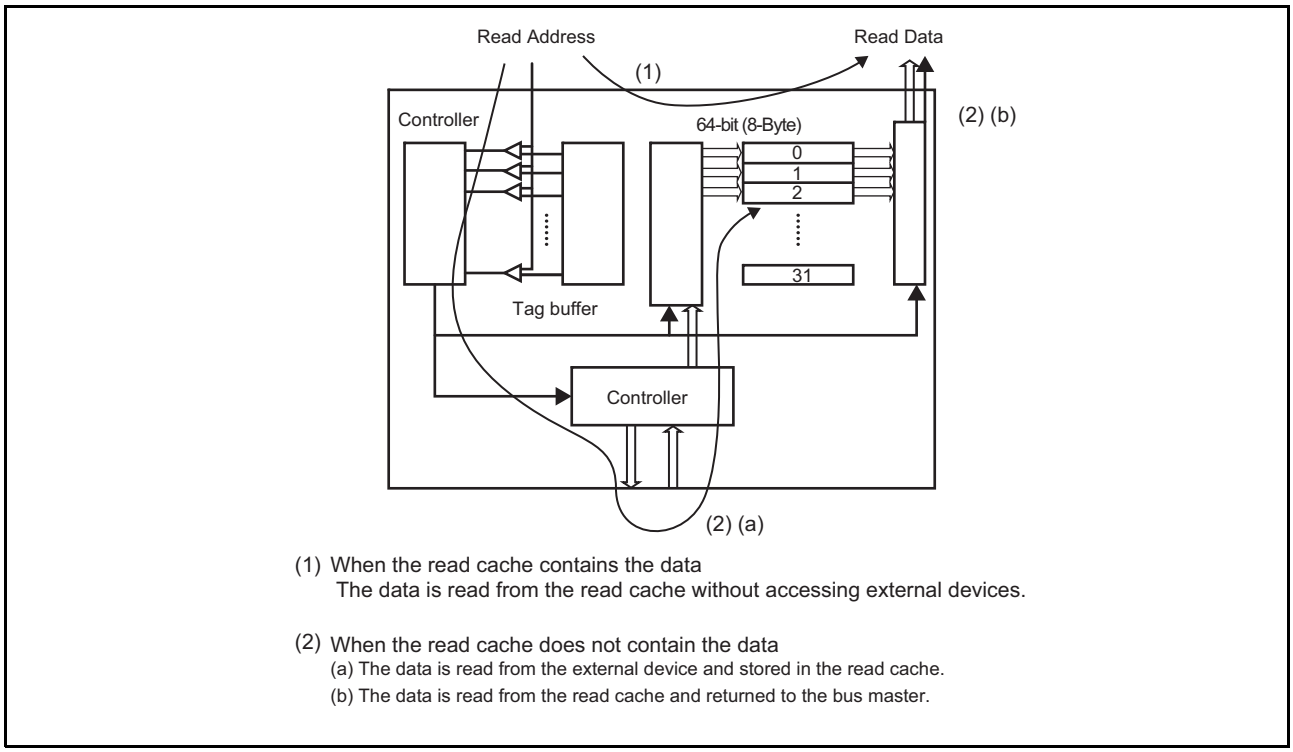


Figure 20.8 Burst Read Operation

Note: When connecting this LSI to a single serial flash memory device, use the pins for channel 0. The pins for channel 1 cannot be used in this configuration.

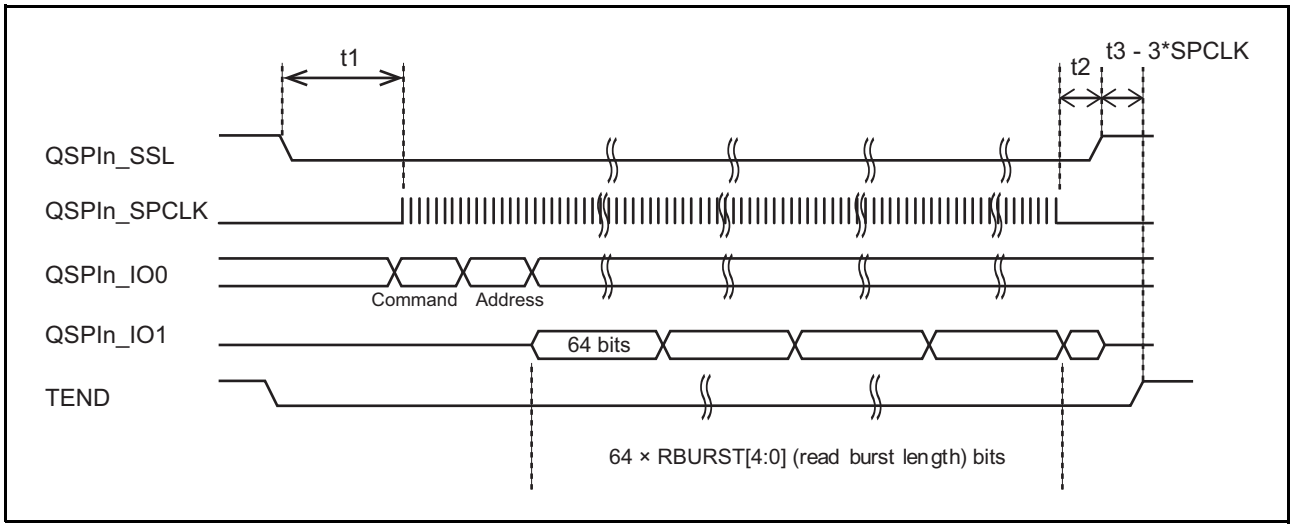


Figure 20.9 Burst Read Operation Timing

Note: T1 to T3 are specified with SSLDR.SCKDL, SSLDR.SLNDL and SSLDR.SPNDL bits, respectively.

(3) Burst Read Operation with Automatic QSPIn_SSL Negation

When SSLE bit in DRCR is set to 1, this module does not negate the QSPIn_SSL pin after the burst read transfer. When accessing the next time, if the address is continuous with the previous read address, the burst read operation is performed without issuing the command, optional command, address, option data, or dummy cycle. If the address is not continuous with the previous read address, the QSPIn_SSL pin is once negated and the burst read operation is performed after issuing the command, optional command, address, option data, or dummy cycle.
Burst read timing diagrams for continuous address and non-continuous address are shown in Figure 20.10 and Figure 20.11 .
For the next access after negation of the QSPIn_SSL with the SSLN bit in DRCR with this operation, read SSLF = 0 in CMNSR to confirm that the QSPIn_SSL has been negated.
This function is available when serial flash memory is connected.

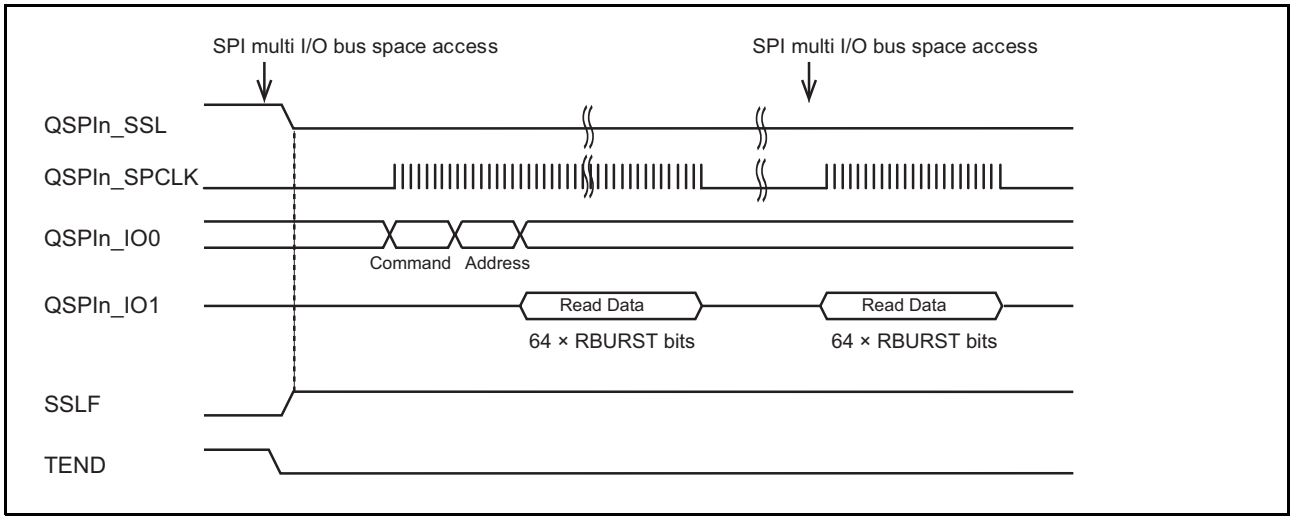


Figure 20.10 Burst Read Timing for Continuous Address (SSLE Bit = 1)

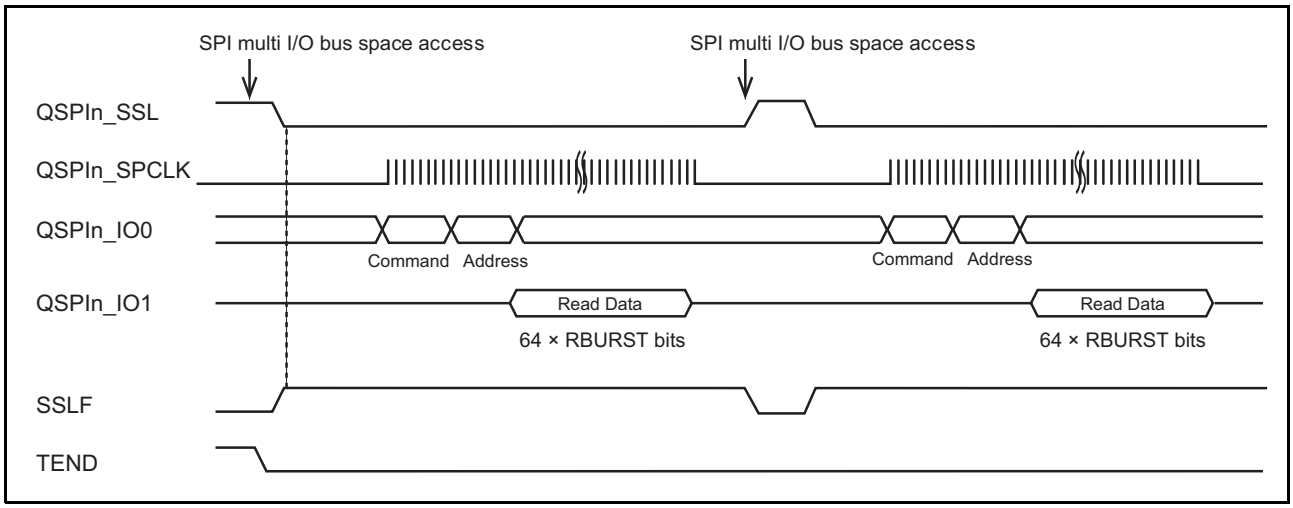


Figure 20.11 Burst Read Timing for Non-Continuous Address (SSLE Bit = 1)

(4) Initial Setting Flow

An example of an initial setting flow in external address space read mode is shown in Figure 20.12.

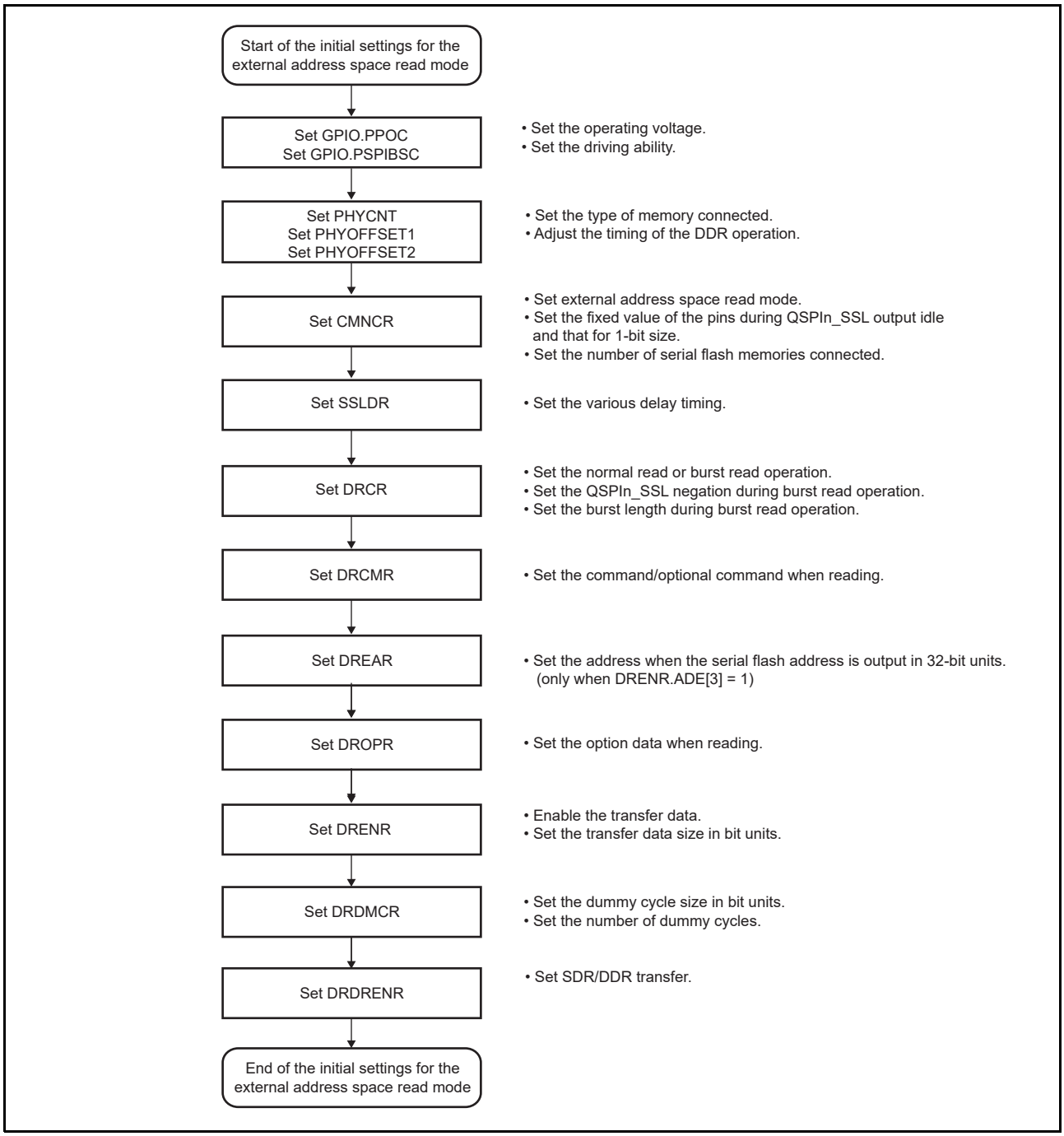


Figure 20.12 Example of Initial Setting Flow in External Address Space Read Mode

20.5.6 Read Cache

This module has a simple built-in read cache. The read cache can be used during external address space read mode and burst read operation. The read cache is configured with a line size of 64 bits and 32 entries.
Read cache configuration is shown in Figure 20.13.

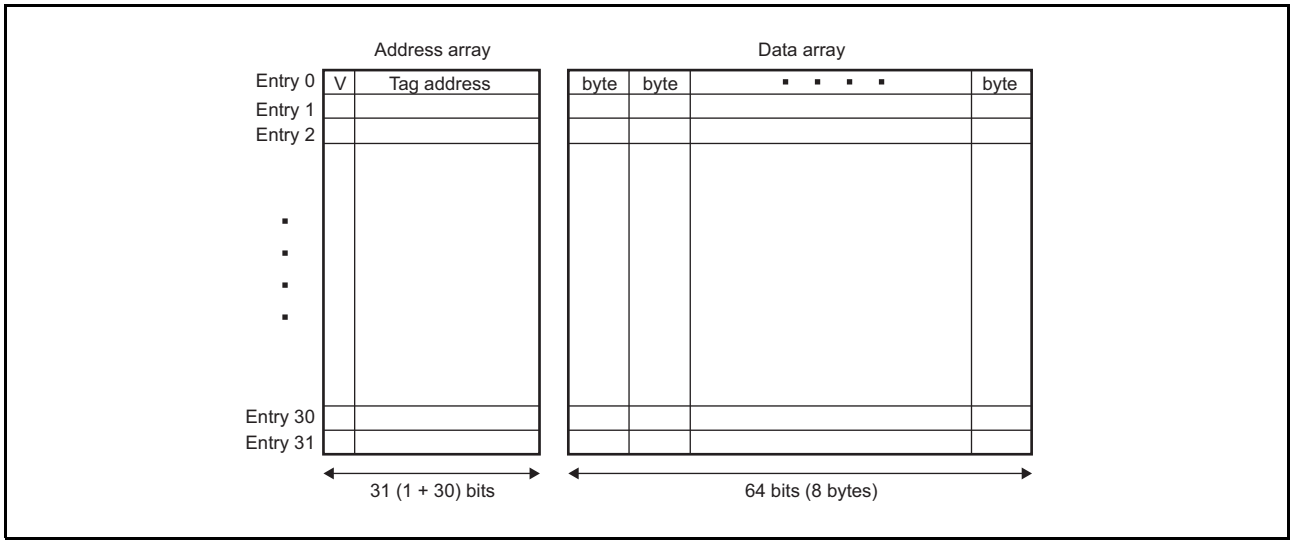


Figure 20.13 Read Cache Configuration

(1) Address Array

The V bit indicates whether the entry data is valid. When the V bit is 1, the data is valid and when V bit is 0, the data is invalid.
The tag address bits hold the address used for the serial flash memory. Address bits 32 to 3 are used for the purpose. Address bits 23 to 3 are enabled when address output is 24 bits and one serial flash memory is connected; and address bits 24 to 3 are enabled when two serial flash memories are connected.
Address bits 31 to 3 are enabled when address output is 32 bits and one serial flash memory is connected; and address bits 32 to 3 are enabled when two serial flash memories are connected.

(2) Data Array

It retains the 64-bit read data. Registration in the read cache is performed in line units.

(3) Read Operation

In case of read-hit, data is read from the read cache. In case of read-miss, after the 64 × RBURST (read burst length) data is read from the serial flash memory and the read cache is updated, the data is returned to the bus master.

(4) Data Replacement

The write pointer is used to update data. In case of read-miss, the RBURST (read burst length) portion data is replaced starting at the entry specified by the write pointer. In other words, the data is replaced in the storage order of the data. Whether data is referred to or not will not affect the replacement order of data.

20.5.7 Manual mode

This module can carry out an arbitrary serial transmission operation by using the register settings.
The transfer format is determined based on the common control register (CMNCR), SSL delay register (SSLDR), Manual mode control register (SMCR), Manual mode command setting register (SMCMR), Manual mode address setting register (SMADR), Manual mode option setting register (SMOPR), and Manual mode enable setting register (SMENR), Manual mode read data register (SMRDR), Manual mode write data register (SMWDR), Manual mode dummy cycle setting register (SMDMCR), and Manual mode DDR enable register (SMDRENR).
Manual mode can be used for reading the status of the serial flash memory and writing to the serial flash memory.
In this mode, one transfer refers to the operation from when the SPIE bit in SMCR is set to 1 to when the TEND bit is set to 1.

(1) Transfer Start

The transfer of data is started in the set transfer format by setting the SPIE bit in SMCR to 1. When write operation is enabled, the Manual mode write data register is transmitted to the serial flash memory. When read operation is enabled, data read from the serial flash memory is stored into the Manual mode read data register.
The Manual operation timing is shown in Figure 20.14.

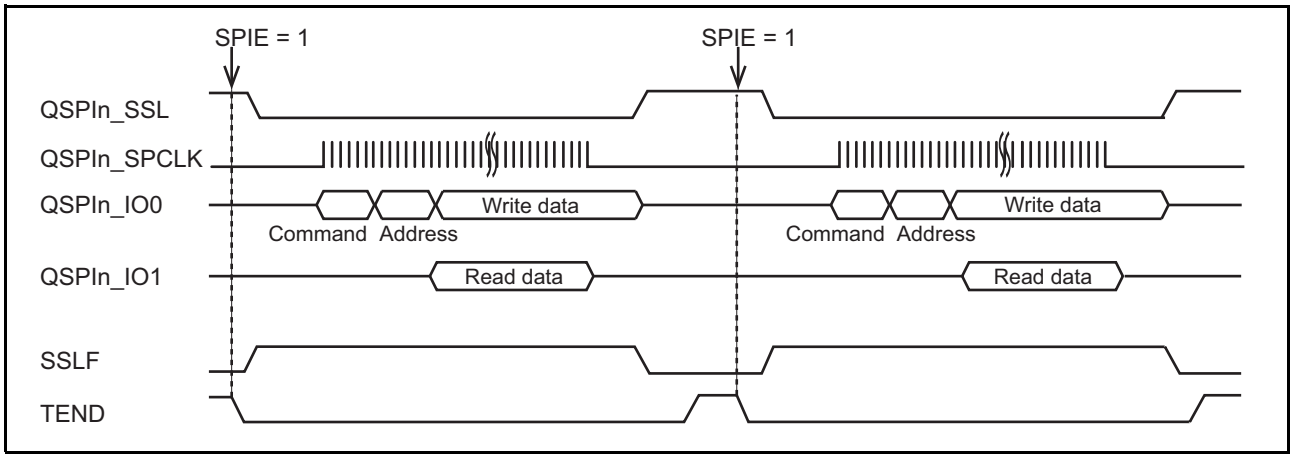


Figure 20.14 Serial transmission operation timing

(2) Read/Write Enable

- Read operation: Data can be read by setting the SPIRE bit in SMCR to 1. The read data is stored into SMRDR.
- Write operation: Data can be written by setting the SPIWE bit in SMCR to 1. The data stored in SMWDR is output.

When the data size is set to 1 bit using the SPIDB[1:0] bits in SMENR, data can be transmitted and received by setting the SPIRE and SPIWE bits to 1. However, when the data size is set to 4 bits by using the SPIDB[1:0] bits, only one of the SPIRE and SPIWE bits should be enabled. The operation is not guaranteed if the both bits are enabled.

(3) Retention of QSPIn_SSL Pin Assertion

By setting the SSLKP bit in SMCR to 1, assertion of the QSPIn_SSL pin can be continued till the next transfer. With this function, the transfer can be carried out continuously with the QSPIn_SSL kept in the asserted state.
The data transfer timing using the SSLKP bit is shown in Figure 20.15.

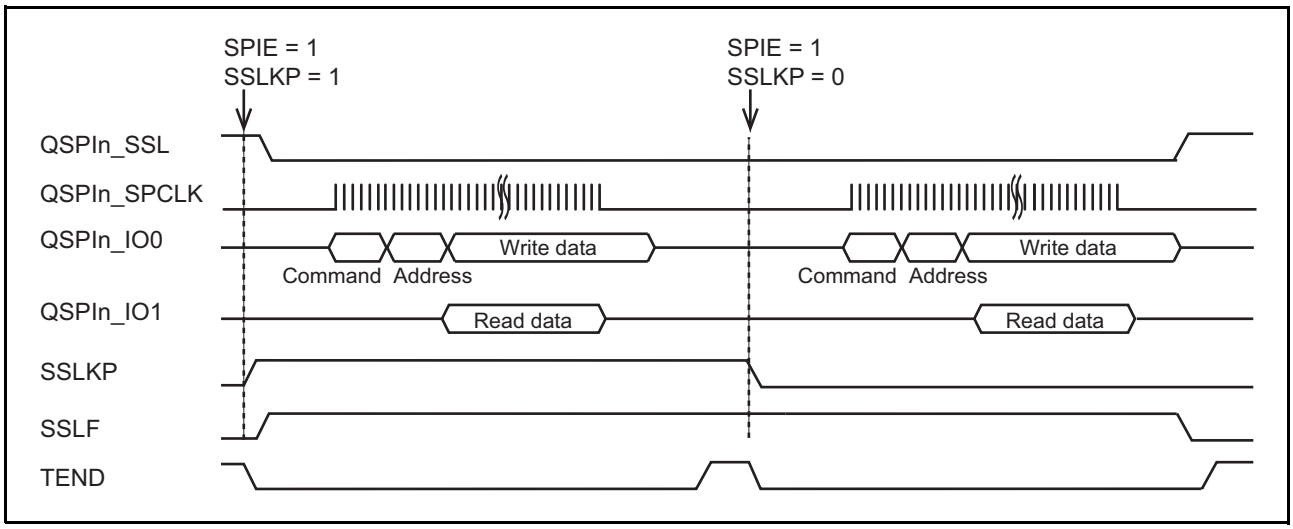


Figure 20.15 Data Transfer Timing using the SSLKP Bit

(4) Initial Setting Flow

An example of an initial setting flow in Manual mode is shown in Figure 20.16.

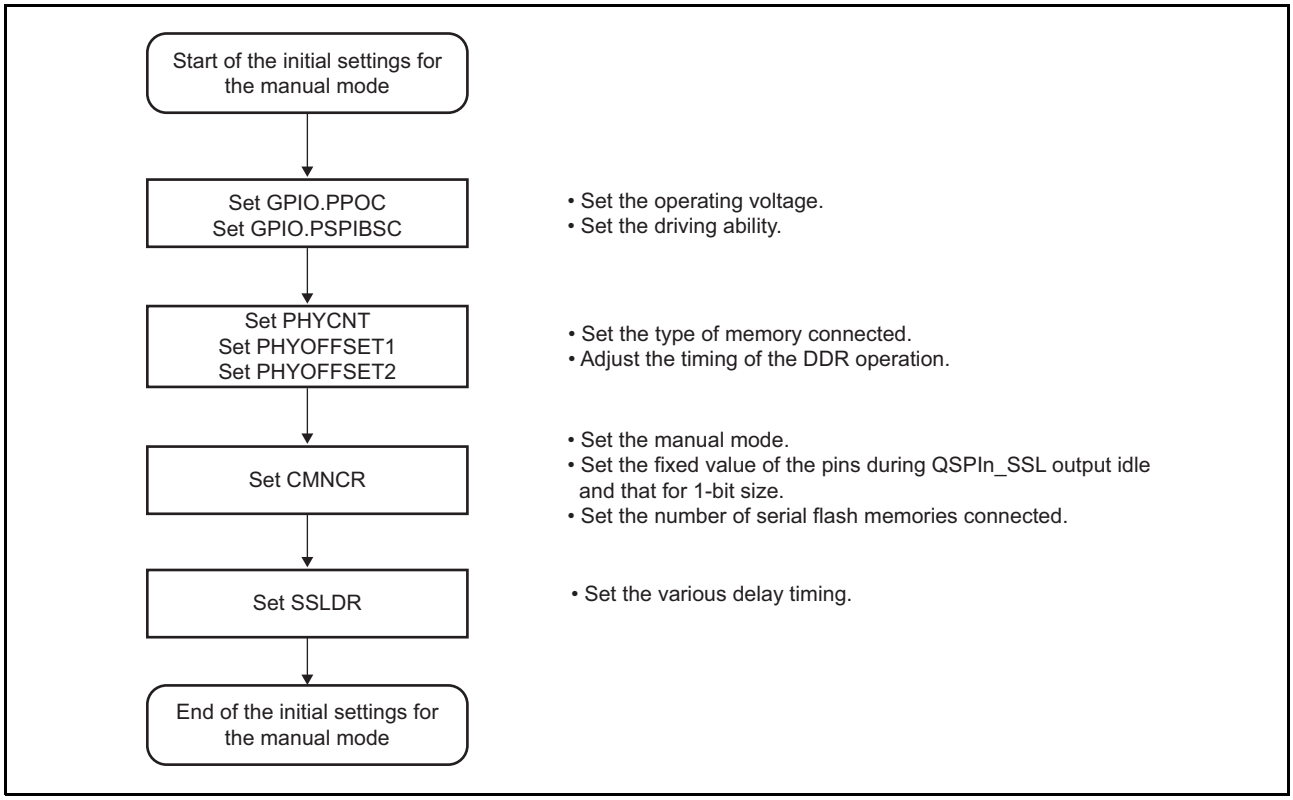


Figure 20.16 Example of Initial Setting Flow in Manual mode

(5) Data Transfer Setting Flow

An example of a data transfer setting flow in manual mode is shown in Figure 20.17.

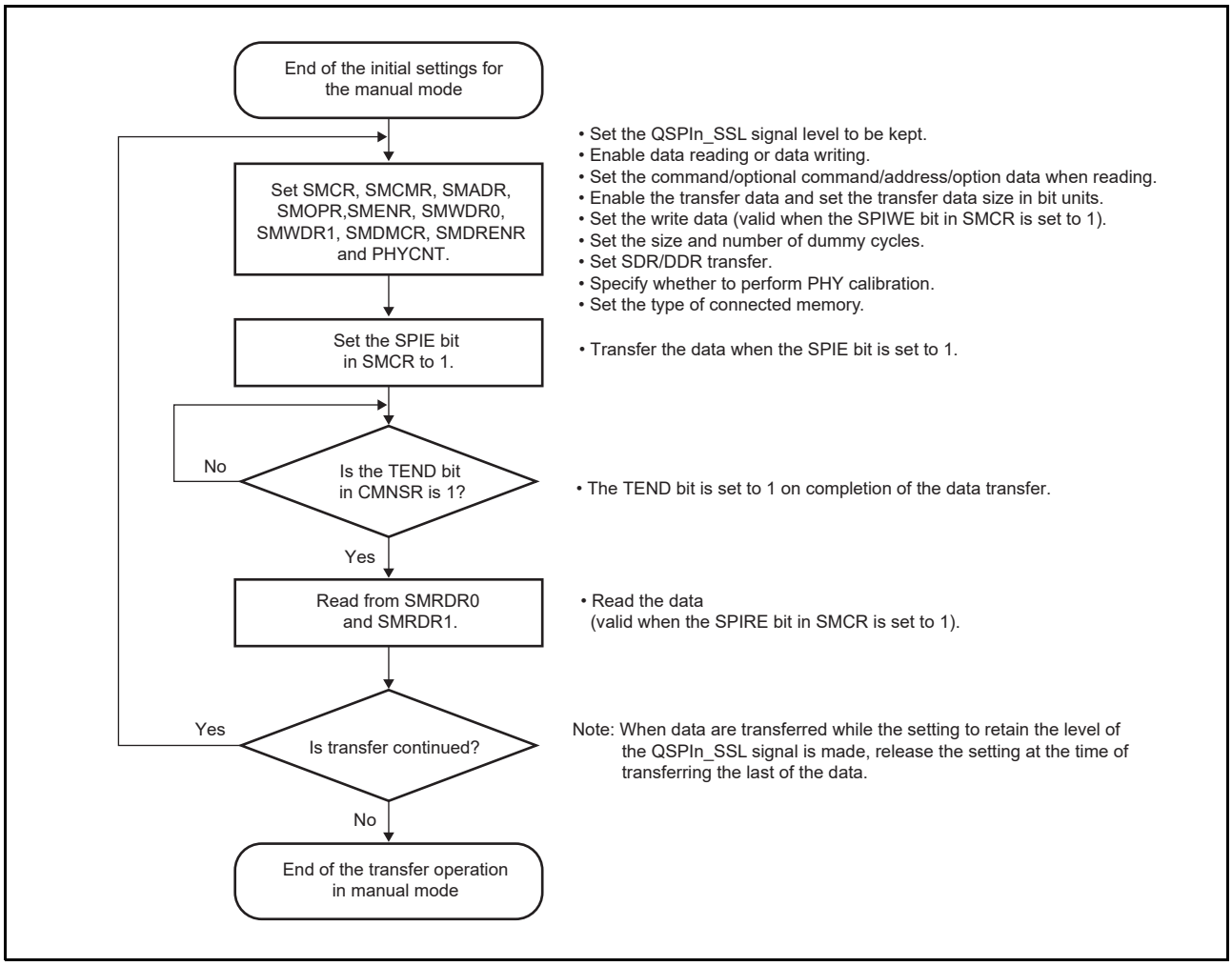


Figure 20.17 Example of a Data Transfer Setting Flow in Manual Mode

20.5.8 Command Sequence

This module can input and output data in the order of command, optional command, address, option data, dummy cycle, and data.

(1) Data Registers

Table 20.4 shows the input and output data.

Table 20.4 Data Registers

Data		External Address Space Read Mode	Manual Mode
Command (8 bits)		CMD[7:0] bits in DRCMR	CMD[7:0] bits in SMCMR
Optional command (8 bits)		OCMD[7:0] bits in DRCMR	OCMD[7:0] bits in SMCMR
Address (32/24 bits)	BSZ[1:0] = 00 (one flash memory connected)	32 bits: DREAR.EAV[6:3 to 0] bits + lower [27 to 24:0] bits of the read address. 24 bits: Lower [23:0] bits of the read address.	32 bits: ADR[31:0] bits in SMADR 24 bits: ADR[23:0] bits in SMADR
	BSZ[1:0] = 01 (two flash memories connected)	32 bits: DREAR.EAV[7:3 to 0] bits + lower [27 to 24:1] bits of the read address. 24 bits: Lower [24:1] bits of the read address.	
Option data (8 bits × 4)		DROPR	SMOPR
Dummy cycle (2 to 20 cycles)		DRDMCR	SMDMCR (only when read)
Transfer data		Normal read: 8, 16, and 32 bits Burst read: 64 × RBURST bits	Read: SMRDR0, SMRDR1 Write: SMWDR0, SMWDR1

Table 20.5 Data Registers (HyperFlash)

Data	External Address Space Read Mode	Manual Mode
CA0 (47 to 40)	DRCMR.CMD[7:5] bits + [31:27] bits of the read address.	SMCMR.CMD[7:5] + SMADR.ADR[31:27]
CA0 (39 to 32)	[26:19] bits of the read address.	ADR[26:19] bits in SMADR
CA1 (31 to 24)	[18:11] bits of the read address.	ADR[18:11] bits in SMADR
CA1 (23 to 16)	[10:3] bits of the read address.	ADR[10:3] bits in SMADR
CA2 (15 to 8)	[15:8] bits in DROPR	[15:8] bits in SMOPR
CA2 (7 to 0)	DROPR [7:3] bits + [2:0] bits of the read address.	SMOPR [7:3] bits + SMADR.ADR[2:0] bits
Dummy cycle (2 to 20 cycles)	DRDMCR	SMDMCR
Transfer data	Normal read: 8, 16, and 32 bits Burst read: 64 × RBURST bits	Read: SMRDR0, SMRDR1 Write: SMWDR0, SMWDR1

(2) Data Enable

In external address space read mode, enabling or disabling transfer of the command, optional command, address, option data, and dummy cycle can be controlled with the CDE, OCDE, ADE[3:0], OPDE[3:0], and DME bits in DRENR, respectively. The number of dummy cycles can be controlled with the data read dummy cycle setting register (DRDMCR).

Similarly, in manual mode, enabling or disabling the command, optional command, address, option data, dummy cycle, and transfer data can be controlled with the CDE, OCDE, ADE[3:0], OPDE[3:0], DME, and SPIDE[3:0] bits in SMENR, respectively. However, disabling all the above parameters is prohibited in manual mode. At least one of them except dummy cycle must be enabled. The number of dummy cycles can be controlled with the Manual mode dummy cycle setting register (SMDMCR).

For the address and option data in external address space read mode and the address, option data, and transfer data in manual mode, the enable bit setting allowed is determined according to the transfer data size. For the allowed setting combinations of the enable bits and transfer data size, refer to the description of the pertinent register.

If data is disabled, that data is skipped, and input and output of the next data is carried out. The command, optional command, address, and option data are always output. During dummy cycles, the state of the used pins is Hi-Z. In external address space read mode, data is always input; and in manual mode, input and output of data is determined based on the settings of the SPIRE and SPIWE bits in SMCR.

There are some restrictions on dummy cycle insertion; refer to the description of the DME bits in DRENR and SMENR for details.

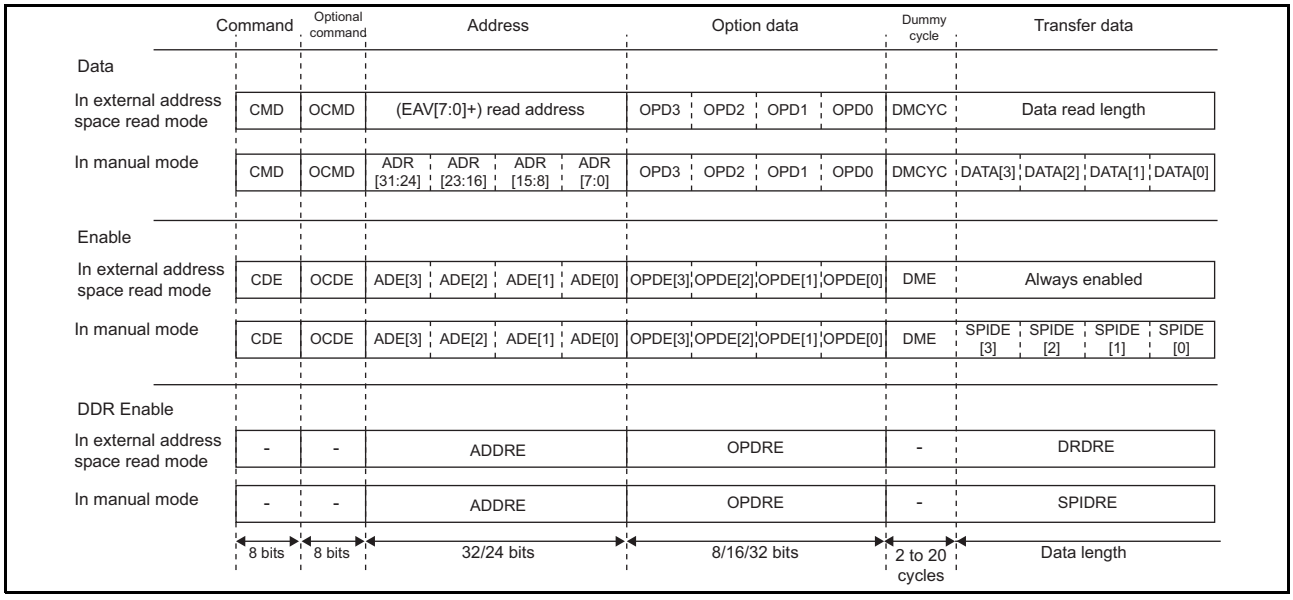


Figure 20.18 Data and Enable (Serial Flash)

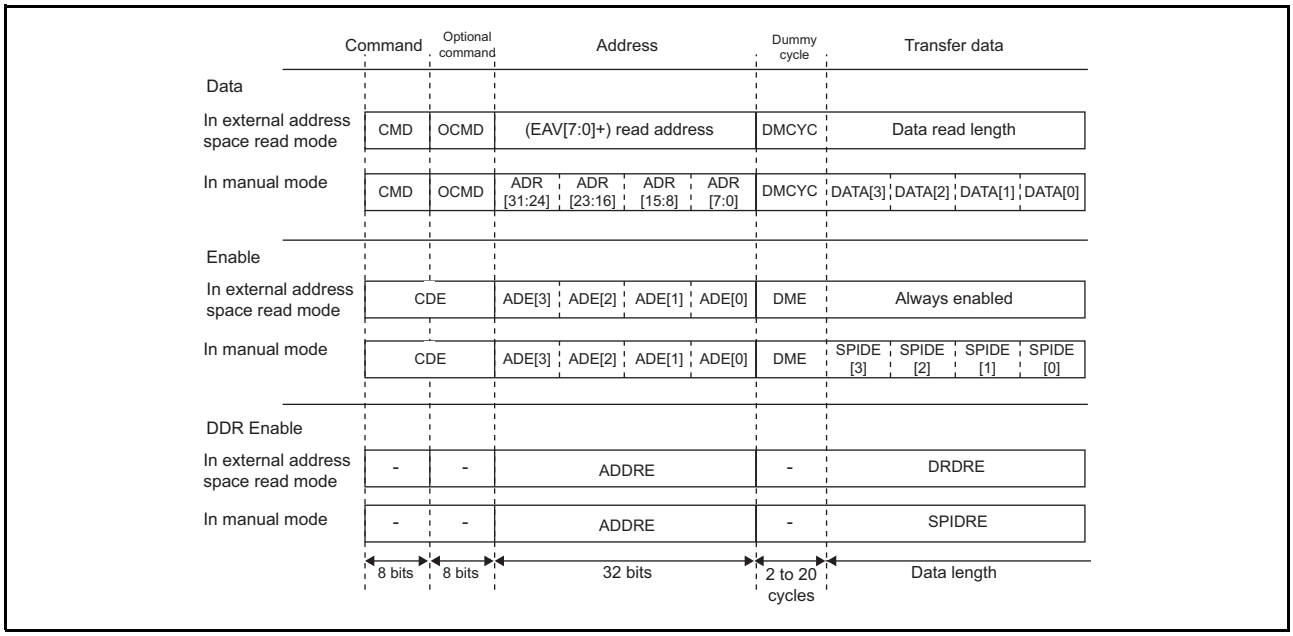


Figure 20.19 Data and Enable (Octal-SPI Flash Memory Protocol Mode)

When the HyperFlash memory is connected to this LSI, the CDE, OCDE, ADE[3:0], OPDE[3:0], and DME bits in the DRENR register and the HYPE[2:0], ADDRE, and OPDRE bits in the DRDRENR register should be set to "enabled" in external address space read mode. In Manual mode, the CDE, OCDE, ADE[3:0], OPDE[3:0], and DME bits in the SMENR register and the HYPE[2:0], ADDRE, OPDRE, and DRDRE bits in the SMDRENR should be set to "enabled".

Table 20.6 Enable Register (HyperFlash)

External Address Space Read Mode			Manual Mode		
Register	Bit	Setting Value	Register	Bit	Setting Value
DRENR	CDE	1	SMENR	CDE	1
	OCDE	1		OCDE	1
	ADE[3:0]	0100		ADE[3:0]	0100
	OPDE[3:0]	0000		OPDE[3:0]	0000
	DME	1		DME	Read: 1 Write: 0
				SPIDE[3:0]	
				16 bits	1000
				32 bits	1100
				64 bits	1111
DRDRENR	HYPE[2:0]	101	SMDRENR	HYPE[2:0]	101
	ADDRE	1		ADDRE	1
	OPDRE	0		OPDRE	0
	DRDRE	1		SPIDRE	1

Table 20.7 Enable Register (Octal-SPI Flash Memory Protocol Mode)

External Address Space Read Mode			Manual Mode		
Register	Bit	Setting Value	Register	Bit	Setting Value
DRENR	CDE	1	SMENR	CDE	1
	OCDE	0		OCDE	0
	ADE[3:0]	1100		ADE[3:0]	1100
	OPDE[3:0]	0000		OPDE[3:0]	0000
	DME	1		DME	Read: 1 Write: 0
SMENR	CDE	0		SPIDE[3:0]	
				16 bits	1000
				32 bits	1100
				64 bits	1111
DRDRENR	HYPE[2:0]	100, 101	SMDRENR	HYPE[2:0]	100, 101
	ADDRE	1		ADDRE	1
	OPDRE	0		OPDRE	0
	DRDRE	1		SPIDRE	1

Note: When only commands are to be transferred, set DRDRENR.HYPE or SMDRENR.HYPE to 100.

(3) Bit Size

In external address space read mode, the size of the command, optional command, address, option data, and the read data in bit units is respectively controlled with the CDB[1:0], OCDB[1:0], ADB[1:0], OPDB[1:0], and DRDB[1:0] bits in DRENR.
Similarly, in manual mode, the size of the command, optional command, address, option data, and read write data in bit units is controlled with the CDB[1:0], OCDB[1:0], ADB[1:0], OPDB[1:0], and SPIDB[1:0] bits in SMENR.

(a) 1-bit Size

When the size is set to 1 bit, QSPI0_IO1 and QSPI1_IO1 pins will be the input pins and QSPI0_IO0 and QSPI1_IO0 pins will be the output pins. QSPI0_IO2, QSPI1_IO2, QSPI0_IO3, and QSPI1_IO3 pins are not used.
Figure 20.20 and Figure 20.21 show the transfer format examples.

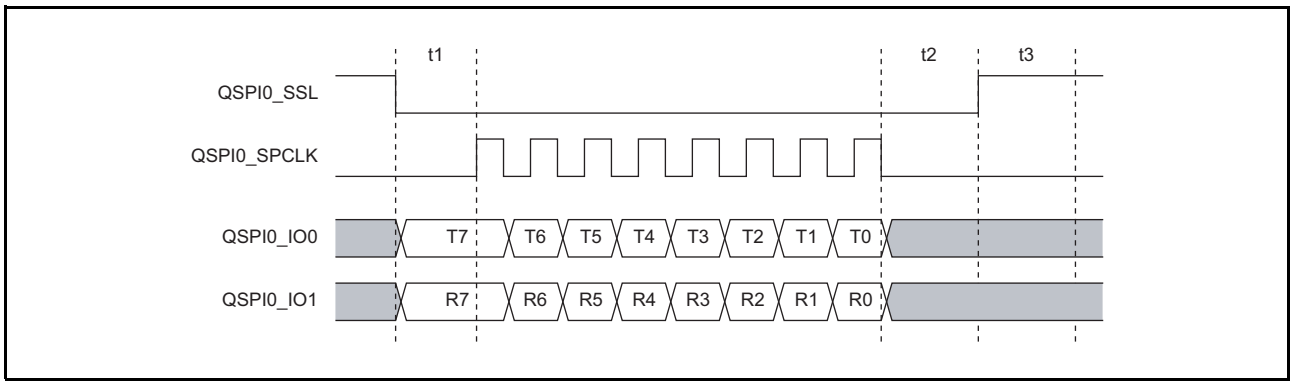


Figure 20.20 Transfer Format Example with 1-Bit Data Size and One Serial Flash Memory Connected

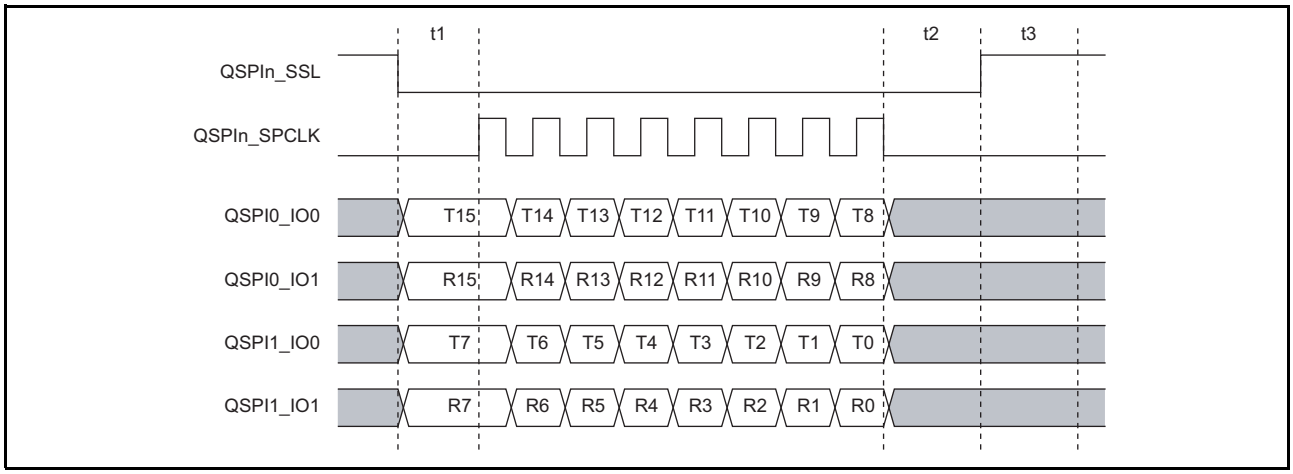


Figure 20.21 Transfer Format Example with 1-Bit Data Size and Two Serial Flash Memories Connected

(b) 4-bit Size (Serial Flash)

When the size is set to 4 bits, QSPI0_IO0, QSPI1_IO0, QSPI0_IO1, QSPI1_IO1, QSPI0_IO2, QSPI1_IO2, QSPI0_IO3, and QSPI1_IO3 pins will be either the input pins or the output pins.
Figure 20.22 and Figure 20.23 show the transfer format examples.

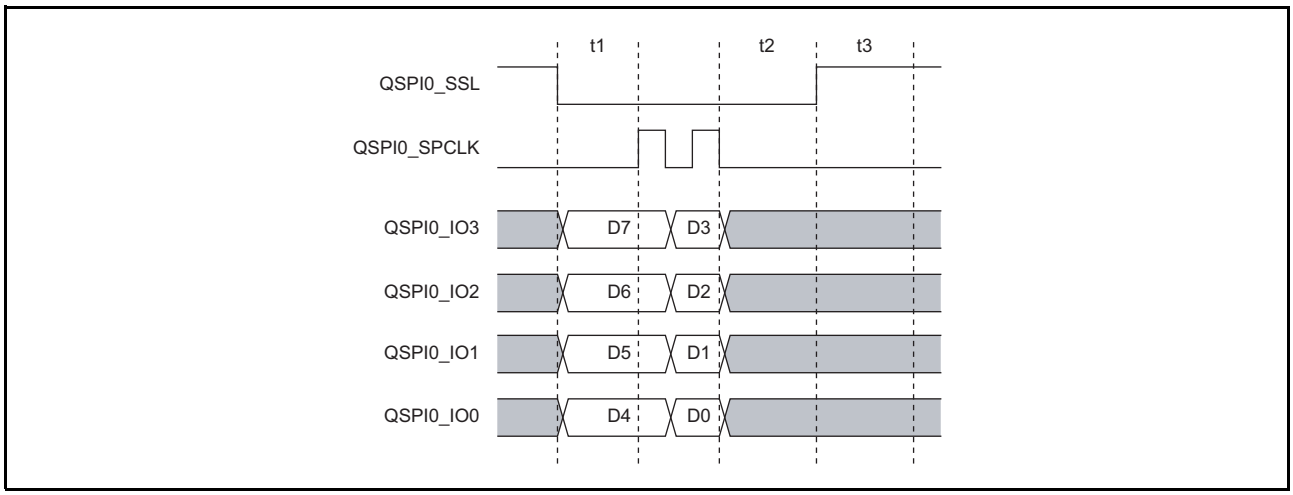


Figure 20.22 Transfer Format Example with 4-Bit Data Size and One Serial Flash Memory Connected

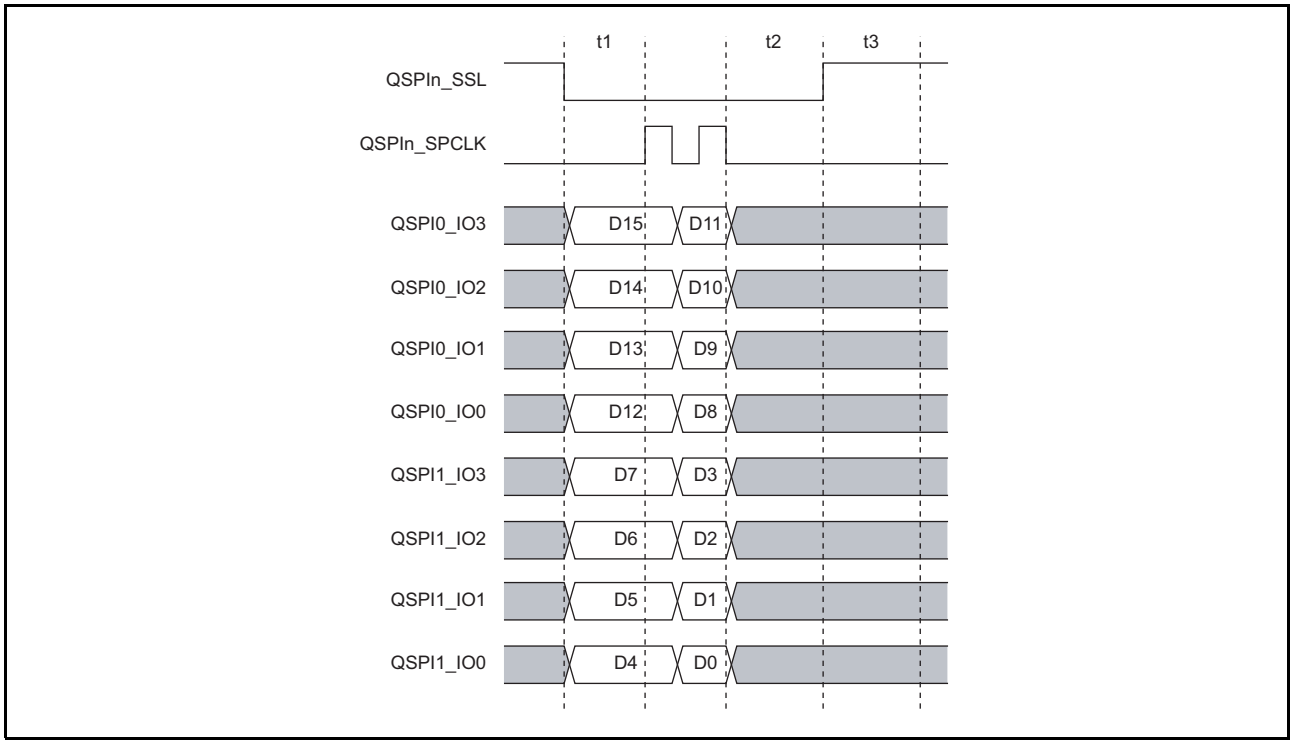


Figure 20.23 Transfer Format Example with 4-Bit Data Size and Two Serial Flash Memories Connected

(c) 8-bit Size (HyperFlash)

When the HyperFlash memory is connected to this LSI, QSPI0_SPCLK and QSPI1_SPCLK work as differential clock pins, QSPI1_SSL works as the read data strobe (RDS) input pin, and QSPI0_IO0, QSPI1_IO0, QSPI0_IO1, QSPI1_IO1, QSPI0_IO2, QSPI1_IO2, QSPI0_IO3, and QSPI1_IO3 work as either input pins or output pins.
Figure 20.24 and Figure 20.25 show the transfer format examples.

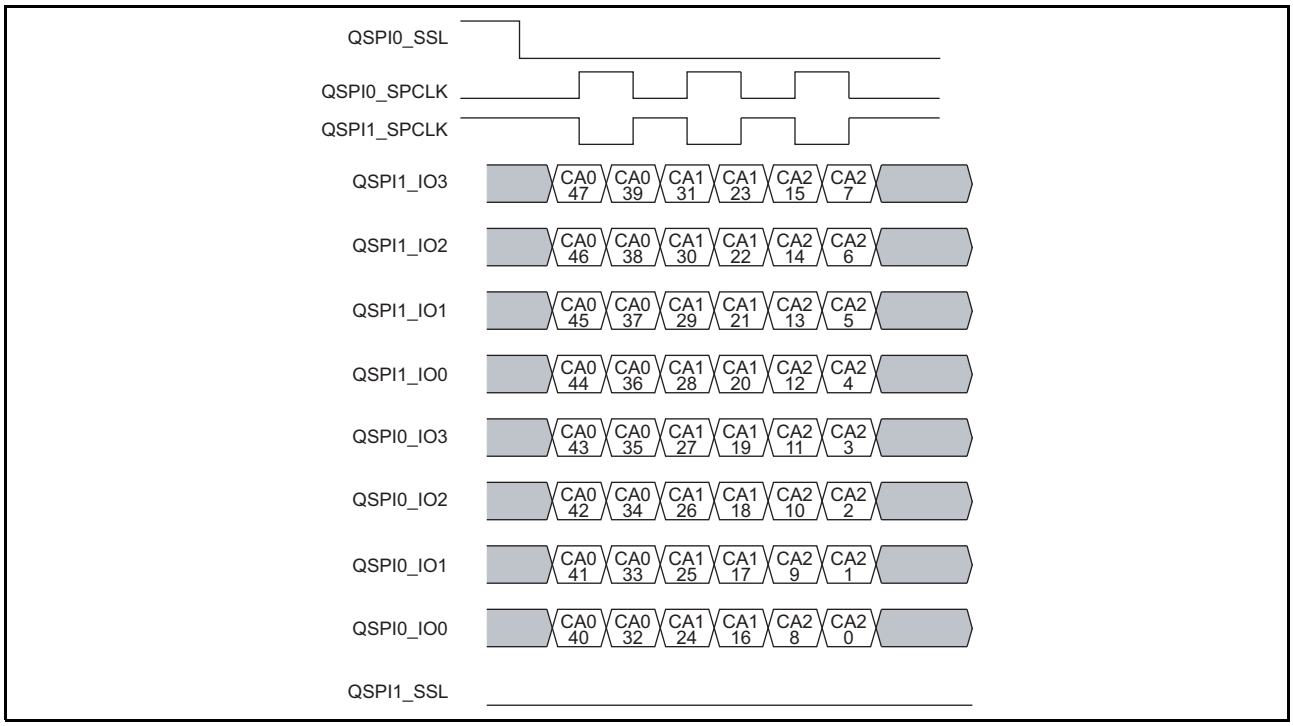


Figure 20.24 Transfer Format Example in the HyperFlash Command and Address Phase

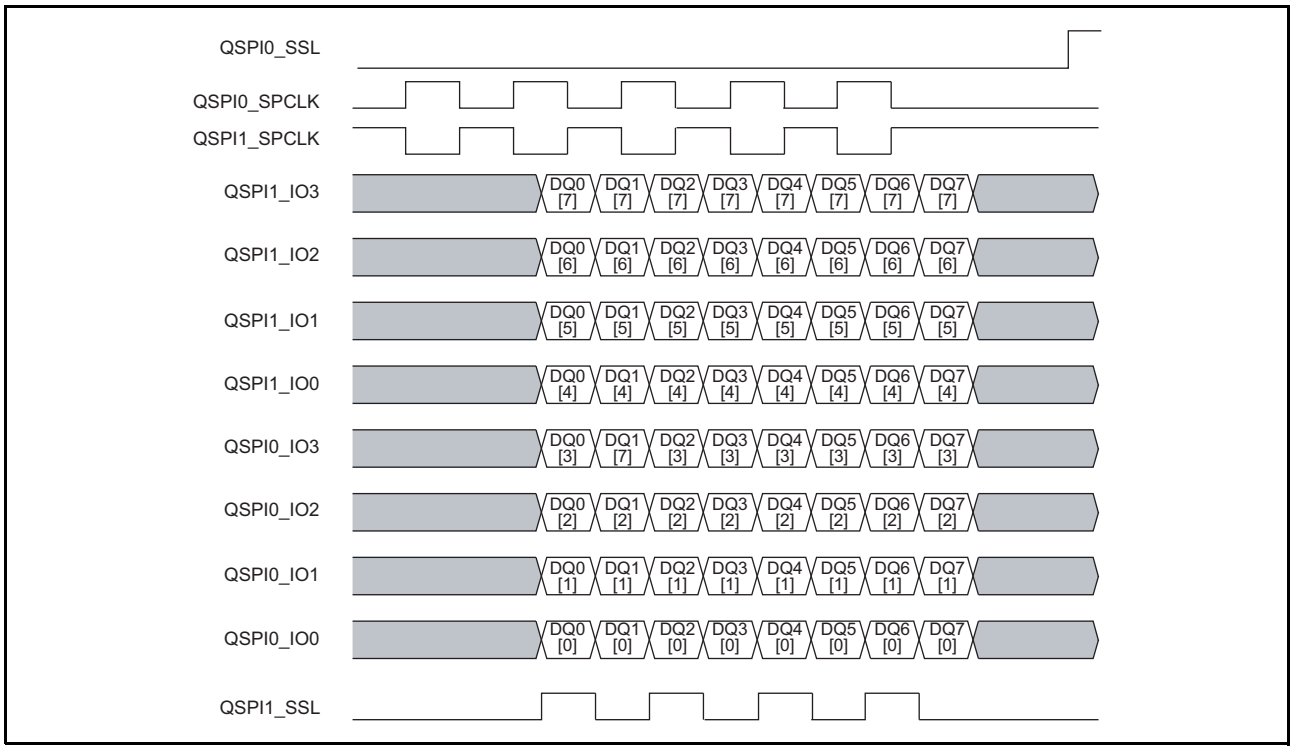


Figure 20.25 Transfer Format Example in the HyperFlash Read Phase

20.5.9 Data Pin Control

With this module, the status of pins can be automatically changed based on the data size to be used and the read/write settings. The pin status during the QSPIn_SSL negation can be set with the MOIIO3, MOIIO2, MOIIO1, and MOIIO0 bits in CMNCR. The QSPIn_SSL and QSPIn_SPCLK pins are always output pins.
The status of respective pins is specified in Table 20.8 to Table 20.11.
When using only a single serial flash memory device, note that connecting the pins for channel 1 to the serial flash memory is prohibited.

Table 20.8 Pin Status (1)

		QSPIn_SSL Assertion	
		Command, Optional Command, Address, Option Data	
Pin	QSPIn_SSL Negation	1-bit Size	4-bit Size
QSPI0_IO0, QSPI1_IO0	MOIIO0 bit value	Output	Output
QSPI0_IO1, QSPI1_IO1	MOIIO1 bit value	Hi-Z	Output
QSPI0_IO2, QSPI1_IO2	MOIIO2 bit value	IO2FV bit value	Output
QSPI0_IO3, QSPI1_IO3	MOIIO3 bit value	IO3FV bit value	Output

Table 20.9 Pin Status (2)

	Transfer Data			
	External Address Space Read Mode		SPI Operation	
			SPIRE Bit = 1, SPIWE Bit = 0	
Pin	1-bit Size	4-bit Size	1-bit Size	4-bit Size
QSPI0_IO0, QSPI1_IO0	IO0FV bit value	Input	IO0FV bit value	Input
QSPI0_IO1, QSPI1_IO1	Input	Input	Input	Input
QSPI0_IO2, QSPI1_IO2	Hi-Z	Input	Hi-Z	Input
QSPI0_IO3, QSPI1_IO3	Hi-Z	Input	Hi-Z	Input

Table 20.10 Pin Status (3)

	Transfer Data			
	SPI Operation			
	SPIRE Bit = 0, SPIWE Bit = 1		SPIRE Bit = 1, SPIWE Bit = 1	
Pin	1-bit Size	4-bit Size	1-bit Size	4-bit Size
QSPI0_IO0, QSPI1_IO0	Output	Output	Output	Setting prohibited
QSPI0_IO1, QSPI1_IO1	Hi-Z	Output	Input	Setting prohibited
QSPI0_IO2, QSPI1_IO2	IO2FV bit value	Output	IO2FV bit value	Setting prohibited
QSPI0_IO3, QSPI1_IO3	IO3FV bit value	Output	IO3FV bit value	Setting prohibited

Table 20.11 Pin Status (4)

Pin	Dummy Cycle
QSPI0_IO0, QSPI1_IO0	IO0FV bit value
QSPI0_IO1, QSPI1_IO1	Hi-Z
QSPI0_IO2, QSPI1_IO2	Hi-Z
QSPI0_IO3, QSPI1_IO3	Hi-Z

20.5.10 QSPIn_SSL Pin Control

Negation conditions of the QSPIn_SSL pin are as follows.

(1) External Address Space Read Mode

(a) Normal read operation (RBE bit in DRCR = 0)

QSPIn_SSL negated after completing the data transfer and t2 cycle.

(b) Burst read without automatic QSPIn_SSL negation (RBE bit in DRCR = 1, SSLE bit in DRCR = 0)

QSPIn_SSL negated after completing the data transfer and t2 cycle.

(c) Burst read with automatic QSPIn_SSL negation (RBE bit in DRCR = 1, SSLE bit in DRCR = 1)

- QSPIn_SSL negated after t2 cycle when the read address is not continuous with the previously read address
- QSPIn_SSL negated after the SSLN bit in DRCR is set to 1

(2) Manual Mode

(a) QSPIn_SSL pin assertion not retained (SSLKP bit in SMCR = 0)

QSPIn_SSL negated after completing the data transfer and t2 cycle.

(b) QSPIn_SSL pin assertion retained (SSLKP bit in SMCR = 1)

QSPIn_SSL not negated.
When to be negated, data should be transferred after setting the SSLKP bit to 0.

Note: When HyperFlash memory is connected or a connection is made in Octal-SPI flash memory protocol mode, the QSPIn_SSL pin is used as an input terminal for data strobe.

20.5.11 QSPIn_SPCLK Pin Control

QSPI0_SPCLK and QSPI1_SPCLK are output according to the following settings.

Table 20.12 QSPIn_SPCLK Output

Pin	HyperFlash connected. (CMNCR.BSZ = 01, PHYCNT.PHYMEM = 11)	One serial flash memory connected. (CMNCR.BSZ = 00, PHYCNT.PHYMEM = 00, 01	Two serial flash memories connected. (CMNCR.BSZ = 01, PHYCNT.PHYMEM = 00, 01
QSPI0_SPCLK	Output	Output	Output
QSPI1_SPCLK	Output (Complementary output of QSPI0_SPCLK)	Hi-Z	Output

20.5.12 Flags

This module has two flag bits SSLF and TEND in CMNSR. These bits are read-only bits.

(1) SSLF Bit

This bit indicates the QSPIn_SSL pin status. The status is 1 when the QSPIn_SSL is asserted, and the status is 0 when the QSPIn_SSL is negated.

(2) TEND Bit

This bit indicates whether transfer of data is in progress or the transfer of data has ended.
During t1 time period, data transfer, t2 time period, t3 time period, and waiting for read access by burst read and QSPIn_SSL automatic negation, the TEND bit is read as 0 to indicate that the transfer of data is in progress. Otherwise, the TEND bit is read as 1 to indicate that transfer of data has ended.

(3) Register Re-writing Timing

The status of the TEND bit determines the rewritable registers.
The registers which can be written to, except the SSLN bit in DRCR, should be modified when TEND = 1. Read SMRDR0 and SMRDR1 when TEND = 1. CMNSR can always be read.

20.5.13 Write Buffer Operation

This module uses the read cache space also as the write buffer space during write operation.
The write buffer improves the performance of write access.

Table 20.13 Write Buffer Space

Address	Access Size
H'1F80_8000 to H'1F80_80FF	4, 8, 16, 32, or 64 bytes

Note: Be sure to sequentially access this space beginning from its start address with the transfer size set to eight bytes.
If access is non-sequential or random, correct operation is not guaranteed.
After an operation in Manual mode or with the use of the cache area as the write buffer, clearing of this cache area by writing 1 to the RCF bit is required.

Figure 20.26 shows the procedure for using the write buffer. For the initial settings in Manual mode, see Figure 20.16.

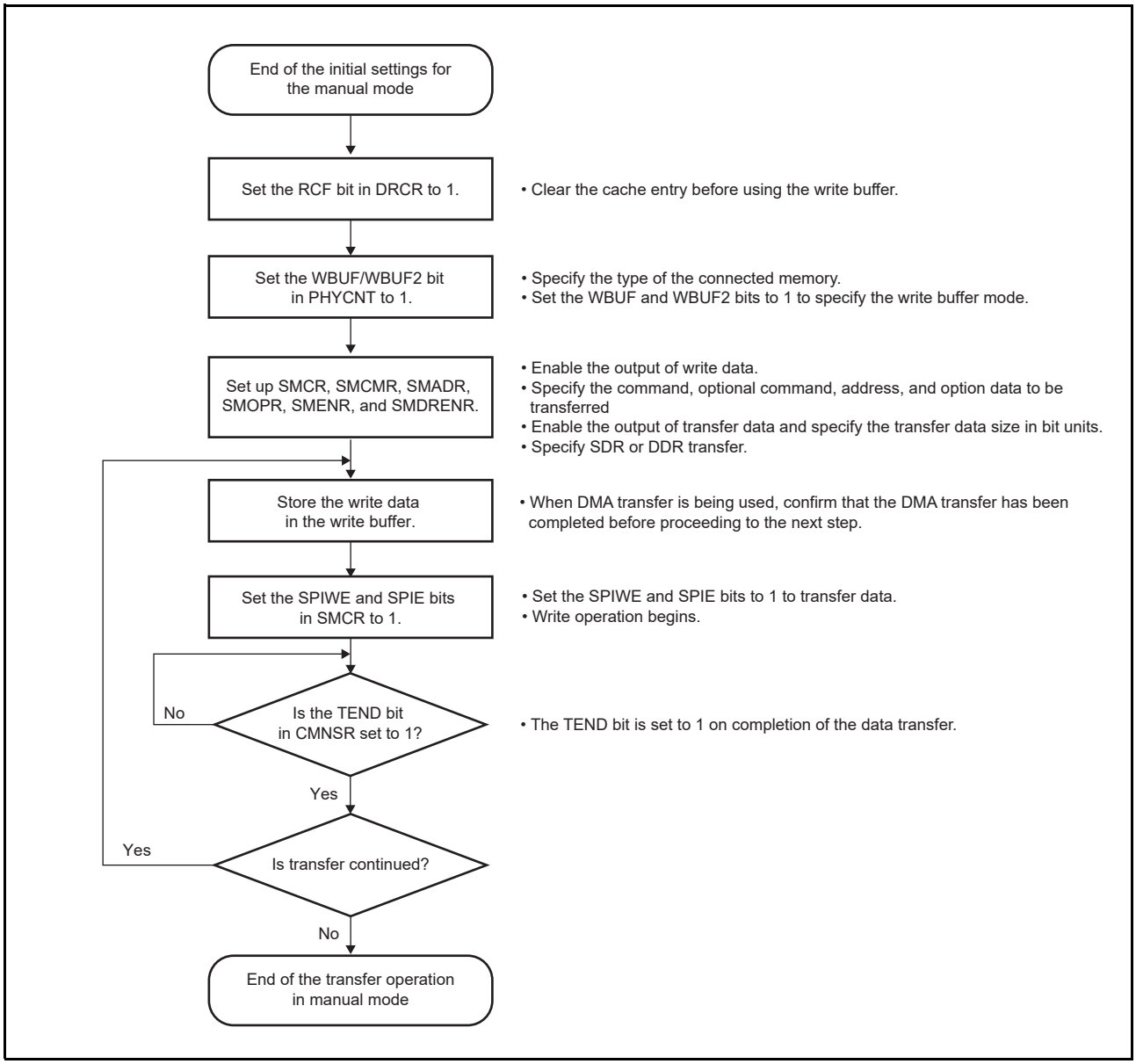


Figure 20.26 Procedure for Using the Write Buffer

20.5.14 Data Alignment in Octal-SPI Flash Memory Protocol Mode

When this LSI chip is connected in Octal-SPI flash memory protocol mode, two data alignment types are available in this module.

Figure 20.27 shows the sequential alignment type and Figure 20.28 shows the alternative alignment type.

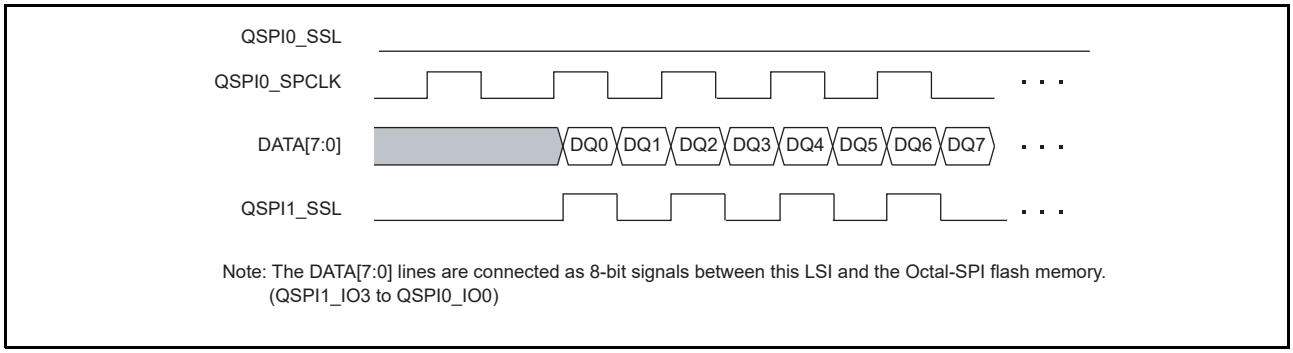


Figure 20.27 Sequential Alignment in Octal-SPI Flash Memory Protocol Mode (PHYCNT.OCTA = 10)

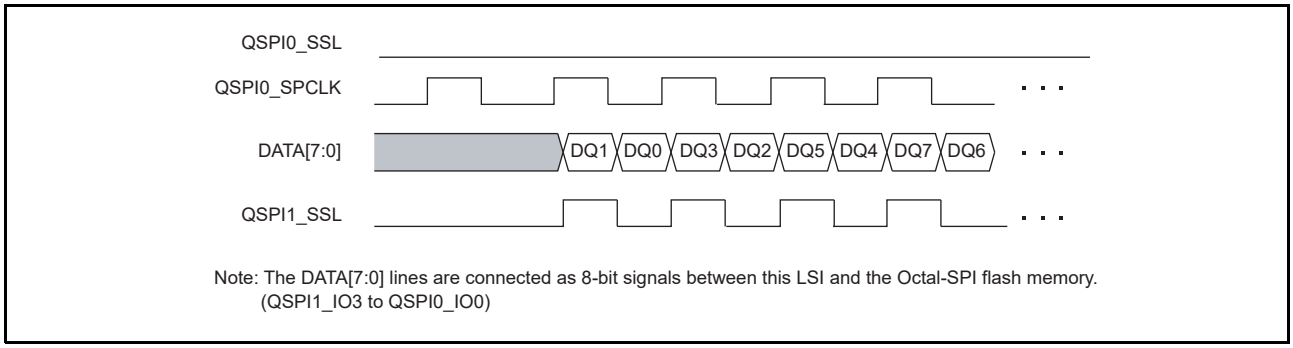


Figure 20.28 Alternative Alignment in Octal-SPI Flash Memory Protocol Mode (PHYCNT.OCTA = 01)

20.5.15 Supported Protocol for Serial Flash Memory

The protocol for serial flash memory that this LSI supports is indicated in Table 20.14.

Table 20.14 Supported Protocol for Serial Flash Memory

Bit Width (Command-Address-Data)	Command	Address	Data
1-0-0	SDR	NA	NA
1-0-1	SDR	NA	SDR
1-1-0	SDR	SDR	NA
1-1-1	SDR	SDR	SDR
	SDR	DDR	DDR
1-1-4	SDR	SDR	SDR
1-4-4	SDR	SDR	SDR
	SDR	DDR	DDR

20.5.16 Supported Protocol for Octal-SPI Flash Memory

The protocol for Octal-SPI flash memory that this LSI supports is indicated in Table 20.15.

Table 20.15 Supported Protocol for Octal-SPI Flash Memory

Bit Width (Command-Address-Data)	Command	Address	Data
1-0-0	SDR	NA	NA
1-0-1	SDR	NA	SDR
1-1-0	SDR	SDR	NA
1-1-1	SDR	SDR	SDR
8-0-0	SDR	NA	NA
	DDR	NA	NA
8-0-8	DDR	NA	DDR
8-8-0	DDR	DDR	NA
8-8-8	DDR	DDR	DDR

20.5.17 Timing Adjustment

When data is transferred with serial flash memory connected, adjustment of the timing between the clock operation and the acquisition of input data is required. When a connection is made in Octal-SPI flash memory protocol mode with the data strobe to be used or HyperFlash memory is connected, the timing of the acquisition of input data is automatically adjusted.

Figure 20.29 (1) shows the flow for timing adjustment when a serial flash memory (SDR mode except in the case of Octal-SPI flash memory) is connected.

Execute this flow before changing the setting of SCLKSEL.SPICR[1:0] from its initial value when the serial flash memory is connected (SDR mode). Execution of this flow at regular intervals is not required.

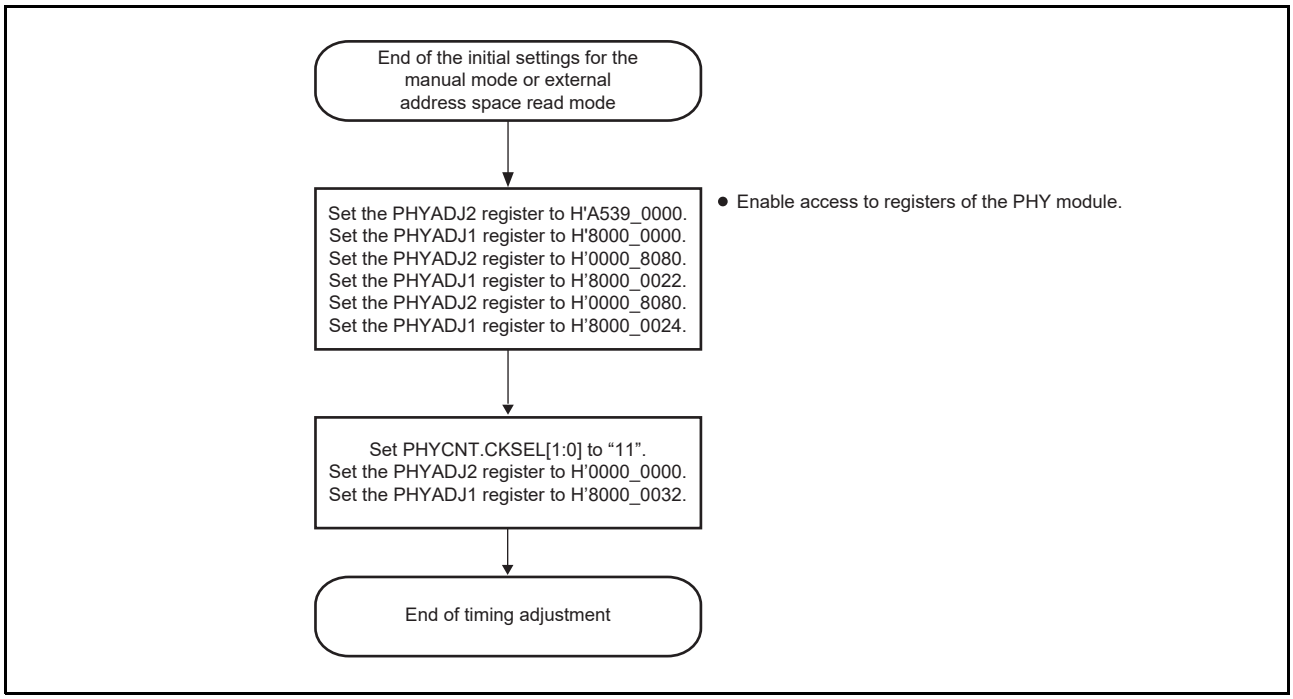


Figure 20.29 (1) Flow of Timing Adjustment when the Serial Flash Memory is Connected (SDR Mode Except in the Case of Octal-SPI Flash Memory)

Figure 20.29 (2) shows the timing adjustment flow for serial flash (DDR mode) connection.
Since the set value acquired in this timing adjustment flow is affected by voltage and temperature variation, this flow should be performed periodically.

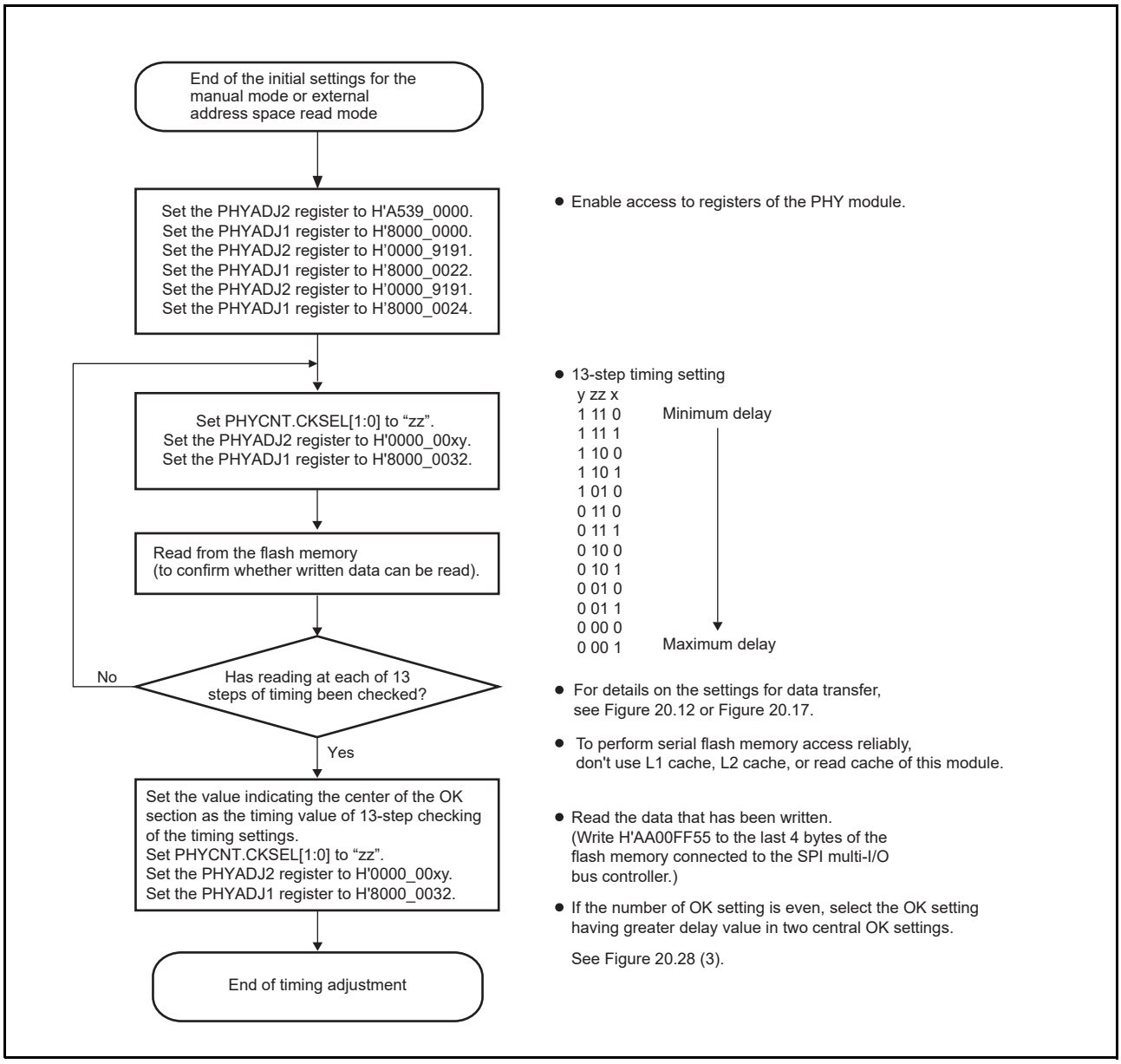
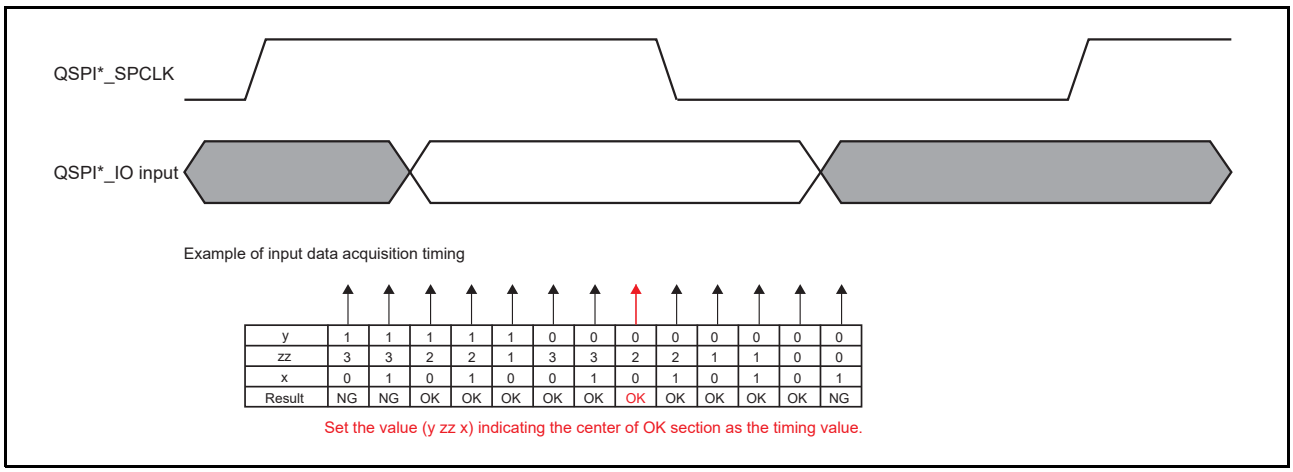


Figure 20.29 (2) Flow of Timing Adjustment when the Serial Flash Memory is Connected (DDR Mode)

Figure 20.29 (3) shows an example of timing setting for serial flash (DDR mode) connection.



y	1	1	1	1	1	0	0	0	0	0	0	0	0
zz	3	3	2	2	1	3	3	2	2	1	1	0	0
x	0	1	0	1	0	0	1	0	1	0	1	0	1
Result	NG	NG	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	NG

Figure 20.29 (3) Example of Timing Adjustment when the Serial Flash Memory is Connected (DDR Mode)

20.5.18 Data Alignment

Data alignments in external address space read mode and Manual mode are shown in Figure 20.30 and Figure 20.31, respectively. When two serial flash memory devices are connected, the addresses of one device connected to pins QSPI0_IO0 to QSPI0_IO3 are treated as 2n and those of the other device connected to pins QSPI1_IO0 to QSPI1_IO3 are treated as 2n + 1. Be sure that access is in at least word units. Access in byte units is not supported.

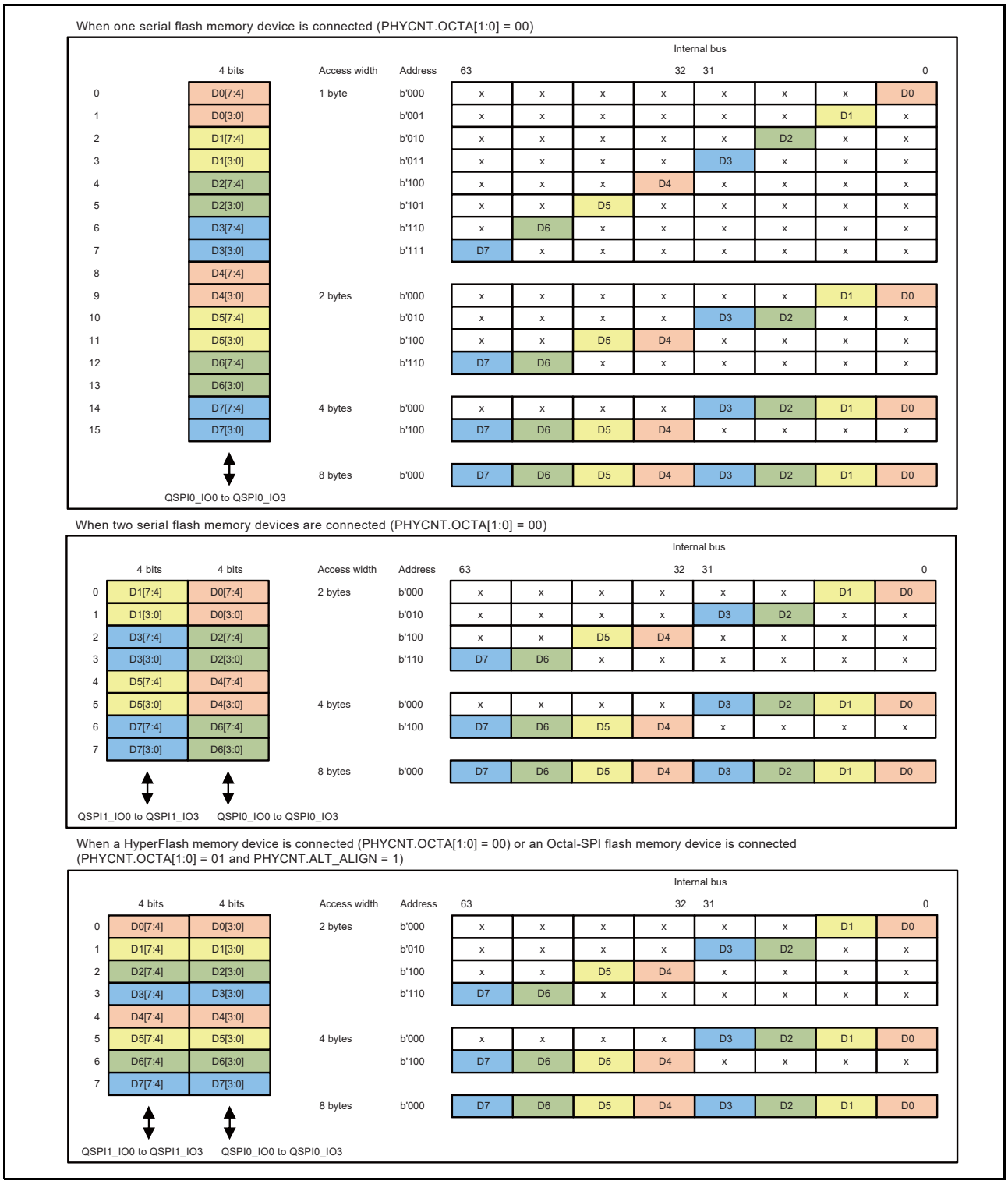
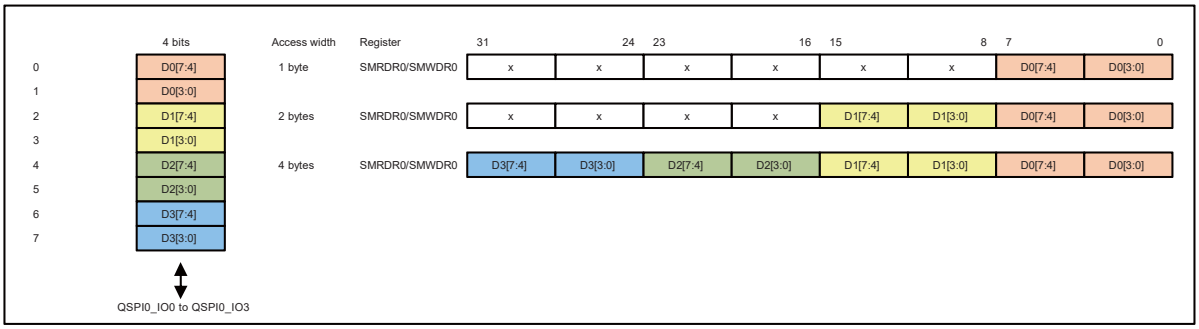
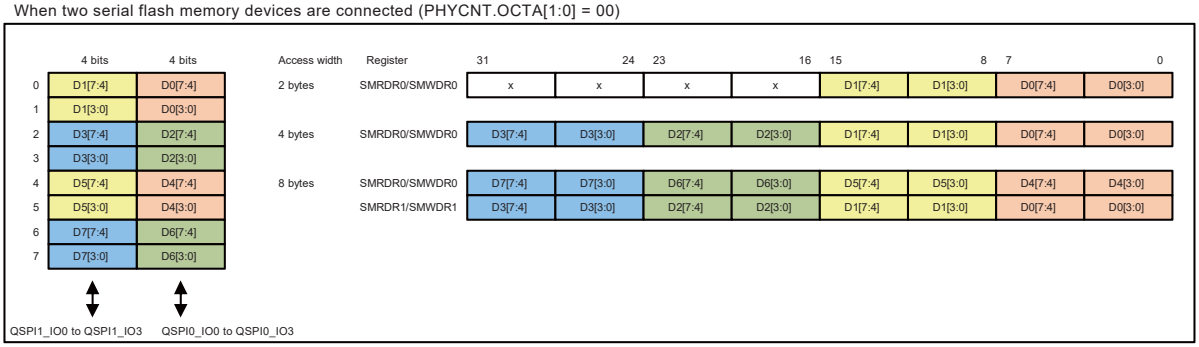


Figure 20.30 Data Alignment in External Address Space Read Mode

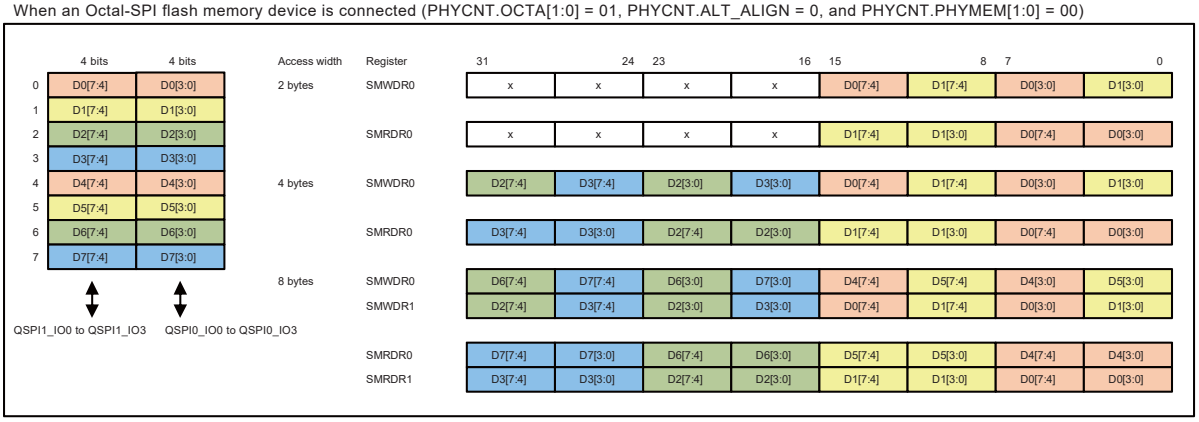
When one serial flash memory device is connected (PHYCNT.OCTA[1:0] = 00)



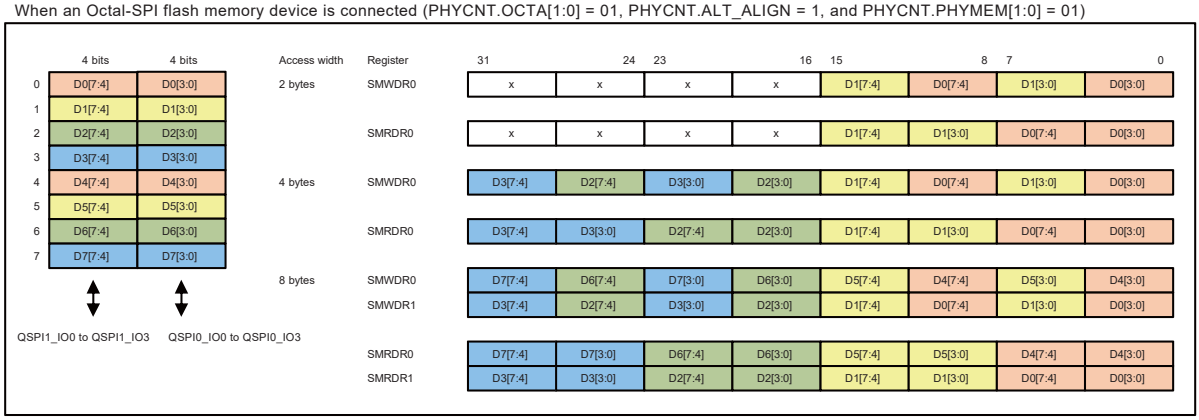
When two serial flash memory devices are connected (PHYCNT.OCTA[1:0] = 00)



When an Octal-SPI flash memory device is connected (PHYCNT.OCTA[1:0] = 01, PHYCNT.ALT_ALIGN = 0, and PHYCNT.PHYMEM[1:0] = 00)



When an Octal-SPI flash memory device is connected (PHYCNT.OCTA[1:0] = 01, PHYCNT.ALT_ALIGN = 1, and PHYCNT.PHYMEM[1:0] = 01)



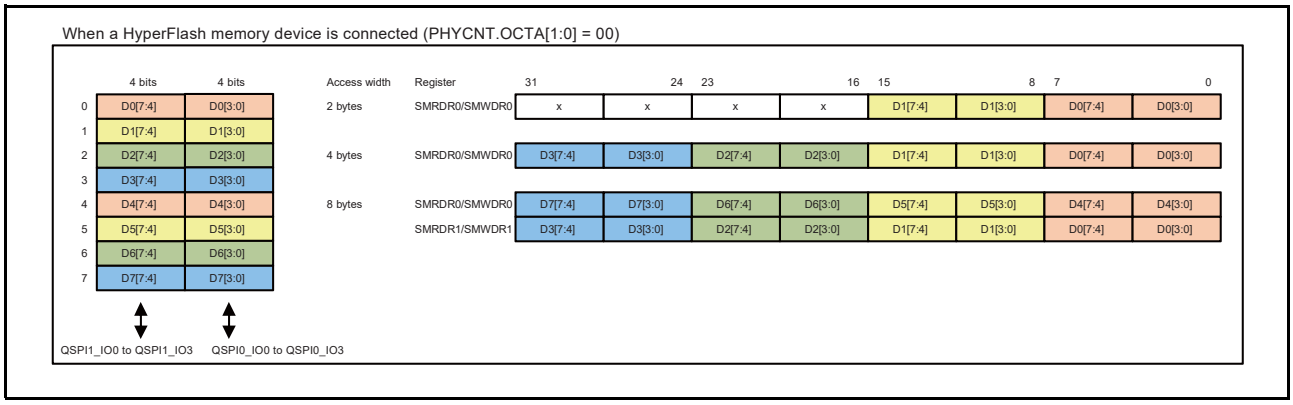


Figure 20.31 Data Alignment in Manual Mode

20.6 Usage Notes

20.6.1 Transfer to read data while the signal on the QSPIn_SSL pin is de-asserted

Set the SMENR.SPIDE[3:0] bits to 1100 or 1111 when transfer only for reading data is to proceed.
Transfer will not proceed normally if the setting of the SMENR.SPIDE[3:0] bits is 1000.

20.6.2 Notes on Starting Transfer from the QSPIn_SSL Retained State in Manual Mode

Be sure to set the SPIWE bit in the SMCR register to 1 when the transfer of a command, optional command, address, or option data is started while the QSPIn_SSL pin is being asserted in Manual mode.
In addition, while the QSPIn_SSL signal is being asserted in Manual mode, the setting to start data transfer for reading is prohibited.

20.6.3 Handling of RPC_RESET#

The state of the RPC_RESET# pin can be changed in the following two ways.

1. Use the RSTEN and RSTVAL bits in the PHYINT register.
   The state of the RPC_RESET# pin depends on the value of the RETVAL bits in the PHYINT register.
2. Activate the reset to this module.
   While a power-on reset or a software reset is applied to this LSI, the signal on the RPC_RESET# pin goes low.
   When the reset is deactivated, the signal on the RPC_RESET# pin goes high.

20.6.4 Software Reset after Release from the Module Stop State

Continue applying a software reset to this module even after the module is released from the module stop state. Note that the signal on the RPC_RESET# pin is low while a software reset is active. Be sure to satisfy the necessary reset period and accessible period for the connected device after a reset.

20.6.5 Writing Data in Octal-SPI Flash Memory Protocol Mode

When writing data in Octal-SPI flash memory protocol mode, the data to be written must be correctly aligned.
For details, refer to section 20.4.15, Manual Mode Write Data Register 0 (SMWDR0), and section 20.4.16, Manual Mode Write Data Register 1 (SMWDR1).

21. HyperBus™ Controller

The HyperBus™ controller enables the connection of HyperFlash™ and HyperRAM™ memory to this LSI chip.

Note: HyperBus™, HyperFlash™ and HyperRAM™ are trademarks of Cypress Semiconductor Corporation.

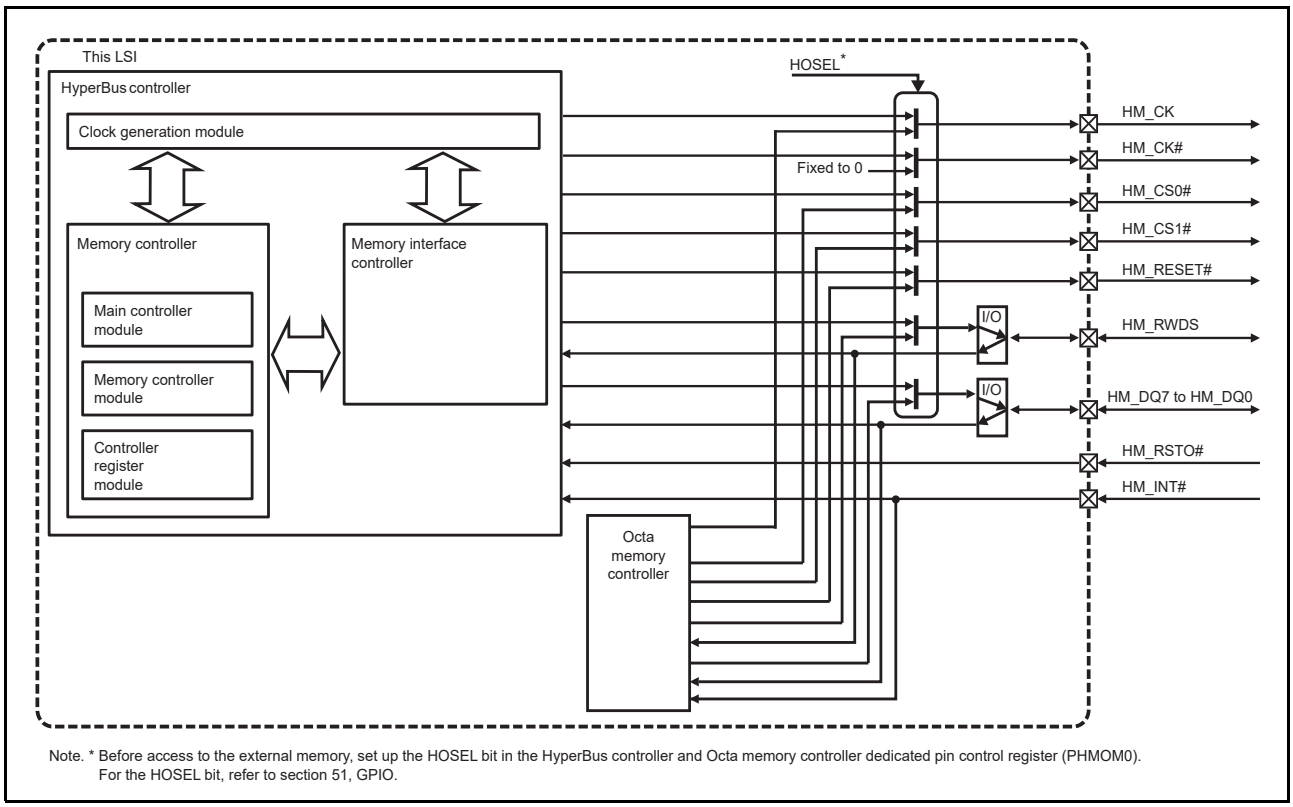
21.1 Features

- The controller supports a HyperBus interface.
- One each of a HyperFlash device and a HyperRAM device compliant with the HyperBus interface specifications are connectable.
- A chip select signal is assigned to each memory device (HM_CS0#: HyperFlash; HM_CS1#: HyperRAM).
  Note: Only one of the memory devices can operate (be read or written to) at a time.

- HM_CK and HM_CK#: 132 MHz
- PVcc_HO: 1.8 V

21.2 Block Diagram

Figure 21.1 shows a block diagram of this module.



Note. * Before access to the external memory, set up the HOSEL bit in the HyperBus controller and Octa memory controller dedicated pin control register (PHMOM0). For the HOSEL bit, refer to section 51, GPIO.

Figure 21.1 Block Diagram

21.3 System Configuration of the HyperBus

Figure 21.2 shows an example of connections between the HyperBus in this LSI and HyperFlash and HyperRAM devices.

CS0# and CS1# should respectively be connected to the HyperFlash and HyperRAM devices.

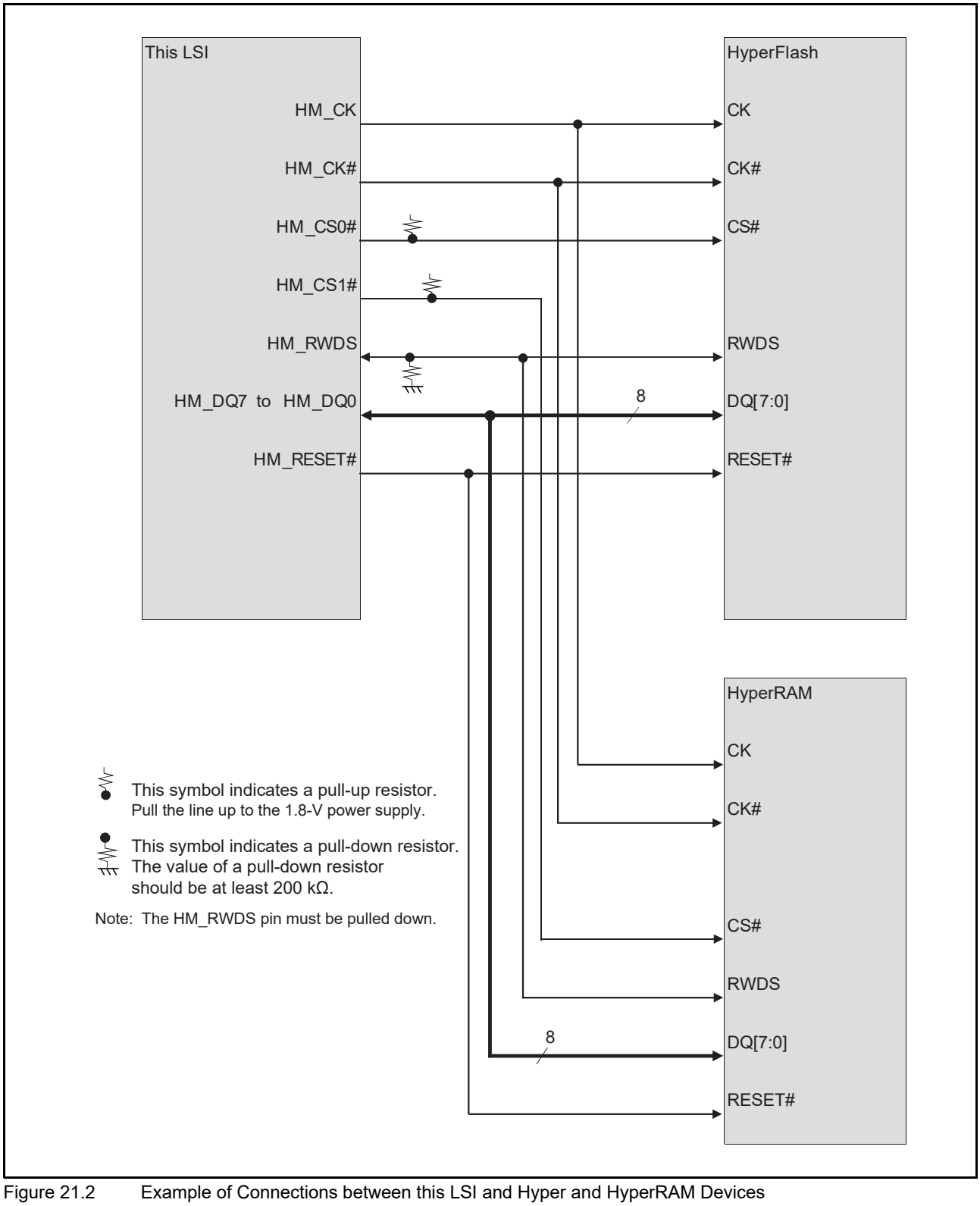


Figure 21.2 Example of Connections between this LSI and Hyper and HyperRAM Devices

21.4 Input/Output Pins

Table 21.1 lists the pins of this module.

Table 21.1 List of Input/Output Pins

Terminal	I/O	Name	Function
HM_CK	Output	Clock	Differential clock output
HM_CK#	Output	Clock	Differential clock output
HM_CS0#	Output	Chip Select 0	Chip-select signal for a HyperFlash device
HM_CS1#	Output	Chip Select 1	Chip-select signal for a HyperRAM device
HM_RWDS	Input/output	Read Write Data strobe	Read data strobe/write data mask
HM_DQ7 to HM_DQ0	Input/output	Data	Data in/out
HM_RESET#	output	Reset Output	Reset signal for both HyperFlash and HyperRAM devices
HM_RSTO#	Input	Reset Input	Reset signal from a HyperFlash device
HM_INT#	Input	Interrupt Input	Interrupt signal from a HyperFlash device

Note: HM_RSTO# and HM_INT# are input pins that are multiplexed with alternative pin functions that are driven by the 3.3-V power supply. The other pins are solely used for functions that are driven by the 1.8-V power supply.

21.5 Description of Registers

Table 21.2 lists the registers of this module.
Access to these registers is only possible in 32-bit units.
Do not attempt to access addresses other than those listed below.

Table 21.2 List of the HyperBus Controller Registers

Name	Symbol	R/W	Initial Value	Address	Access Size
Controller status register	CSR	R	0x00000000	0x1F400000	32
Interrupt enable register	IEN	R/W	0x00000000	0x1F400004	32
Interrupt status register	ISR	R	0x00000000	0x1F400008	32
CS0 memory configuration register	MCR0	R/W	0x00000003	0x1F400020	32
CS1 memory configuration register	MCR1	R/W	0x00000003	0x1F400024	32
CS0 memory timing register	MTR0	R/W	0x00000001	0x1F400030	32
CS1 memory timing register	MTR1	R/W	0x00000001	0x1F400034	32

21.5.1 Controller Status Register (CSR)

CSR indicates the internal state of the HyperBus controller.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	WRSTO ERR	WTRS ERR	WDEC ERR	—	—	—	—	—	—	—	WACT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RDS STALL	RRSTO ERR	RTRS ERR	RDEC ERR	—	—	—	—	—	—	—	RACT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0.
26	WRSTOERR	0	R	Write RSTO Error This bit indicates if the memory was in the reset state during attempted writing. The bit is set to 1 if the HyperBus memory was in the reset state during attempted writing. 0: No error 1: The device was in the reset state and the AXI slave error signal was output.
25	WTRSERR	0	R	Write Transaction Error This bit indicates whether the attempted writing conformed to AXI protocol. This bit is set to 1 if the attempted writing did not conform to AXI protocol. 0: No error 1: The attempted writing did not conform to the AXI protocol and the AXI slave error signal was output.
24	WDECERR	0	R	Write Decode Error This bit indicates whether the destination address for writing was acceptable or not. This bit is set to 1 if there is a problem with a destination address. 0: No error 1: The destination address was not acceptable and the AXI slave error signal was output.
23 to 17	—	All 0	R	Reserved These bits are always read as 0.
16	WACT	0	R	Write State This bit indicates the state of the write transaction. This bit is set to 1 when a write command is received and set to 0 at the end of the write transaction. 0: Write transaction is idle. 1: Write transaction is in progress.
15 to 12	—	All 0	R	Reserved These bits are always read as 0.
11	RDSSTALL	0	R	RDS Timeout This bit indicates whether the read transaction timed out. This bit is set to 1 when the RWDS signal of the HyperBus memory remains low for the specified time in the read transaction. 0: No error 1: The RWDS signal of the HyperBus memory remains low for the specified time and the AXI slave error signal was output.

Bit	Bit Name	Initial Value	R/W	Description
10	RRSTOERR	0	R	Read RSTO Error This bit indicates if the memory was in the reset state during attempted reading. The bit is set to 1 if the HyperBus memory was in the reset state during attempted reading. 0: No error 1: The device was in the reset state and the AXI slave error signal was output.
9	RTRSERR	0	R	Read Transaction Error This bit indicates whether the attempted reading conformed to the AXI protocol. This bit is set to 1 when the attempted reading did not conform to the AXI protocol. 0: No error 1: The attempted reading did not conform to the AXI protocol and the AXI slave error signal was output.
8	RDECERR	0	R	Read Decode Error This bit indicates whether the destination address for reading was acceptable or not. This bit is set to 1 if there is a problem with a destination address. 0: No error 1: The destination address was not acceptable and the AXI slave error signal was output.
7 to 1	—	All 0	R	Reserved These bits are always read as 0.
0	RACT	0	R	Read State This bit indicates the state of the read transaction. This bit is set to 1 when a read command is received and set to 0 at the end of the read transaction. 0: Read transaction idle state 1: Read transaction is in progress.

21.5.2 Interrupt enable register (IEN)

IEN sets the interrupt signal.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INTP	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPC INTE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	INTP	0	R/W	Interrupt polarity control 0: The IENOn signal is active low. 1: The IENOn signal is active high.
30 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RPCINTE	0	R/W	Interrupt enable 0: Disable interrupt 1: Enable interrupt by HM_INT# signal of HyperBus memory

21.5.3 Interrupt status register (ISR)

ISR indicates the state of the interrupt signal.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPC INTS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0.
0	RPCINTS	0	R	HyperBus memory interrupt state This bit indicates the state of the interrupt signal input to this controller from the HyperBus memory. 0: No interrupt 1: Interrupt

21.5.4 CS0 Memory configuration register (MCR0)

MCR0 specifies the operations of the HyperBus controller during access to the HyperFlash device.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MAXEN	—	—	—	—	MAXLEN[8:0]									—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	MAXEN	0	R/W	Maximum time setting enable This bit sets whether the time at low level of the HM_CS0# signal is to be controlled by MAXLEN[8:0]. 0: Time at low level of HM_CS0# is not configurable. 1: Time at low level of HM_CS0# is configured by MAXLEN[8:0].
30 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 18	MAXLEN[8:0]	All 0	R/W	Maximum time setting Sets the maximum read/write transaction time (the time at low level of the HM_CS0# signal). 000000000: 1 clock cycle (sufficient for 2 bytes) 000000001: 2 clock cycles (sufficient for 4 bytes) 000000010: 3 clock cycles (sufficient for 6 bytes) ... 111111111: 512 clock cycles (sufficient for 1024 bytes) When MAXEN is 0, this setting has no effect. Clock cycles above refers to cycles of the HM_CK and HM_CK# signals.
17 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	—	11	R	Reserved These bits are always read as 1. The write value should always be 1

21.5.5 CS1 Memory configuration register (MCR1)

MCR1 specifies the operations of the HyperBus controller during access to the HyperRAM device.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MAXEN	—	—	—	—	MAXLEN[8:0]									—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CRT	DEV TYPE	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	MAXEN	0	R/W	Maximum time setting enable This bit sets whether the time at low level of the HM_CS1# signal is to be controlled by MAXLEN[8:0]. 0: Time at low level of HM_CS1# is not configurable. 1: Time at low level of HM_CS1# is configured by MAXLEN[8:0].
30 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 18	MAXLEN[8:0]	All 0	R/W	Maximum time setting Sets the maximum read/write transaction time (the time at low level of the HM_CS1# signal). 000000000: 1 clock cycle (sufficient for 2 bytes) 000000001: 2 clock cycles (sufficient for 4 bytes) 000000010: 3 clock cycles (sufficient for 6 bytes) ... 111111111: 512 clock cycles (sufficient for 1024 bytes) When MAXEN is 0, this setting has no effect. Clock cycles above refers to cycles of the HM_CK and HM_CK# signals.
17 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	CRT	0	R/W	Access target This bit sets the access target of read/write operation. 0: Memory space 1: Register space
4	DEVTYPE	0	R/W	Device Type Set the value to 1.
3, 2	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	—	11	R	Reserved These bits are always read as 1. The write value should always be 1.

21.5.6 CS0 Memory Timing Register (MTR0)

MTR0 sets access timings of the HM_CS0# signal.

For each timing parameter specified in this register, refer to section 21.6.4, MTR0 and MTR1 Timing Parameters.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RCSHI[3:0]				WCSHI[3:0]				RCSS[3:0]				WCSS[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCSH[3:0]				WCSH[3:0]				—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	RCSHI[3:0]	0000	R/W	Setting Read Chip Select timing Minimum wait time between negating the HM_CS0# pin and starting the next read access 0000: 1.5 clock cycles … 1111: 16.5 clock cycles
27 to 24	WCSHI[3:0]	0000	R/W	Setting Write Chip Select timing Minimum wait time between negating the HM_CS0# pin and starting the next write access 0000: 1.5 clock cycles … 1111: 16.5 clock cycles
23 to 20	RCSS[3:0]	0000	R/W	Setting Read Chip Select Setup timing Sets setup time of the HM_CS0# signal for reading, after the HM_CS0# signal is asserted. 0000: 1 clock cycle … 1111: 16 clock cycles
19 to 16	WCSS[3:0]	0000	R/W	Setting Write Chip Select Setup timing Sets setup time of the HM_CS0# signal for writing, after the HM_CS0# signal is asserted. 0000: 1 clock cycle … 1111: 16 clock cycles
15 to 12	RCSH[3:0]	0000	R/W	Setting Read Chip Select Hold timing Set time from the end of the reading processing until the HM_CS0# signal is negated. 0000: 1 clock cycle … 1111: 16 clock cycles
11 to 8	WCSH[3:0]	0000	R/W	Setting Write Chip Select Hold timing Set time from the end of the writing processing until the HM_CS0# signal is negated. 0000: 1 clock cycle … 1111: 16 clock cycles
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

21.5.7 CS1 Memory Timing Register (MTR1)

MTR1 sets access timings of the HM_CS1# signal.
For each timing parameter specified in this register, refer to section 21.6.4, MTR0 and MTR1 Timing Parameters.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RCSHI[3:0]				WCSHI[3:0]				RCSS[3:0]				WCSS[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCSH[3:0]				WCSH[3:0]				—	—	—	—	LTCY[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	RCSHI[3:0]	0000	R/W	Setting Read Chip Select timing Minimum wait time between negating the HM_CS1# pin and starting the next read access 0000: 1.5 clock cycles … 1111: 16.5 clock cycles
27 to 24	WCSHI[3:0]	0000	R/W	Setting Write Chip Select timing Minimum wait time between negating the HM_CS1# pin and starting the next write access 0000: 1.5 clock cycles … 1111: 16.5 clock cycles
23 to 20	RCSS[3:0]	0000	R/W	Setting Read Chip Select Setup timing Sets setup time of the HM_CS1# signal for reading, after the HM_CS1# signal is asserted. 0000: 1 clock cycles … 1111: 16 clock cycles
19 to 16	WCSS[3:0]	0000	R/W	Setting Write Chip Select Setup timing Sets setup time of the HM_CS1# signal for writing, after the HM_CS1# signal is asserted. 0000: 1 clock cycles … 1111: 16 clock cycles
15 to 12	RCSH[3:0]	0000	R/W	Setting Read Chip Select Hold timing Set time from the end of the reading processing until the HM_CS1# signal is negated. 0000: 1 clock cycles … 1111: 16 clock cycles
11 to 8	WCSH[3:0]	0000	R/W	Setting Write Chip Select Hold timing Set time from the end of the writing processing until the HM_CS1# signal is negated. 0000: 1 clock cycles … 1111: 16 clock cycles
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	LTCY[3:0]	0001	R/W	Setting Latency Cycle These bits specify the read and write latency. These bits should only be used in a HyperRAM device. These bits have no effect when the setting of MCR1.DEVTYPE is 0 (HyperFlash is specified). 0000: 5 clock cycles of latency 0001: 6 clock cycles of latency … 0010 to 1111: Setting prohibited

21.6 Operation

21.6.1 Address Map

For read access in the memory-map mode, HyperFlash is allocated to the HyperFlash space (0x30000000 to 0x3FFFFFFF), and HyperRAM is allocated to the HyperRAM space (0x40000000 to 0x4FFFFFFF).
One each of a HyperFlash device and a HyperRAM device can be connected to this LSI chip, with up to 256 Mbytes being accessible in each of the devices.

	Internal Address	Maximum Accessible Area
HyperFlash	0x30000000 to 0x3FFFFFFF	Up to 256 Mbytes
HyperRAM	0x40000000 to 0x4FFFFFFF	Up to 256 Mbytes

21.6.2 HyperBus Memory Interface

This section describes the HyperBus memory interface.

21.6.2.1 Write Operation

This section describes the write operation of HyperBus controller.
HyperBus controller asserts HM_CS0# (HM_CS1#), HM_CK, HM_CK#, HM_DQ7 to HM_DQ0. Then, it outputs 3 words command/address, and writes 1 or more data. The cycle of writing to the HyperRAM device includes the initial clock latency before writing of the data.
Figure 21.3 shows the write waveform of HyperFlash interface Figure 21.4 shows the write waveform of HyperRAM interface.

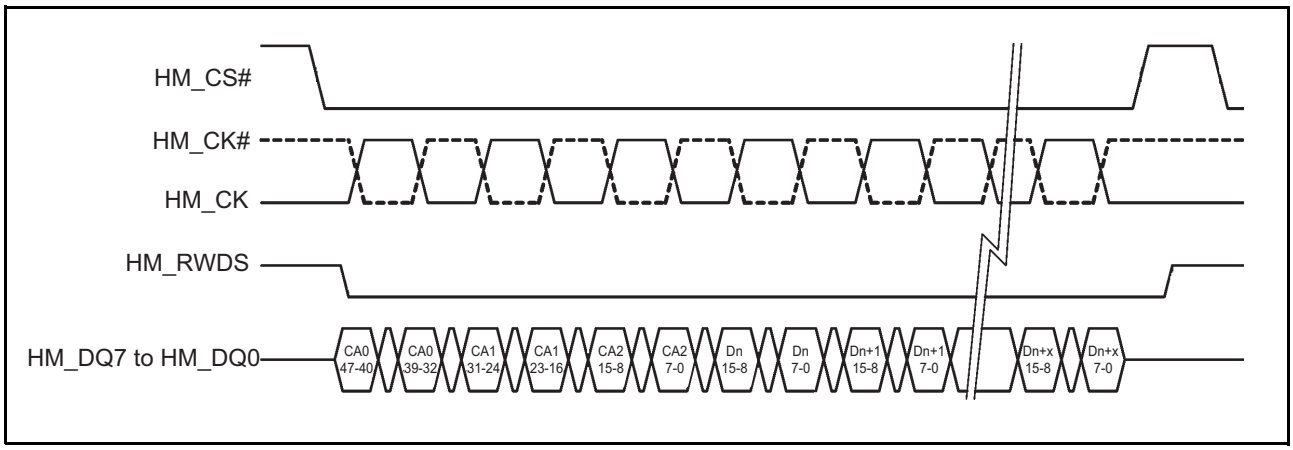


Figure 21.3 Write Waveform of HyperFlash Interface

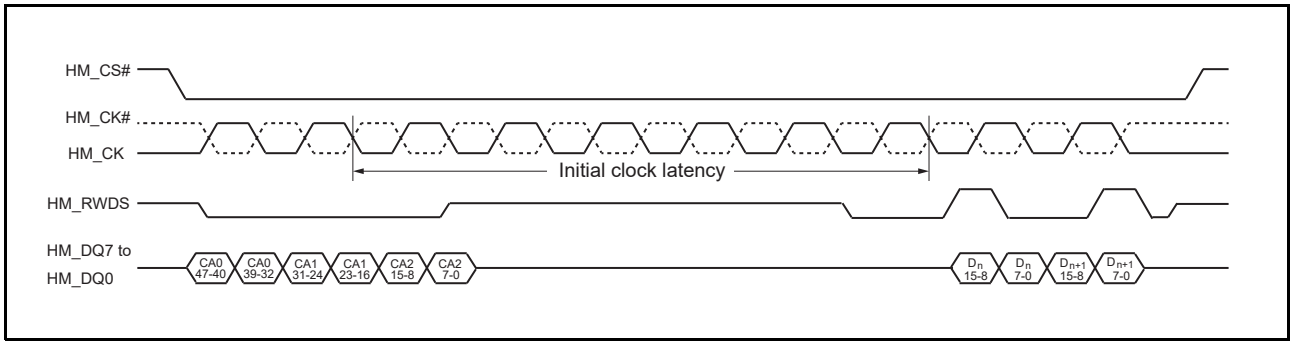


Figure 21.4 Write waveform of HyperRAM interface

21.6.2.2 Read Operation

This section describes the read operation of HyperBus memory interface controller.
HyperBus controller asserts HM_CS0# (HM_CS1#), HM_CK, HM_CK#, and HM_DQ7 to HM_DQ0. Then, it outputs 3 words command/address, and reads 1 or more data by timing of HM_RWDS.
Figure 21.5 shows the read waveform of HyperBus memory interface.

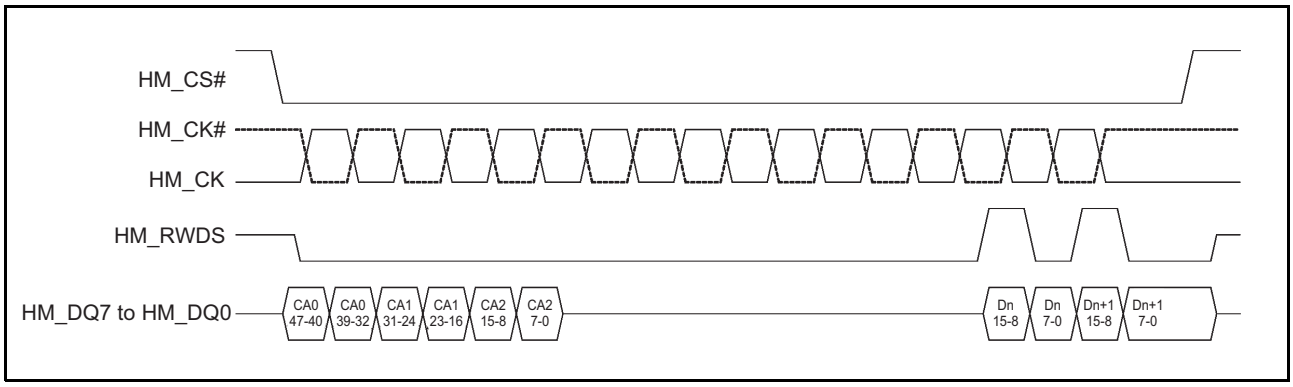


Figure 21.5 Read waveform of HyperBus memory interface

21.6.2.3 One-Byte Write Access

This section describes one-byte write access.
The HyperBus controller outputs three words of command and address data, followed after a cycle for latency by the data to be written in units of two bytes.
When the target device is HyperRAM, the output RWDS signal of the HyperBus interface can be used to mask bytes by driving it to the high level during the operation of writing. This allows the masking of invalid write data.
When the target device is HyperFlash, one-byte write access must not be attempted since HyperFlash only supports two-byte access.

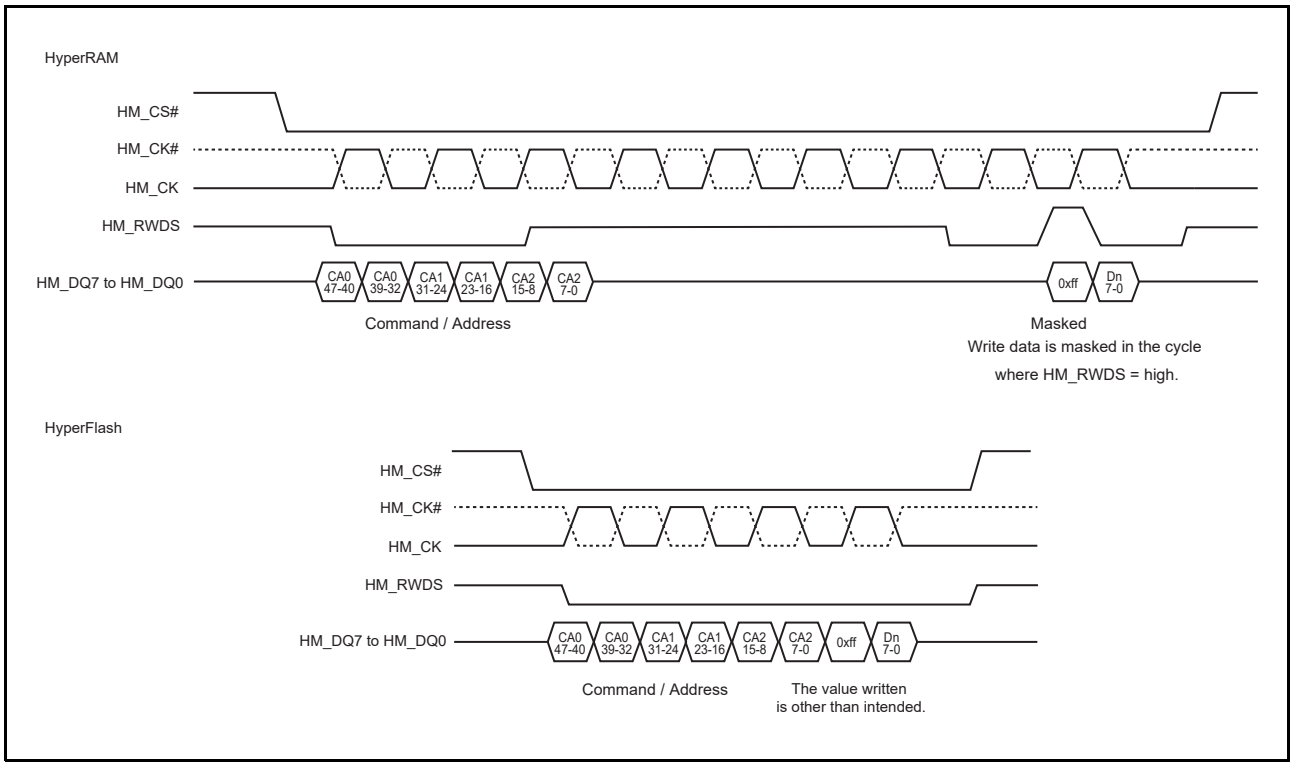


Figure 21.6 Waveforms of One-Byte Writing

21.6.2.4 Command/Address Bit Assignments

HyperBus memory device uses 6 bytes of command/address information to define the transactions characteristics. Table 21.3 shows the assignments of command/address bit in the HyperBus controller.

Table 21.3 Command/Address Bit Assignments

Bit	Bit Name	Description
47	R/W#	0: Write, 1: Read
46	Access target	0: Memory access, 1: Register access
45	Burst type	0: Wrapped burst, 1: Linear burst Note: The output of data must be in accord with the burst type of the AXI bus. The HyperBus controller only supports the transfer of 32-byte and 64-byte bursts. When the secondary cache is used, the burst-access setting of the secondary cache must match the burst-length settings for HyperFlash and HyperRAM so that the wrap boundaries of the burst addresses for the AXI bus match the address boundaries of the wrapped bursts for HyperFlash and HyperRAM.
44	Reserved	0
43 to 16	High-order address	System word address bits [30:3]
15	Reserved	0
14, 13	Reserved	11
12 to 3	Reserved	All 0
2 to 0	Low-order address	System word address bits [2:0]

Burst-Length Settings for HyperFlash and HyperRAM	Setting in the Prefetch Control Register for the Secondary Cache		
	Bit 30	Bit 27	Bit 23
32 bytes	0	0	0
64 bytes	1	0	1
Other than above: Setting prohibited	—		

21.6.2.5 Configuration Register Data Alignment in the HyperRAM Space

Figure 21.7 shows the data alignment when the configuration register in the HyperRAM space is accessed.

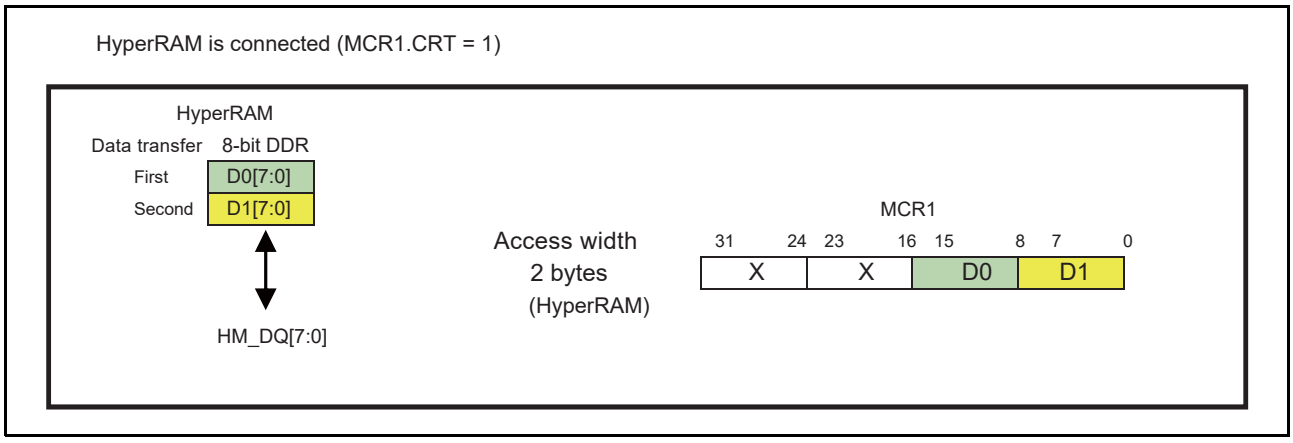


Figure 21.7 Configuration Register Data Alignment in the HyperRAM Space

21.6.3 Operation Flows

This section shows examples of flows of operation of the HyperBus controller.
Before access to the external memory, set up the HOSEL bit in the HyperBus controller and Octa memory controller dedicated pin control register (PHMOM0) and the HYMCR bit field in the SCLK select register (SCLKSEL) in the initial setting steps.
For the HOSEL bit, refer to section 51, GPIO.
For the HYMCR bit field, refer to section 6, Clock Pulse Generator.

21.6.3.1 Write Operation Flow

Figure 21.8 shows the procedure for the writing of N bytes to a HyperFlash device in 2-byte units (Word Program) as an example of the operation flow of writing.

Note: When writing four or more bytes, use a clock frequency that is compliant with the specifications of the HyperBus memory interface.

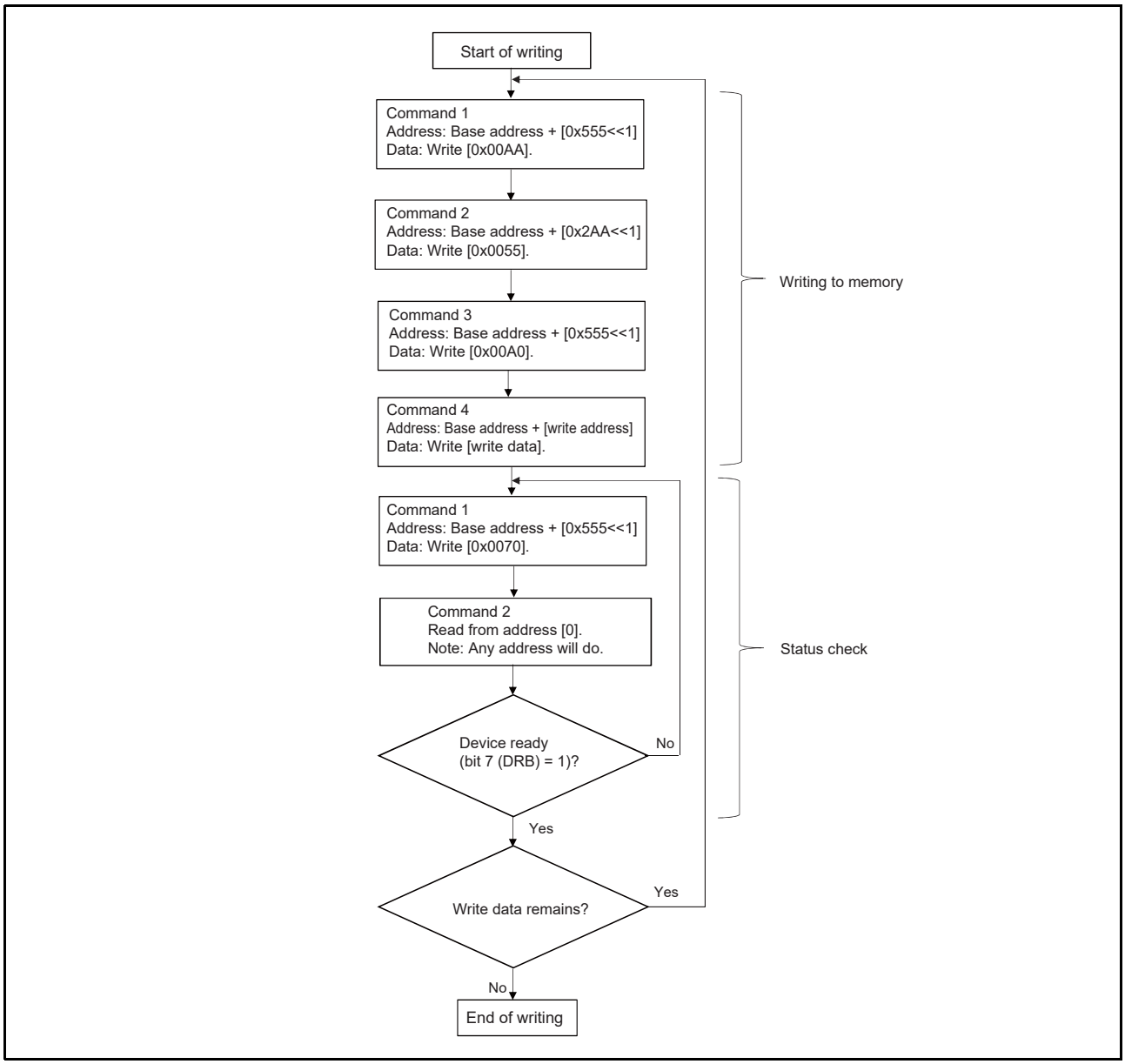


Figure 21.8 Flow of Write Operation for HyperFlash

21.6.3.2 Erase Operation Flow

Figure 21.9 shows an example of the procedure for sector erase.

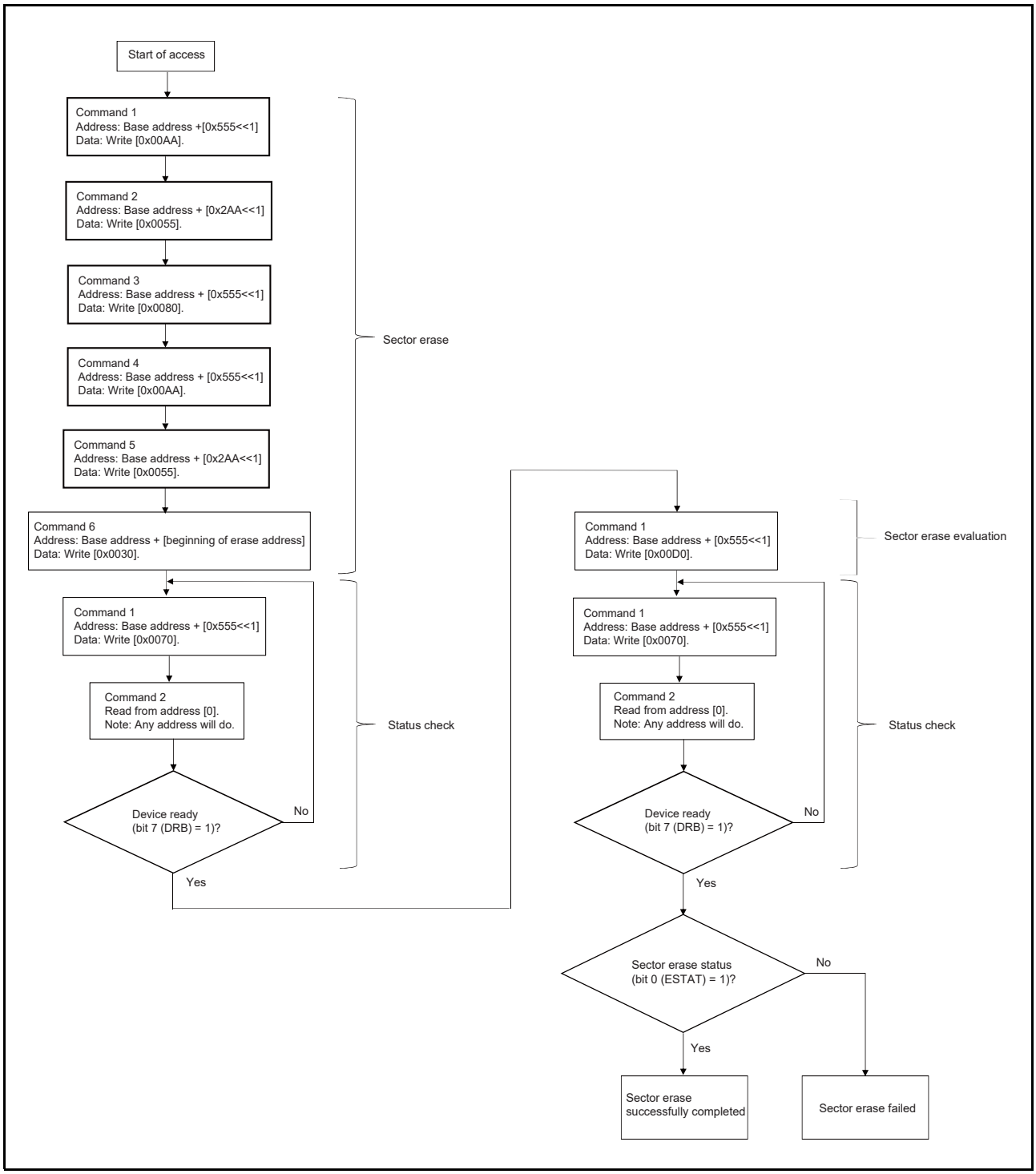


Figure 21.9 Flow of Sector Erase Operation for HyperFlash

21.6.3.3 Flow of Write and Read Operations for the Configuration Register

Figure 21.10 shows an example of writing and reading operations for the volatile configuration registers in a HyperFlash device.

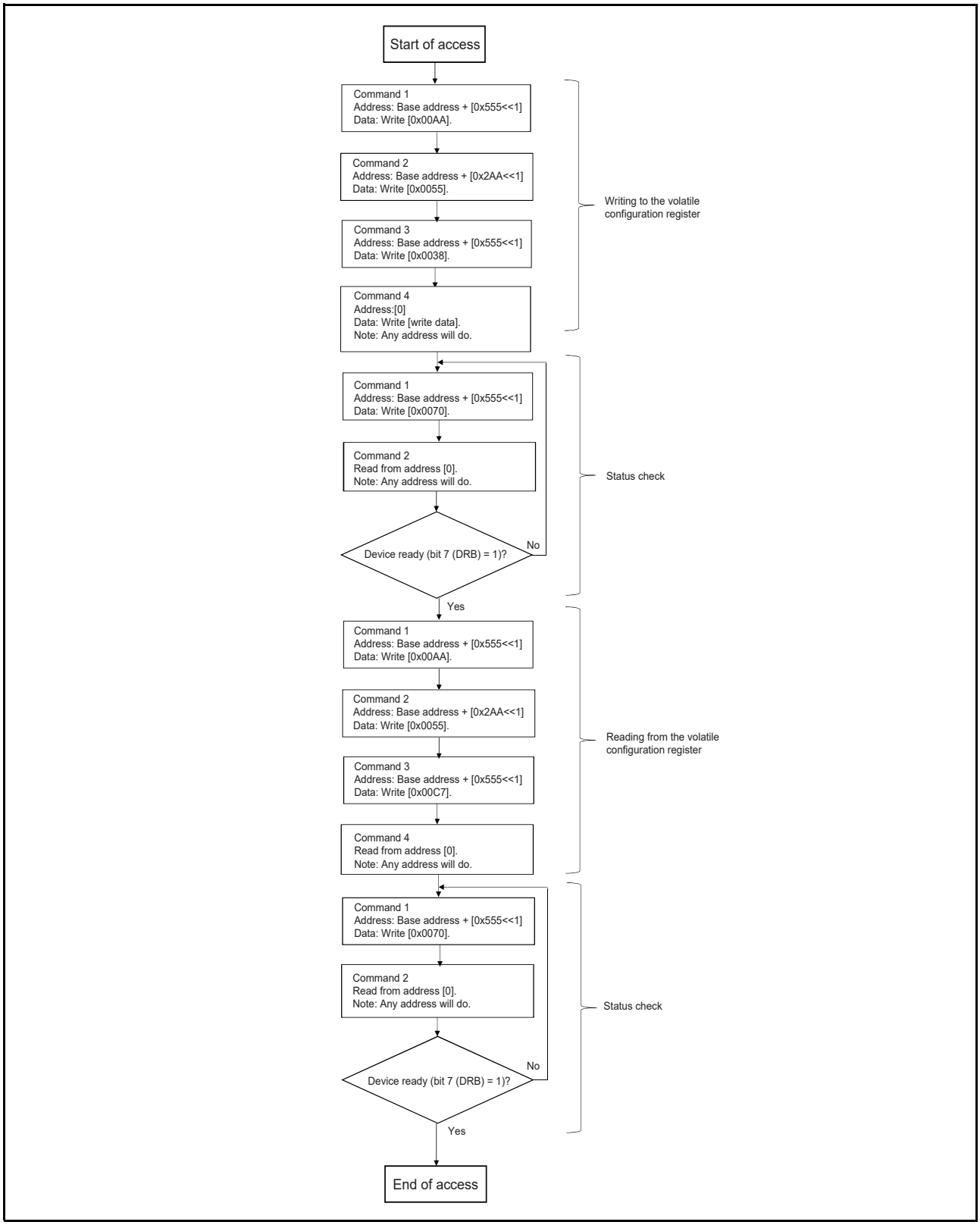


Figure 21.10 Flow of Write and Read Operations for the Volatile Configuration Register

21.6.4 MTR0 and MTR1 Timing Parameters

The timing parameters specified in the CS0 memory timing register (MTR0) and CS1 memory timing register (MTR1) described in section 21.5.6 and section 21.5.7 are shown in the following timing chart.

The chart shows the timings of read and write access by the HyperBus controller to memory (HyperFlash or HyperRAM).

(1) WCSS/RCSS is the period between assertion of the HM_CS0#/HM_CS1# signal and the first edge of the HM_CK/HM_CK# signal.
(2) WCSH/RCSH is the period between the last edge of the HM_CK/HM_CK# signal for the current data transfer and negation of the HM_CS0#/HM_CS1# signal.
(3) WCSHI/RCSHI is the period between the end of a current transfer and the beginning of the next.
(4) LTCY should only be specified in MTR1. Specifying this timing is only required for a HyperRAM device; it is not required for a HyperFlash device.

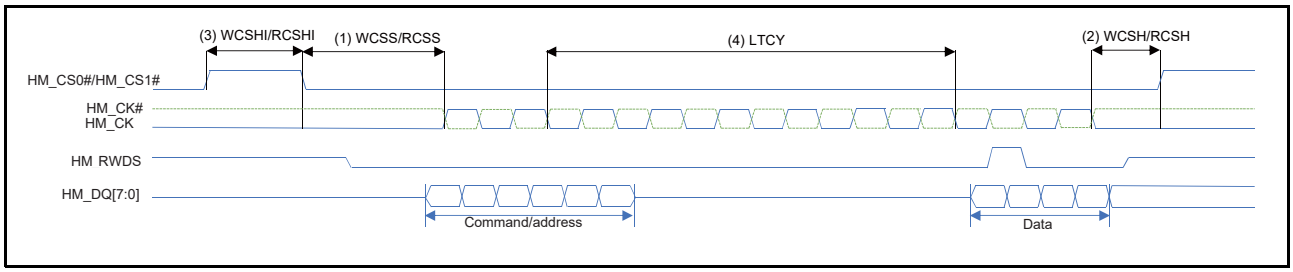


Figure 21.11 Timing Chart of Read and Write Operations

22. Octa Memory Controller

The Octa memory controller enables the connection of OctaFlash™ and OctaRAM™ to this LSI chip.

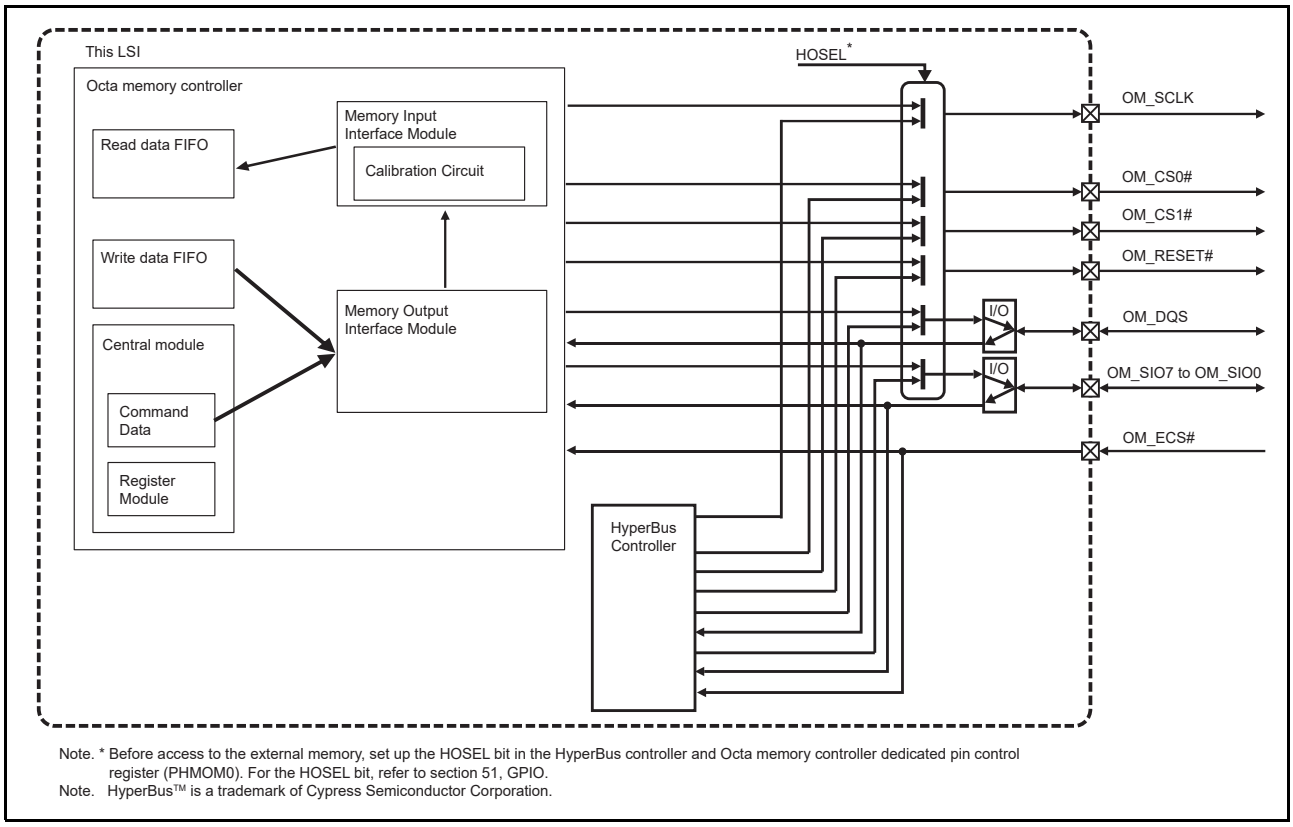
Note: OctaFlash™ and OctaRAM™ are trademarks of Macronix International Co., Ltd.

22.1 Features

- Macronix Serial Multi I/O (MXSMIO®) Octa Peripheral Interface (OPI) for high-end consumer applications is supported.
- One each of an OctaFlash device and an OctaRAM device compliant with the OPI specifications are connectable.
- A chip select signal is assigned to each memory device (OM_CS0#: OctaFlash; OM_CS1#: OctaRAM).
  Note: Only one of the memory devices can operate (be read or written to) at a time.
- Supported device interfaces
  SPI: Serial peripheral interface (OctaFlash, SPI mode)
  SOPI: Single Octa I/O (8 bits) (OctaFlash, single data rate)
  DOPI: Double Octa I/O (8 bits) (OctaFlash and OctaRAM, double data rate)
- 3-byte and 4-byte OctaFlash address commands are supported.
- 4-byte OctaRAM address commands are supported.
- The read-while-write (RWW) operation is supported.
- 1LC and 2LC (latency count) OctaRAM devices are supported.
- The error corrected signal (ECS#) is available, and ECC errors can be detected (for OctaFlash only).
- Fastboot mode is not supported.
- Direct access (memory-mapped reading and writing) by the CPU to memory devices in separate flash and RAM address spaces is supported.
- OM_CK: 132 MHz
- PVcc_HO: 1.8 V

22.2 Block Diagram

Figure 22.1 shows a block diagram of this module.



Note. * Before access to the external memory, set up the HOSEL bit in the HyperBus controller and Octa memory controller dedicated pin control register (PHMOM0). For the HOSEL bit, refer to section 51, GPIO.
Note. HyperBus™ is a trademark of Cypress Semiconductor Corporation.

Figure 22.1 Block Diagram

22.3 Input/Output Pins

Table 22.1 shows the pin configuration.

Table 22.1 Configuration of Hyper Memory Controller Pins

Pin	I/O	Name	Function
OM_SCLK	Output	Clock	Clock output
OM_CS0#	Output	Chip select 0	Chip select signal for an OctaFlash device
OM_CS1#	Output	Chip select 1	Chip select signal for an OctaRAM device
OM_DQS	Input/Output	Read/write data strobe	Read data strobe/write data mask signal
OM_SIO7 to OM_SIO0	Input/Output	Data	Data input/output
OM_RESET#	Output	Reset output	Reset signal for both OctaFlash and OctaRAM devices
OM_ECS#	Input	ECC error detection	ECC error detection signal from the external memory

Note: OM_ECS# is an input pin that is multiplexed with alternative pin functions that are driven by the 3.3-V power supply. The other pins are solely used for functions that are driven by the 1.8-V power supply.

22.3.1 Device Interface

The device interface is compatible with OctaFlash/OctaRAM interface.
When reading data from the memory or programming the memory with data, the bit order changes with the I/O mode. The bit order is shown below:

Table 22.2 Bit order in accordance with I/O mode

I/O mode	Bit order (MSB)
Read data from device	
1 I/O (SPI)	OM_SIO1 input
8 I/O (SOPI)	OM_SIO7 to OM_SIO0 input
8 I/O (DOPI)	OM_SIO7 to OM_SIO0 input. Order of data: {D1, D0}, {D3, D2}, ...
Program/Write data to device	
1 I/O (SPI)	OM_SIO0 output
8 I/O (SOPI)	OM_SIO7 to OM_SIO0 output
8 I/O (DOPI)	OM_SIO7 to OM_SIO0 output. Order of data: {D1, D0}, {D3, D2}, ...

22.4 Register Descriptions

Table 22.3 shows the register configuration.
Access to the flash memory or RAM is allowed in either of the following modes.

Memory-map mode

- Reading from and writing to the memory array areas of OctaFlash memory or RAM is possible.

Configuration mode

- Transfer of any commands and data is possible.

Table 22.3 Octa memory controller register configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Device command register	DCR	R/W	H'00000000	H'1F401000	32
Device address register	DAR	R/W	H'00000000	H'1F401004	32
Device command setting register	DCSR	R/W	H'00000000	H'1F401008	32
Device size register 0	DSR0	R/W	H'00000000	H'1F40100C	32
Device size register 1	DSR1	R/W	H'00000000	H'1F401010	32
Memory delay trim register	MDTR	R/W	H'06009400	H'1F401014	32
Auto-calibration timer register	ACTR	R/W	H'10000000	H'1F401018	32
Auto-calibration address register 0	ACAR0	R/W	H'00000000	H'1F40101C	32
Auto-calibration address register 1	ACAR1	R/W	H'00000000	H'1F401020	32
Device memory map read chip select timing setting register	DRCSTR	R/W	H'00000000	H'1F401034	32
Device memory map write chip select timing setting register	DWCSTR	R/W	H'00000000	H'1F401038	32
Device chip select timing setting register	DCSTR	R/W	H'00000000	H'1F40103C	32
Controller and device setting register	CDSR	R/W	H'00000000	H'1F401040	32
Memory map dummy length register	MDLR	R/W	H'00000000	H'1F401044	32
Memory map read/write command register 0	MRWCR0	R/W	H'00000000	H'1F401048	32
Memory map read/write command register 1	MRWCR1	R/W	H'00000000	H'1F40104C	32
Memory map read/write setting register	MRWCSR	R/W	H'00000000	H'1F401050	32
Error status register	ESR	R	H'00000000	H'1F401054	32
Configure write without data register	CWNDR	W	H'00000000	H'1F401058	32
Configure write data register	CWDR	W	H'00000000	H'1F40105C	32
Configure read register	CRR	R	H'00000000	H'1F401060	32

22.4.1 Device Command Register (DCR)

DCR sets the device commands for controller operation.

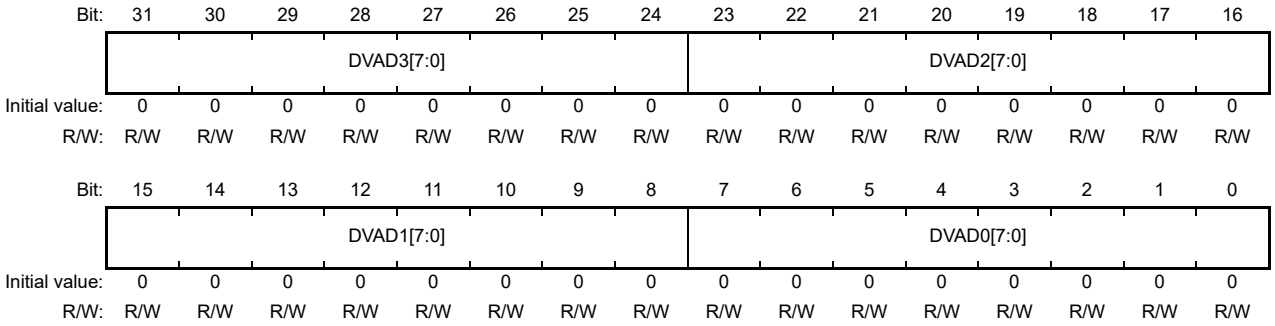
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DVCMD1[7:0]								DVCMD0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	DVCMD1[7:0]	All 0	R/W	Device Command data Sets the data to be transferred as a command to the device. If DCSR.CMDLEN[2:0] = 2, the controller will send DCR[15:8] as the 1st byte to memory.
7 to 0	DVCMD0[7:0]	All 0	R/W	Device Command data Sets the data to be transferred as a command to the device. If DCSR.CMDLEN[2:0] = 1, the controller will send DCR[7:0] as the 1st byte to memory. If DCSR.CMDLEN[2:0] = 2, the controller will send DCR[7:0] as the 2nd byte to memory.

22.4.2 Device Address Register (DAR)

DAR sets the device address for controller operation.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DVAD3[7:0]	All 0	R/W	Device Address data 3 Set the device address data. When DCSR.ADLEN[2:0] = 4 only, set as the first byte.
23 to 16	DVAD2[7:0]	All 0	R/W	Device Address data 2 Set the device address data. When DCSR.ADLEN[2:0] = 3, set as the first byte. When DCSR.ADLEN[2:0] = 4, set as the second byte.
15 to 8	DVAD1[7:0]	All 0	R/W	Device Address data 1 Set the device address data. When DCSR.ADLEN[2:0] = 3, set as the second byte. When DCSR.ADLEN[2:0] = 4, set as the third byte.
7 to 0	DVAD0[7:0]	All 0	R/W	Device Address data 0 Set the device address data. When DCSR.ADLEN[2:0] = 3, set as the third byte. When DCSR.ADLEN[2:0] = 4, set as the fourth byte.

22.4.3 Device Command Setting Register (DCSR)

DCSR sets device commands.

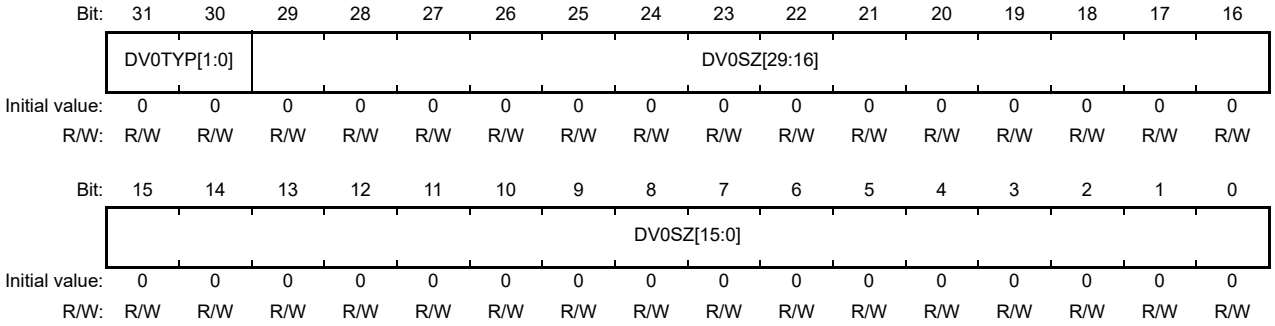
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	ACDA	DOPI	ADLEN[2:0]			DAOR	CMDLEN[2:0]			ACDV	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMLEN[7:0]								DALEN[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	ACDA	0	R/W	Data Access Control Set data access or register access 0: Register access Do not arrange the transfer data. 1: Data access (1) the address bit0 will be forced to 0 in DOPI mode. (2) If DAR[0] of the lead type command is 1, byte 0 from the device will be dropped in DOPI mode. (3) The AXI read data will be arranged according to the DAR. For example: When DAR[1:0] = 1, the first valid data is stored in RDATA[15:8]. (4) The AXI write data is used according to the DAR in DOPI mode. For example: If the target device is Flash Memory and DAR[1:0] = 1, the first byte sent to the flash memory is forced to 0xff. If the target device is RAM and DAR[1: 0] = 1, the first byte sent to RAM is masked by DQSM. (5) When the target device is RAM, the address sent to the memory changes according to RAM specification.
27	DOPI	0	R/W	DOPI single byte setting Even in DOPI mode, set this bit to 1 when the read data has only one byte in each cycle. For example: OctaFlash RDID, RDSR command 0: Each cycle has two bytes data. (normal DOPI mode) 1: Each cycle has one byte data. (The byte data changes at the rising edge of the clock and does not change at the falling edge of the clock.)
26 to 24	ADLEN[2:0]	000	R/W	Transfer address length setting Sets the length of the address to be transferred in bytes.
23	DAOR	0	R/W	Data order setting Sets the byte order of data during read and write operations. 0: byte0, byte1, byte2, byte3 1: byte1, byte0, byte3, byte2
22 to 20	CMDLEN[2:0]	000	R/W	Transfer command length setting Sets the length of the command to be transferred in bytes.
19	ACDV	0	R/W	Access Device setting Sets the device to be accessed. 0: Send commands to device 0. 1: Send commands to device 1.
18 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	DMLEN[7:0]	All 0	R/W	Dummy cycle (latency) setting Sets the length of dummy cycle (latency) in OM_SCLK units.
7 to 0	DALEN[7:0]	All 0	R/W	Transfer data length setting Sets the length of data to be transferred in bytes. When [7:0] is 0, there are no transmission data to memory and reception data from memory.

22.4.4 Device Size Register 0 (DSR0)

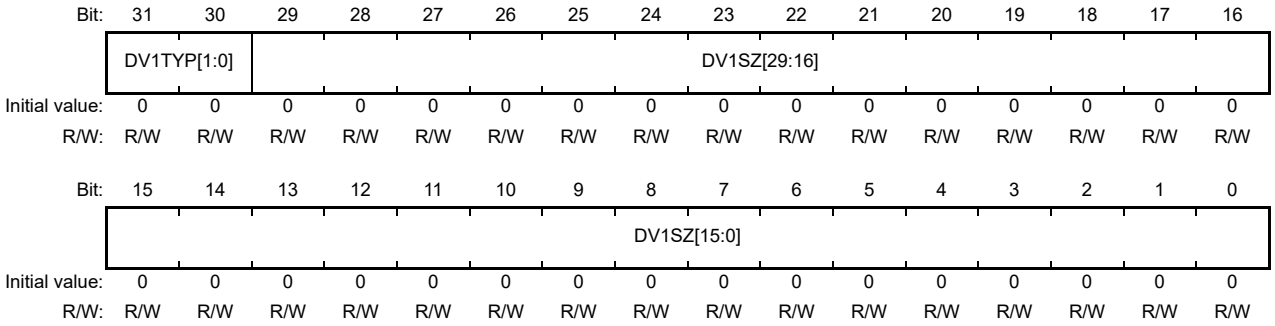
DSR0 specifies the type of memory to be accessed as device 0 and the size of the flash memory.



Bit	Bit Name	Initial Value	R/W	Description
31, 30	DV0TYP[1:0]	00	R/W	Device 0 type setting Set to 00.
29 to 0	DV0SZ[29:0]	All 0	R/W	Device 0 size setting Set a 30-bit value for the size of memory connected as device 0. Examples: H'10000000: 256-Mbyte flash memory H'08000000: 128-Mbyte flash memory

22.4.5 Device Size Register 1 (DSR1)

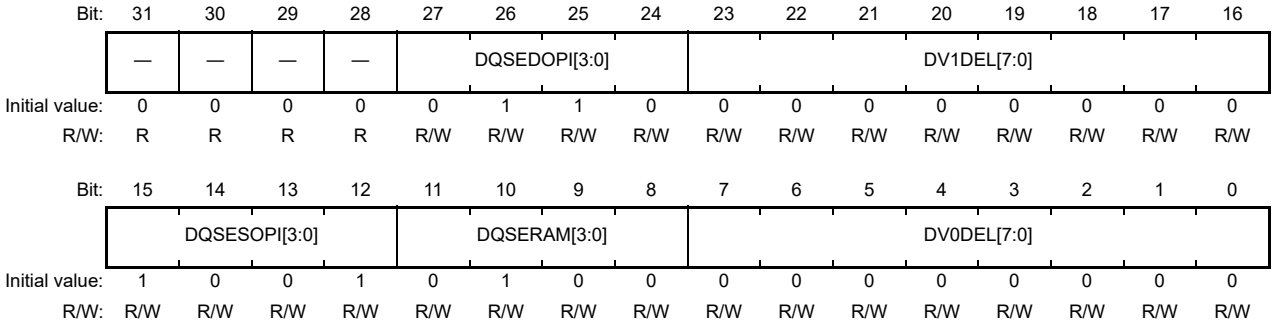
DSR1 specifies the type of memory to be accessed as device 1 and the size of the RAM.



Bit	Bit Name	Initial Value	R/W	Description
31, 30	DV1TYP[1:0]	00	R/W	Device 1 type setting Set to 01.
29 to 0	DV1SZ[29:0]	All 0	R/W	Device 1 size setting Set a 30-bit value for the size of memory connected as device 1. Examples: H'01000000: 16-Mbyte RAM H'00800000: 8-Mbyte RAM

22.4.6 Memory Delay Trim Register (MDTR)

MDTR sets the timing adjustment of memory access.
For the information about the auto-calibration, refer to section 22.5, Operation.



Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	DQSEDOPI[3:0]	H'6	R/W	OM_DQS enable counter Setting for the flash memory in the DOPI mode
23 to 16	DV1DEL[7:0]	All 0	R/W	Device 1 delay setting These bits specify the number of cycles of delay to be inserted in the input strobe signal (OM_DQS) to adjust the timing for latching data during read access to device 1 in the DOPI mode (the RAM).
15 to 12	DQSESOPI[3:0]	H'9	R/W	OM_DQS enable counter Setting for the flash memory in the DOPI mode
11 to 8	DQSERAM[3:0]	H'4	R/W	OM_DQS enable counter Setting for the RAM in the DOPI mode
7 to 0	DV0DEL[7:0]	All 0	R/W	Device 0 delay setting These bits specify the delay cycles to be inserted on the input strobe signal (OM_DQS) to adjust the timing for latching data during read access to device 0 in the SOPI or DOPI mode for the flash memory.

During a read operation with the OM_DQS clock input (in the SOPI or DOPI mode), the OM_DQS clock transitions from the high-impedance state to the input state (for receiving a value of 0 from the external device) after the command and address phases are completed. To obtain valid data, adjust the DQS enable counter and delay cycles.

OM_DQS enable counter setting (OM_SCLK units)
0000: 1 clock cycle
0001: 2 clock cycles
0010: 3 clock cycles
0010: 4 clock cycles
...
1111: 16 clock cycles

OM_DQS enable counter setting examples

Table 22.4 Examples of Setting the OM_DQS Enable Counter for OctaRAM

	RAM	
Number of dummy cycles (latency)	Pre-cycle On	Pre-cycle Off
3	4	4 to 5
4	4 to 5	4 to 6
5	4 to 6	4 to 7
6	4 to 7	4 to 8
7	4 to 8	4 to 9
8	4 to 9	4 to 10

Table 22.5 Examples of Setting the OM_DQS Enable Counter for OctaFlash Memory

	SOPI	DOPI	
Number of dummy cycles (latency)	Pre-cycle On	Pre-cycle On	Pre-cycle Off
4	8 to 9	5 to 6	5 to 7
6	8 to 11	5 to 8	5 to 9
8	8 to 13	5 to 10	5 to 11
10	8 to 15	5 to 12	5 to 13
12	8 to 15	5 to 14	5 to 15
14	8 to 15	5 to 15	5 to 15
16	8 to 15	5 to 15	5 to 15
18	8 to 15	5 to 15	5 to 15
20	8 to 15	5 to 15	5 to 15

These two tables list cases of DQS delay.
If DQS delay cycle is not within this range, refer to section 22.7, OM_DQS Enable Counter, and recalculate the correct range.

22.4.7 Auto-Calibration Timer Register (ACTR)

ACTR sets the cycle at which automatic calibration is executed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CTP[31:16]															
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CTP[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CTP[31:0]	H'10000000	R/W	Automatic calibration cycle time setting Sets performing cycle for the automatic calibration with 32 bits. Automatic calibration cycle time (ns) = CTP[31:0] × Bφ When automatic calibration is enabled (CDSR.ACMODE[1:0] =H'1) and the value of the internal timer is equal to that of this register, automatic calibration will start.

22.4.8 Auto-Calibration Address Register 0 (ACAR0)

ACAR0 sets the address in OctaFlash memory for writing to and reading from in auto-calibration of device 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAD0[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAD0[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CAD0[31:0]	All 0	R/W	Automatic calibration address These bits set the address in OctaFlash memory for writing to and reading from in auto-calibration of device 0. Note: Set an address in terms of the memory space of the OctaFlash device, not in the OctaFlash space within this LSI chip.

22.4.9 Auto-Calibration Address Register 1 (ACAR1)

ACAR1 sets the address in OctaRAM for writing to and reading from in auto-calibration of device 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAD1[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAD1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CAD1[31:0]	All 0	R/W	Automatic calibration address These bits set the address in OctaRAM for writing to and reading from in auto-calibration of device 1. Note: Set an address in terms of the memory space of the OctaRAM device, not in the OctaRAM space within this LSI chip.

22.4.10 Device Memory Map Read Chip Select Timing Setting Register (DRCSTR)

DRCSTR sets the timing of memory-mapped reading for each device.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DVRDLO1[1:0]		DVRDHI1[2:0]			DVRDCMD1[2:0]			—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DVRDLO0[1:0]		DVRDHI0[2:0]			DVRDCMD0[2:0]			CTR0	CTRW0[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	DVRDLO1[1:0]	00	R/W	Device 1 select signal pull-down timing setting* Indicates the timing between the select signal pull-down to the read operation for the device 1. Timing definition from OM_CS1# low to the first OM_SCLK high DOPI mode 00: 1.5 clock cycles 01: 2.5 clock cycles 10: 3.5 clock cycles 11: 4.5 clock cycles Other mode 00: 2 clock cycles 01: 3 clock cycles 10: 4 clock cycles 11: 5 clock cycles
29 to 27	DVRDHI1[2:0]	000	R/W	Device 1 select signal High timing setting* Indicates the timing, from the end of read operation to the select signal is pulled up. Timing definition from the last OM_SCLK low to OM_CS1# high DOPI or (SOPI) mode 000: Setting prohibited 001: Setting prohibited 010: Setting prohibited 011: Setting prohibited 100: Setting prohibited 101: 6.5 (7) clock cycles 110: 7.5 (8) clock cycles 111: 8.5 (9) clock cycles Note: The values in parentheses apply to SOPI mode. Other mode 000: 2 clock cycles 001: 3 clock cycles 010: 4 clock cycles 011: 5 clock cycles 100: 6 clock cycles 101: 7 clock cycles 110: 8 clock cycles 111: 9 clock cycles

Bit	Bit Name	Initial Value	R/W	Description
26 to 24	DVRDCMD1[2:0]	000	R/W	Device 1 Command execution interval* Indicates the timing, between command and command for device 1. Timing, until the OM_CS1# signal goes from High to the next Low 000: 2 clock cycles 001: 5 clock cycles 010: 7 clock cycles 011: 9 clock cycles 100: 11 clock cycles 101: 13 clock cycles 110: 15 clock cycles 111: 17 clock cycles
23 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15, 14	DVRDLO0[1:0]	00	R/W	Device 0 select signal pull-down timing setting* Indicates the timing between the select signal pull-down to the read operation for the device 0. Timing definition from OM_CS0# low to the first OM_SCLK high DOPI mode 00: 1.5 clock cycles 01: 2.5 clock cycles 10: 3.5 clock cycles 11: 4.5 clock cycles Other mode 00: 2 clock cycles 01: 3 clock cycles 10: 4 clock cycles 11: 5 clock cycles
13 to 11	DVRDHI0[2:0]	000	R/W	Device 0 select signal pull-up timing setting* Indicates the timing, from the end of read operation to the select signal is pulled up. Timing definition from the last OM_SCLK low to OM_CS0# high DOPI or (SOPI) mode 000: Setting prohibited 001: Setting prohibited 010: Setting prohibited 011: Setting prohibited 100: Setting prohibited 101: 6.5 (7) clock cycles 110: 7.5 (8) clock cycles 111: 8.5 (9) clock cycles Note: The values in parentheses apply to SOPI mode. Other mode 000: 2 clock cycles 001: 3 clock cycles 010: 4 clock cycles 011: 5 clock cycles 100: 6 clock cycles 101: 7 clock cycles 110: 8 clock cycles 111: 9 clock cycles
10 to 8	DVRDCMD0[2:0]	000	R/W	Device 0 Command execution interval setting* Indicates the timing between command to command for the device 0. Timing definition from OM_CS0# high to the next OM_CS0# low 000: 2 clock cycles 001: 5 clock cycles 010: 7 clock cycles 011: 9 clock cycles 100: 11 clock cycles 101: 13 clock cycles 110: 15 clock cycles 111: 17 clock cycles

Bit	Bit Name	Initial Value	R/W	Description
7	CTR0	0	R/W	Device 0 Continuous read setting When the continuous read is enabled, wait for the continuous read operation to be performed using the value set in CTRW0[6:0]. Do not set this bit to 1 except for OctaFlash DOPI mode. 0: Continuous read disabled for device 0 1: Continuous read enabled for device 0
6 to 0	CTRW0[6:0]	All 0	R/W	Device 0 continuous read cycle setting When continuous read of Device 0 is enabled, the selection signal is fixed to Low for the set clock cycle, after the read operation is performed. If the number of clock cycles exceeds twice the value set in these bits before the next read operation is performed, the select signal is returned to High.

Note: These clock cycles are values in internal clock units. In OM_SCLK units, the clock cycle is 1/2 of the above table.

22.4.11 Device Memory Map Write Chip Select Timing Setting Register (DWCSTR)

DWCSTR sets the timing of memory-mapped writing for each device.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DVWLO1[1:0]		DVWHI1[2:0]			DVWCMD1[2:0]			—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DVWLO0[1:0]		DVWHI0[2:0]			DVWCMD0[2:0]			—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	DVWLO1[1:0]	00	R/W	Device 1 select signal pull-down timing setting* Indicates the timing between the device 1 select signal pull-down to the write operation. Timing definition from OM_CS1# low to the first OM_SCLK high DOPI mode 00: 1.5 clock cycles 01: 2.5 clock cycles 10: 3.5 clock cycles 11: 4.5 clock cycles Other mode 00: 2 clock cycles 01: 3 clock cycles 10: 4 clock cycles 11: 5 clock cycles
29 to 27	DVWHI1[2:0]	000	R/W	Device 1 select signal pull-up timing setting* Indicates the timing, from the end of write operation to the select signal is pulled up. Timing definition from the last OM_SCLK low to OM_CS1# high DOPI mode 000: 1.5 clock cycles 001: 2.5 clock cycles 010: 3.5 clock cycles 011: 4.5 clock cycles 100: 5.5 clock cycles 101: 6.5 clock cycles 110: 7.5 clock cycles 111: 8.5 clock cycles Other mode 000: 2 clock cycles 001: 3 clock cycles 010: 4 clock cycles 011: 5 clock cycles 100: 6 clock cycles 101: 7 clock cycles 110: 8 clock cycles 111: 9 clock cycles

Bit	Bit Name	Initial Value	R/W	Description
26 to 24	DVWCMD1[2:0]	000	R/W	Device 1 Command execution interval setting* Indicates the timing between command to command for the device 1. Timing definition from OM_CS1# high to the next OM_CS1# low 000: 2 clock cycles 001: 5 clock cycles 010: 7 clock cycles 011: 9 clock cycles 100: 11 clock cycles 101: 13 clock cycles 110: 15 clock cycles 111: 17 clock cycles
23 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15, 14	DVWLO0[1:0]	00	R/W	Device 0 select signal pull-down timing setting* Indicates the timing between the select signal pull-down to the write operation for the device 0. Timing definition from OM_CS0# low to the first OM_SCLK high DOPI mode 00: 1.5 clock cycles 01: 2.5 clock cycles 10: 3.5 clock cycles 11: 4.5 clock cycles Other mode 00: 2 clock cycles 01: 3 clock cycles 10: 4 clock cycles 11: 5 clock cycles
13 to 11	DVWHI0[2:0]	000	R/W	Device 0 select signal pull-up timing setting* Indicates the timing, from the end of write operation to the select signal is pulled up. Timing definition from the last OM_SCLK low to OM_CS0# high DOPI mode 000: 1.5 clock cycles 001: 2.5 clock cycles 010: 3.5 clock cycles 011: 4.5 clock cycles 100: 5.5 clock cycles 101: 6.5 clock cycles 110: 7.5 clock cycles 111: 8.5 clock cycles Other mode 000: 2 clock cycles 001: 3 clock cycles 010: 4 clock cycles 011: 5 clock cycles 100: 6 clock cycles 101: 7 clock cycles 110: 8 clock cycles 111: 9 clock cycles
10 to 8	DVWCMD0[2:0]	000	R/W	Device 0 Command execution interval setting* Indicates the timing between command to command for the device 0. Timing definition from OM_CS0# high to the next OM_CS0# low 000: 2 clock cycles 001: 5 clock cycles 010: 7 clock cycles 011: 9 clock cycles 100: 11 clock cycles 101: 13 clock cycles 110: 15 clock cycles 111: 17 clock cycles
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: These clock cycles are values in internal clock units. In OM_SCLK units, the clock cycle is 1/2 of the above table.

22.4.12 Device Chip Select Timing Setting Register (DCSTR)

DCSTR sets the timing of operations in configuration mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DVSELLO[1:0]		DVSELHI[2:0]			DVSELCMD[2:0]			—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15, 14	DVSELLO[1:0]	All 0	R/W	Device select signal pull-down timing setting* Indicates the timing, from the device selection signal is pulled-down to the command execution. Timing definition from OM_CS# low to the first OM_SCLK high DOPI mode 00: 1.5 clock cycles 01: 2.5 clock cycles 10: 3.5 clock cycles 11: 4.5 clock cycles Other mode 00: 2 clock cycles 01: 3 clock cycles 10: 4 clock cycles 11: 5 clock cycles
13 to 11	DVSELHI[2:0]	000	R/W	Device select signal pull-up timing setting* Indicates the timing when the select signal is pulled up from the end of the command execution. Timing definition from the last OM_SCLK low to OM_CS# high DOPI or (SOPI) mode 000: Setting prohibited 001: Setting prohibited 010: Setting prohibited 011: Setting prohibited 100: Setting prohibited 101: 6.5 (7) clock cycles 110: 7.5 (8) clock cycles 111: 8.5 (9) clock cycles Note: The values in parentheses apply to SOPI mode. Other mode 000: 2 clock cycles 001: 3 clock cycles 010: 4 clock cycles 011: 5 clock cycles 100: 6 clock cycles 101: 7 clock cycles 110: 8 clock cycles 111: 9 clock cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	DVSELCMD[2:0]	000	R/W	Device Command execution interval setting* Indicates the timing between command to command. Timing definition from OM_CS# high to the next OM_CS# low 000: 2 clock cycles 001: 5 clock cycles 010: 7 clock cycles 011: 9 clock cycles 100: 11 clock cycles 101: 13 clock cycles 110: 15 clock cycles 111: 17 clock cycles
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: These clock cycles are values in internal clock units. In OM_SCLK units, the clock cycle is 1/2 of the above table.

22.4.13 Controller and Device Setting Register (CDSR)

CDSR controls the automatic calibration, pre-cycle setting, and device transfer type of the controller and each of the devices.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DLFT	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	ACMODE [1:0]		ACMEME [1:0]		—	—	—	—	DV1PC	DV0PC	DV1TTYP[1:0]		DV0TTYP[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

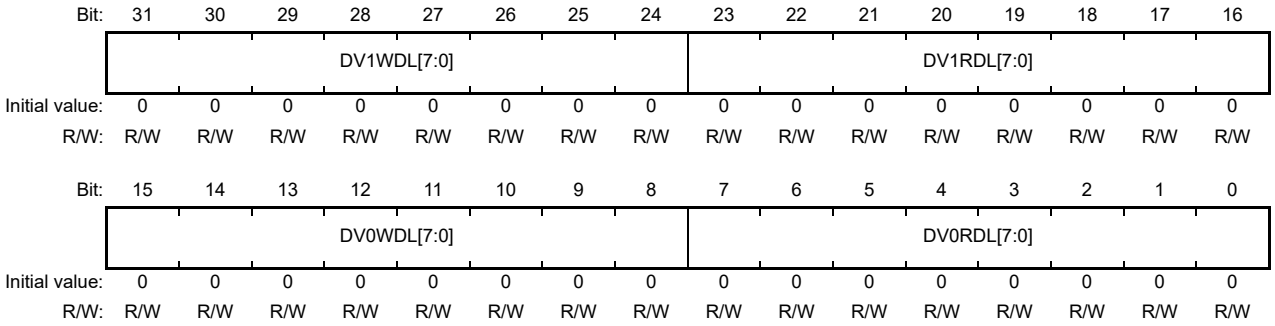
Bit	Bit Name	Initial Value	R/W	Description
31	DLFT	0	R/W	Deadlock Free Timer Enable Enable the timer to prevent the occurrence of controller deadlock. The timeout status is reset only by a system reset. 0: Enable timer 1: Disable timer Note: This function is available only when OM_SCLK is 132 MHz. Otherwise, disable the timer.
30 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	ACMODE[1:0]	00	R/W	Automatic calibration mode 00: Automatic calibration is disabled.* 01: Automatic calibration is enabled. 10: Setting prohibited. 11: Setting prohibited. The ACMODE bits should be set to disabled during read access to the OctaFlash device. Automatic calibration is only available in the DOPI mode.
11, 10	ACMEME[1:0]	00	R/W	Automatic calibration memory enable setting Automatic calibration supports only DOPI mode. To end DOPI mode, disable automatic calibration. [1]: Enable for device 1 1: Enable 0: Disable [0]: Enable for device 0 1: Enable 0: Disable
9 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	DV1PC	0	R/W	Device1_memory precycle setting 0: Disable 1: Enable
4	DV0PC	0	R/W	Device0_memory precycle setting 0: Disable 1: Enable
3, 2	DV1TTYP[1:0]	00	R/W	Device1_transfer_type setting 00: SPI mode 01: SOPI mode 10: DOPI mode 11: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
1, 0	DV0TTYP[1:0]	00	R/W	Device0_transfer_type setting 00: SPI mode 01: SOPI mode 10: DOPI mode 11: Setting prohibited

Note: When automatic calibration is disabled, the device 0 delay (OctaFlash) and device 1 delay (OctaRAM) should be manually specified in the memory delay trim register (MDTR) (see section 22.4.6). Set this register with reference to "initial setting MDTR (OctaFlash memory)" and "initial setting MDTR (OctaRAM)" in Figure 22.16.

22.4.14 Memory Map Dummy Length Register (MDLR)

MDLR sets the number of dummy cycles (latency) for writing and reading the memory map.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DV1WDL[7:0]	All 0	R/W	Device 1 Write dummy cycles (latency) setting Set the dummy length when writing to device 1 in OM_SCLK cycles unit.
23 to 16	DV1RDL[7:0]	All 0	R/W	Device 1 Read dummy cycles (latency) setting Set the dummy length when reading to device 1 in OM_SCLK cycles unit.
15 to 8	DV0WDL[7:0]	All 0	R/W	Device 0 Write dummy cycles (latency) setting Set the dummy length when writing to device 0 in OM_SCLK cycles unit.
7 to 0	DV0RDL[7:0]	All 0	R/W	Device 0 Read dummy cycles (latency) setting Set the dummy length when reading to device 0 in OM_SCLK cycles unit.

Note: When changing the dummy cycles (latency) for the OctaFlash device, modify the setting on the memory and then modify the setting in the DMLEN[7:0] bits in the device command setting register (DCSR) so that the dummy length becomes the same between the memory and this Octa memory controller. After that, execute the status read command to confirm the memory status.

22.4.15 Memory Map Read/Write Command Register 0 (MRWCR0)

MRWCR0 sets the read/write commands for the device 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	D0MWCMD1[7:0]								D0MWCMD0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D0MRCMD1[7:0]								D0MRCMD0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	D0MWCMD1[7:0]	All 0	R/W	Memory map write command 1 setting, Set the memory map write command. According to the flash memory spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will not send command 1. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 0 is specified by CDSR.DV0TTYP[1:0].
23 to 16	D0MWCMD0[7:0]	All 0	R/W	Memory map write command 0 setting, Set the memory map write command. According to the flash memory spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will send only command 0. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 0 is specified by CDSR.DV0TTYP[1:0].
15 to 8	D0MRCMD1[7:0]	All 0	R/W	Memory map read command 1 setting Set the memory map read command. According to the flash memory spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will not send command 1. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 0 is specified by CDSR.DV0TTYP[1:0].
7 to 0	D0MRCMD0[7:0]	All 0	R/W	Memory map read command 0 setting, Set the memory map read command. According to the flash memory spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will send only command 0. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 0 is specified by CDSR.DV0TTYP[1:0].

22.4.16 Memory Map Read/Write Command Register 1 (MRWCR1)

MRWCR1 sets the read/write commands for the device 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	D1MWCMD1[7:0]								D1MWCMD0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D1MRCMD1[7:0]								D1MRCMD0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	D1MWCMD1[7:0]	All 0	R/W	Memory map write command 1 setting Set the memory map write command. According to the RAM spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will not send command 1. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 1 is specified by CDSR.DV1TTYP[1:0].
23 to 16	D1MWCMD0[7:0]	All 0	R/W	Memory map write command 0 setting Set the memory map write command. According to the RAM spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will send only command 0. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 1 is specified by CDSR.DV1TTYP[1:0].
15 to 8	D1MRCMD1[7:0]	All 0	R/W	Memory map read command 1 setting Set the memory map read command. According to the RAM spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will not send command 1. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 1 is specified by CDSR.DV1TTYP[1:0].
7 to 0	D1MRCMD0[7:0]	All 0	R/W	Memory map read command 0 setting Set the memory map read command. According to the RAM spec, there are SPI, SOPI, and DOPI mode. In SPI mode, the controller will send only command 0. In SOPI and DOPI mode, the controller will send command 1 first and then send command 0. The mode of device 1 is specified by CDSR.DV1TTYP[1:0].

22.4.17 Memory Map Read/Write Setting Register (MRWCSR)

MRWCSR sets the memory map read/write operations.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	MWO1	MWCL1[2:0]			MWAL1[2:0]			—	MRO1	MRCL1[2:0]			MRAL1[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	MWO0	MWCL0[2:0]			MWAL0[2:0]			—	MRO0	MRCL0[2:0]			MRAL0[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	MWO1	0	R/W	Device 1 write order setting 0: Write order is byte0, byte1, byte2, byte3. 1: Write order is byte1, byte0, byte3, byte2.
29 to 27	MWCL1[2:0]	000	R/W	Device 1 write command length setting Set the number of bytes in the write command for device 1.
26 to 24	MWAL1[2:0]	000	R/W	Device 1 write address length setting Set the memory map write address byte length for device 1.
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	MRO1	0	R/W	Device 1 read order setting 0: Read order is byte0, byte1, byte2, byte3. 1: Read order is byte1, byte0, byte3, byte2. Note: This setting is only valid in the DOPI mode. It has no effect in the SOPI or SPI mode.
21 to 19	MRCL1[2:0]	000	R/W	Device 1 read command length setting Set the read command byte length for device 1.
18 to 16	MRAL1[2:0]	000	R/W	Device 1 read address length setting Set the memory map read address byte length for device 1.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	MWO0	0	R/W	Device 0 write order setting 0: Write order is byte0, byte1, byte2, byte3. 1: Write order is byte1, byte0, byte3, byte2.
13 to 11	MWCL0[2:0]	000	R/W	Device 0 write command length setting Set the write command byte length for device 0.
10 to 8	MWAL0[2:0]	000	R/W	Device 0 write address length setting Set the memory map write address byte length for device 0.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	MRO0	0	R/W	Device 0 read order setting 0: Read order is byte0, byte1, byte2, byte3. 1: Read order is byte1, byte0, byte3, byte2. Note: This setting is only valid in the DOPI mode. It has no effect in the SOPI or SPI mode.
5 to 3	MRCL0[2:0]	000	R/W	Device 0 read command length setting Set the write command byte length for device 0.
2 to 0	MRAL0[2:0]	000	R/W	Device 0 read address length setting Set the memory map read address byte length for device 0.

22.4.18 Error Status Register (ESR)

ESR indicates the error status when performing memory map read and write commands.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MWESR[7:0]								MRESR[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	0	R	Reserved These bits are always read as 0.
15 to 8	MWESR[7:0]	All 0	R	Memory map write error status H'80 = invalid command, H'01 = write data length error
7 to 0	MRESR[7:0]	All 0	R	Memory map read error status H'80 = invalid command, H'01 = ECC error, H'02 = preamble error, H'03 = wait OM_DQS timeout

22.4.19 Configure Write without Data Register (CWNDR)

CWNDR is used to execute a write command without data in the configuration mode. Writing 0 to this register makes this module issue a write command without actually writing data to memory.
Specify an address in DAR, a command in DCSR, and command data in DCR.
Set the DCSR.DALEN bit field to 0 before writing to this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	—	All 0	W	The write value should always be 0.

22.4.20 Configure Write Data Register (CWDR)

CWDR is used to set the data for writing when a write command is to be executed in the configuration mode. Writing to this register makes the controller to issue a write command and transfer the data to be written. The byte order of data transferred by the controller to the memory is in accord with the setting of the DAOR bit in DCSR. Specify an address in DAR, a command in DCSR, and command data in DCR.
Set the DCSR.DALEN bit field to a non-zero value before writing to this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WD3[7:0]								WD2[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WD1[7:0]								WD0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	WD3[7:0]	All 0	W	Write data 3 Set the third byte of the write data.
23 to 16	WD2[7:0]	All 0	W	Write data 2 Set the second byte of the write data.
15 to 8	WD1[7:0]	All 0	W	Write data 1 Set the first byte of the write data.
7 to 0	WD0[7:0]	All 0	W	Write data 0 Set the zeroth byte of the write data.

22.4.21 Configure Read Register (CRR)

CRR is used to execute a read command in the configuration mode and holds the data that are read. Reading from this register makes the controller issue a read command and the data that are read are stored in this register. Specify an address in DAR, a command in DCSR, and command data in DCR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RD3[7:0]								RD2[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RD1[7:0]								RD0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RD3[7:0]	All 0	R	Read data 3 Store the third byte of the read data.
23 to 16	RD2[7:0]	All 0	R	Read data 2 Store the second byte of the read data.
15 to 8	RD1[7:0]	All 0	R	Read data 1 Store the first byte of the read data.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	RD0[7:0]	All 0	R	Read data 0 Store the zeroth byte of the read data.

22.5 Operation

22.5.1 Octa Memory Controller System Configuration

Figure 22.2 shows an example of connections between the Octa memory controller in this LSI and OctaFlash and OctaRAM devices.
CS0# should be connected to the OctaFlash device and CS1# should be connected to the OctaRAM device.

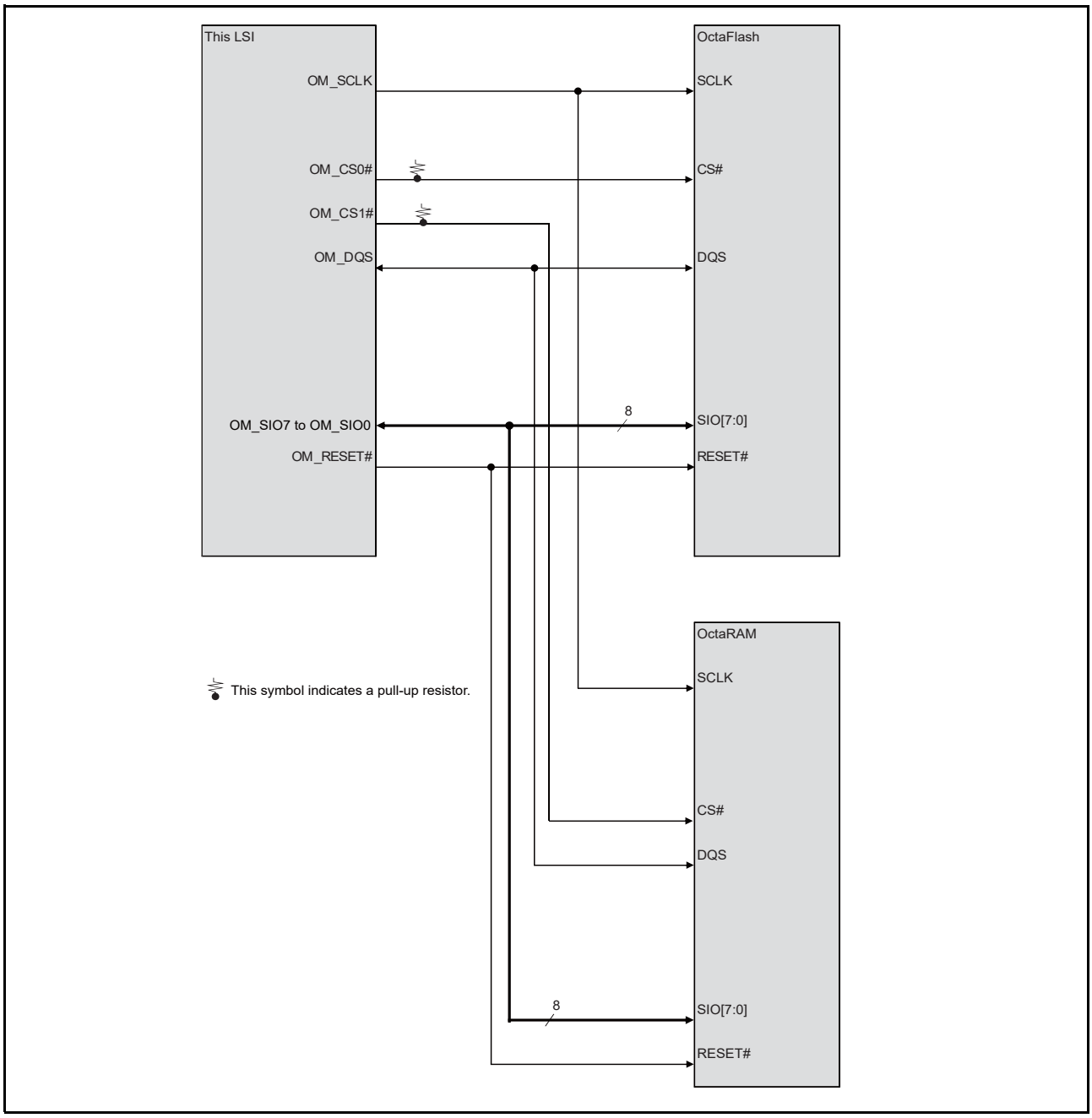


Figure 22.2 Example of Connections between this LSI and OctaFlash and OctaRAM Devices

22.5.2 Address Map

In the memory-map mode, OctaFlash is allocated to the OctaFlash space (H'50000000 to H'5FFFFFFF), and OctaRAM is allocated to the OctaRAM space (H'60000000 to H'6FFFFFFF).
One OctaFlash device and one OctaRAM device can be connected to this LSI, and up to 256 Mbytes can be accessed in each device.

	Internal Address	Maximum Accessible Area
OctaFlash	H'50000000 to H'5FFFFFFF	Up to 256 Mbytes
OctaRAM	H'60000000 to H'6FFFFFFF	Up to 256 Mbytes

22.5.3 Octa Memory Interface

This section describes the Octa memory interface.

22.5.3.1 Write Operation

The Octa memory controller outputs a command and an address and writes one or more data bytes by using lines OM_CS0# (OM_CS1#), OM_SCLK, and OM_SIO7 to OM_SIO0.
The write operation for OctaRAM has latency cycles before data is written to memory.
Figure 22.3 to Figure 22.5 show waveforms of the write operation in the OctaFlash interface, and Figure 22.6 shows a waveform of the write operation in the OctaRAM interface.

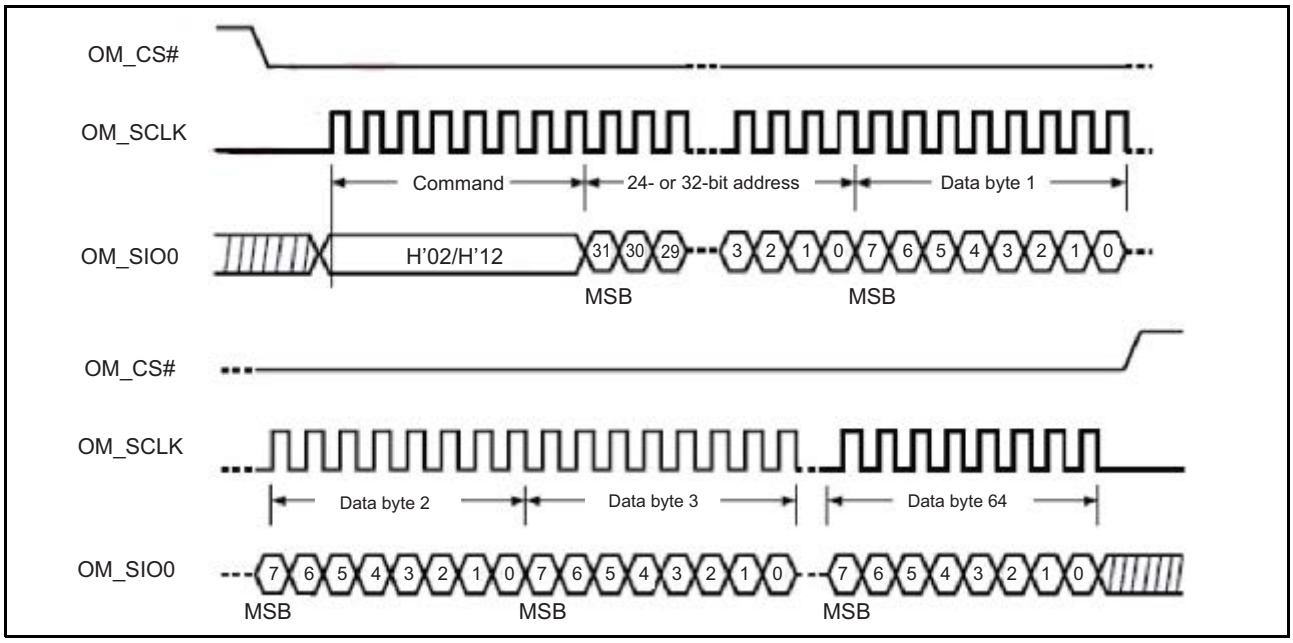


Figure 22.3 Waveform of Write Operation in OctaFlash Interface (SPI Mode)

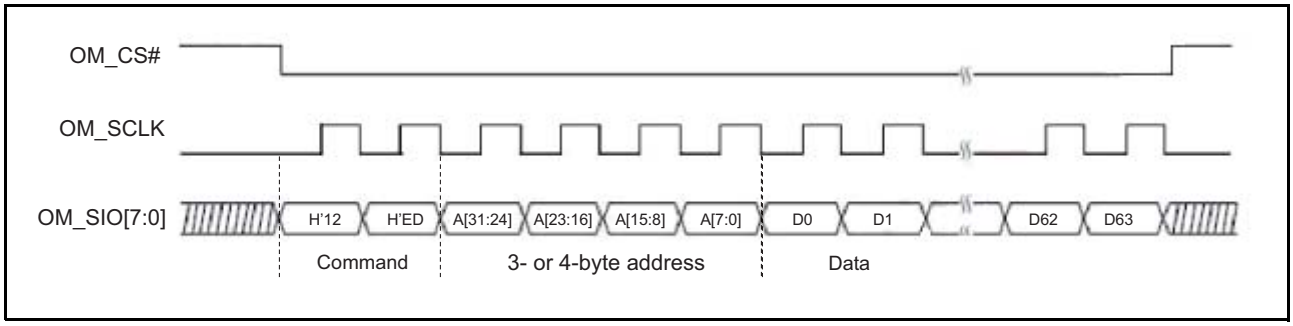


Figure 22.4 Waveform of Write Operation in OctaFlash Interface (SOPI Mode)

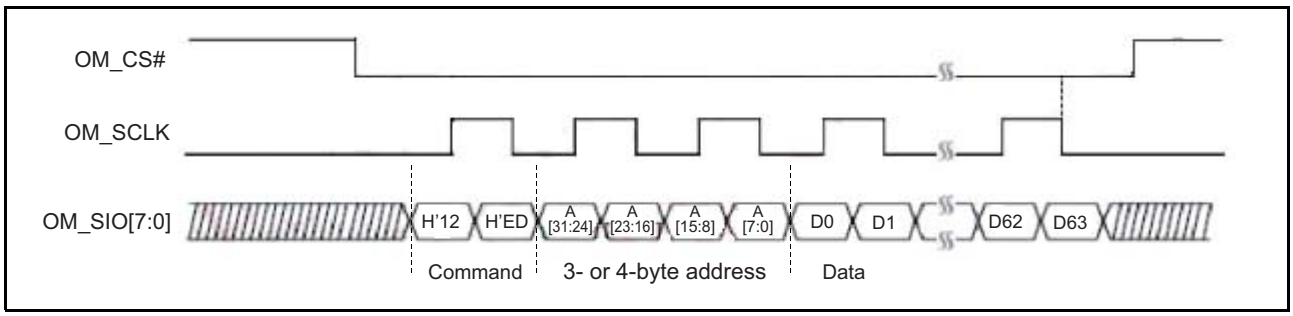


Figure 22.5 Waveform of Write Operation in OctaFlash Interface (DOPI Mode)

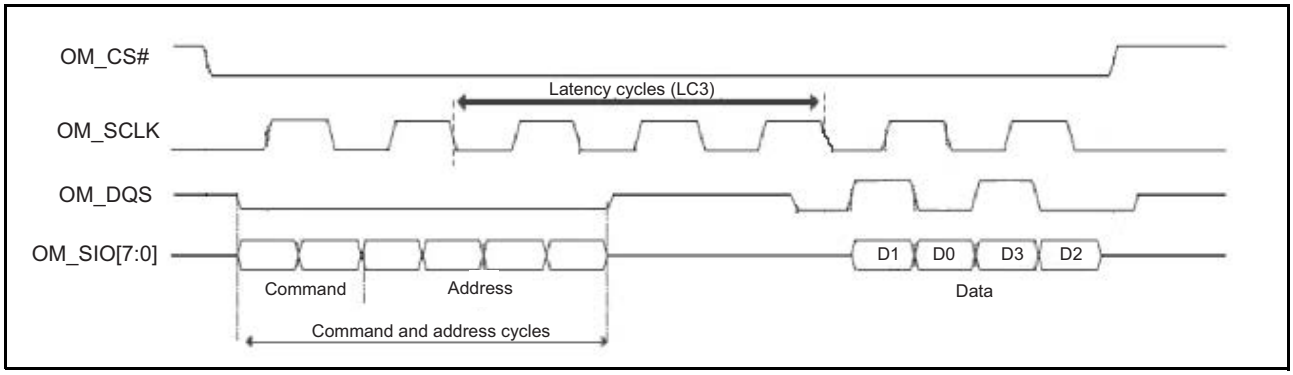


Figure 22.6 Waveform of Write Operation in OctaRAM Interface

22.5.3.2 Read Operation

The Octa memory controller outputs a command and an address and reads one or more data bytes by using lines OM_CS0# (OM_CS1#), OM_SCLK, and OM_SIO7 to OM_SIO0.
The read operation for OctaFlash and OctaRAM has latency cycles before data is read from memory.
The number of latency cycles should be specified in the memory map dummy length setting register (MDLR) (see section 22.4.14) in accordance with the setting in the memory device.
Figure 22.7 to Figure 22.9 show waveforms of the read operation in the OctaFlash interface, and Figure 22.10 shows a waveform of the read operation in the OctaRAM interface.

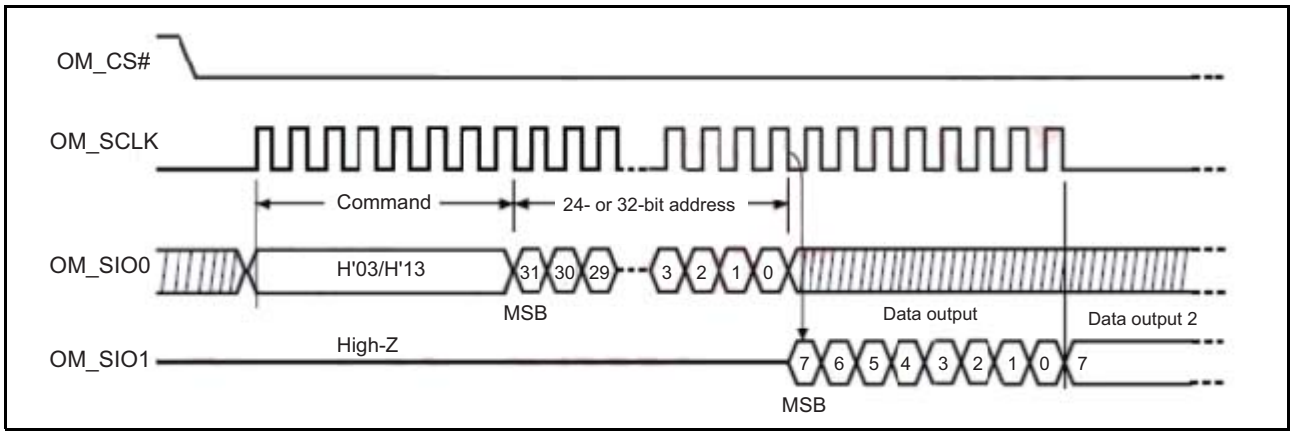


Figure 22.7 Waveform of Read Operation in OctaFlash Interface (SPI Mode)

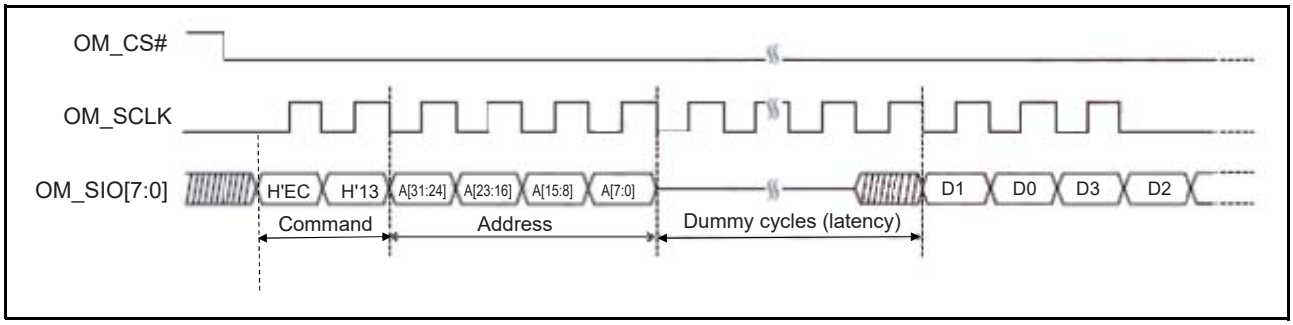


Figure 22.8 Waveform of Read Operation in OctaFlash Interface (SOPI Mode)

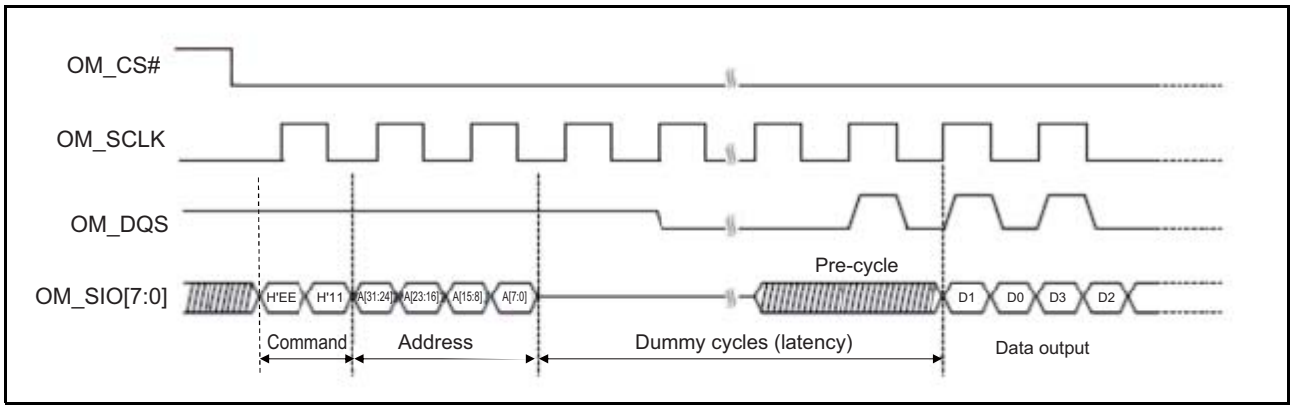


Figure 22.9 Waveform of Read Operation in OctaFlash Interface (DOPI Mode with Pre-cycle Enabled)

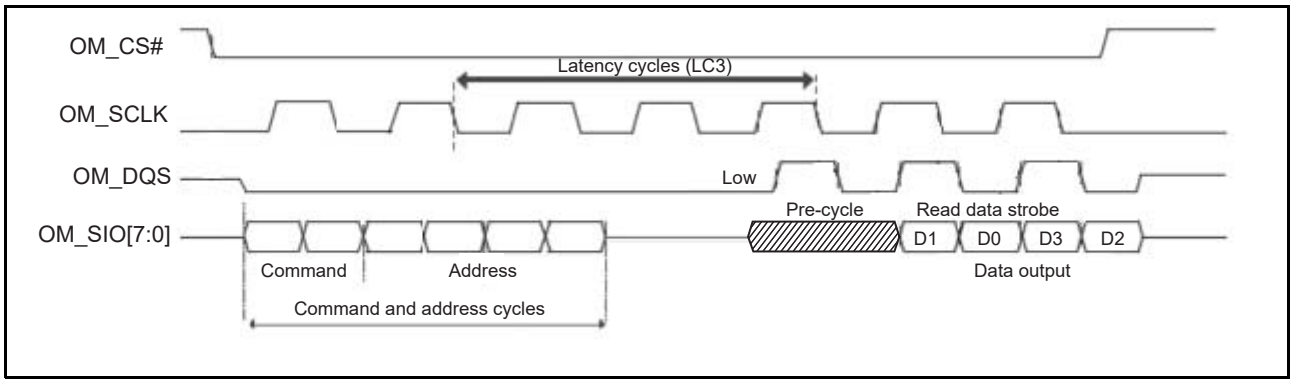


Figure 22.10 Waveform of Read Operation in OctaRAM Interface (with Pre-cycle Enabled)

22.5.4 Data Alignment in the OctaFlash and OctaRAM Spaces

Figure 22.11 shows the memory data alignment in the OctaFlash and OctaRAM spaces.
Figure 22.12 shows the data alignment when the configuration register in OctaFlash or OctaRAM is accessed.

When OctaFlash is connected (MRWCSR.MWO0 = 0, MRWCSR.MRO0 = 0) or
OctaRAM is connected (MRWCSR.MWO1 = 0, MRWCSR.MRO1 = 0)

Flash/RAM 8-bit DDR	Access width	Address	Internal bus 63–56	55–48	47–40	39–32	31–24	23–16	15–8	7–0
0 D0[7:0]	1 byte	b'000	X	X	X	X	X	X	X	D0
1 D1[7:0]		b'001	X	X	X	X	X	X	D1	X
2 D2[7:0]		b'010	X	X	X	X	X	D2	X	X
3 D3[7:0]		b'011	X	X	X	X	D3	X	X	X
4 D4[7:0]		b'100	X	X	X	D4	X	X	X	X
5 D5[7:0]		b'101	X	X	D5	X	X	X	X	X
6 D6[7:0]		b'110	X	D6	X	X	X	X	X	X
7 D7[7:0]		b'111	D7	X	X	X	X	X	X	X
↕ OM_SIO[7:0]	2 bytes	b'000	X	X	X	X	X	X	D1	D0
		b'010	X	X	X	X	D3	D2	X	X
		b'100	X	X	D5	D4	X	X	X	X
		b'110	D7	D6	X	X	X	X	X	X
	4 bytes	b'000	X	X	X	X	D3	D2	D1	D0
		b'100	D7	D6	D5	D4	X	X	X	X
	8 bytes	b'000	D7	D6	D5	D4	D3	D2	D1	D0

When OctaFlash is connected (MRWCSR.MWO0 = 1, MRWCSR.MRO0 = 1) or
OctaRAM is connected (MRWCSR.MWO1 = 1, MRWCSR.MRO1 = 1)

Flash/RAM 8bit DDR	Access width	Address	Internal bus 63–56	55–48	47–40	39–32	31–24	23–16	15–8	7–0
0 D0[7:0]	1 byte	b'000	X	X	X	X	X	X	X	D1
1 D1[7:0]		b'001	X	X	X	X	X	X	D0	X
2 D2[7:0]		b'010	X	X	X	X	X	D3	X	X
3 D3[7:0]		b'011	X	X	X	X	D2	X	X	X
4 D4[7:0]		b'100	X	X	X	D5	X	X	X	X
5 D5[7:0]		b'101	X	X	D4	X	X	X	X	X
6 D6[7:0]		b'110	X	D7	X	X	X	X	X	X
7 D7[7:0]		b'111	D6	X	X	X	X	X	X	X
↕ OM_SIO[7:0]	2 bytes	b'000	X	X	X	X	X	X	D0	D1
		b'010	X	X	X	X	D2	D3	X	X
		b'100	X	X	D4	D5	X	X	X	X
		b'110	D6	D7	X	X	X	X	X	X
	4 bytes	b'000	X	X	X	X	D2	D3	D0	D1
		b'100	D6	D7	D4	D5	X	X	X	X
	8 bytes	b'000	D6	D7	D4	D5	D2	D3	D0	D1

Figure 22.11 Memory Data Alignment in the OctaFlash and OctaRAM Spaces

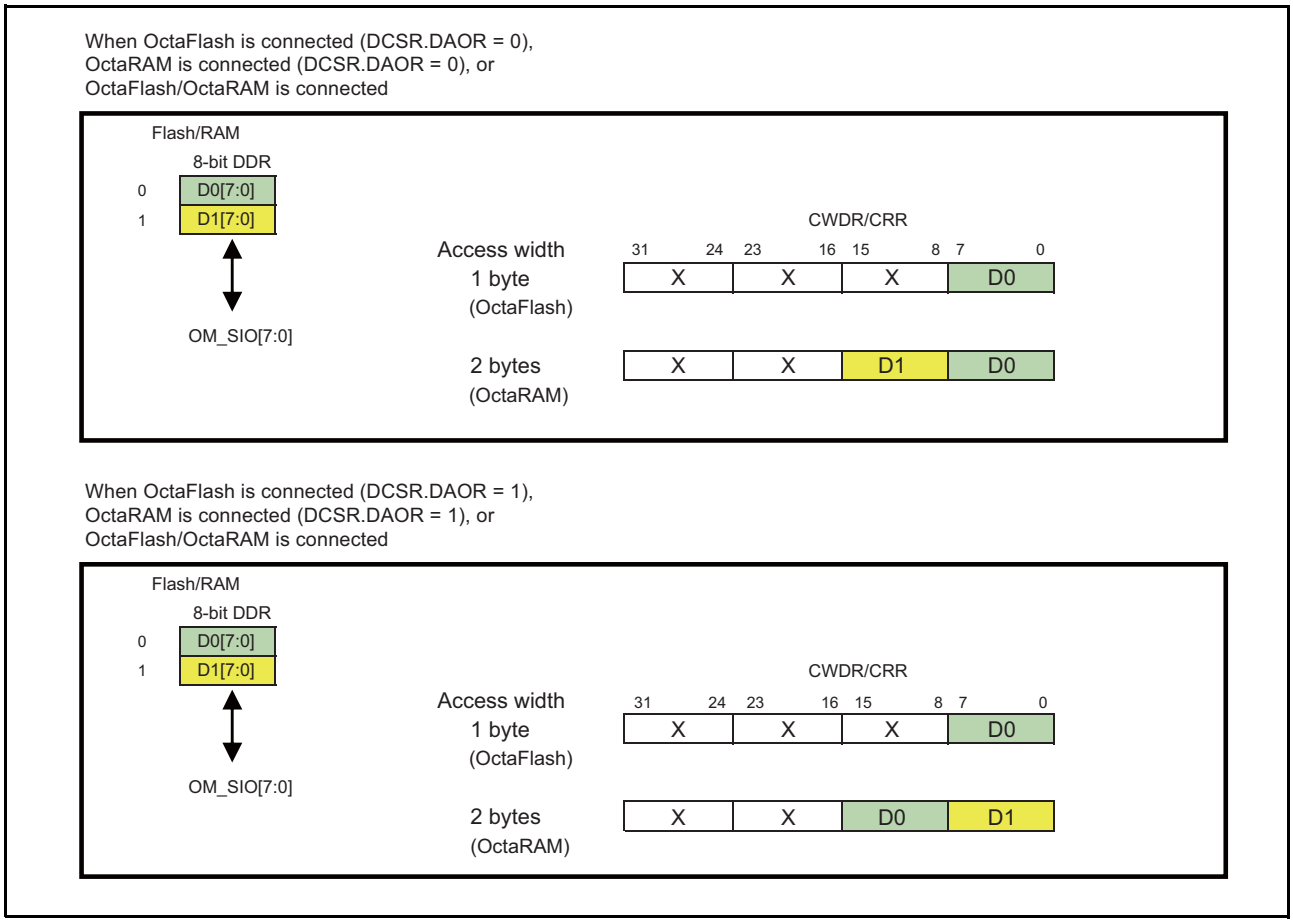


Figure 22.12 Configuration Register Data Alignment in the OctaFlash and OctaRAM Spaces

22.5.5 One-Byte Write Access to an Octa Memory Device in the DOPI Mode

This section describes one-byte write access in the DOPI mode.
Writing to the Octa memory device in the DOPI mode is handled in 2-byte units.
When the target device is OctaRAM, the output OM_DQS signal of the controller can be used to mask bytes by driving it to the high level during the operation of writing. This allows the masking of invalid write data.
When the target device is OctaFlash, one-byte write access must not be attempted since HyperFlash only supports two-byte access.
Figure 22.13 shows the waveforms in one-byte write access in the DOPI mode.

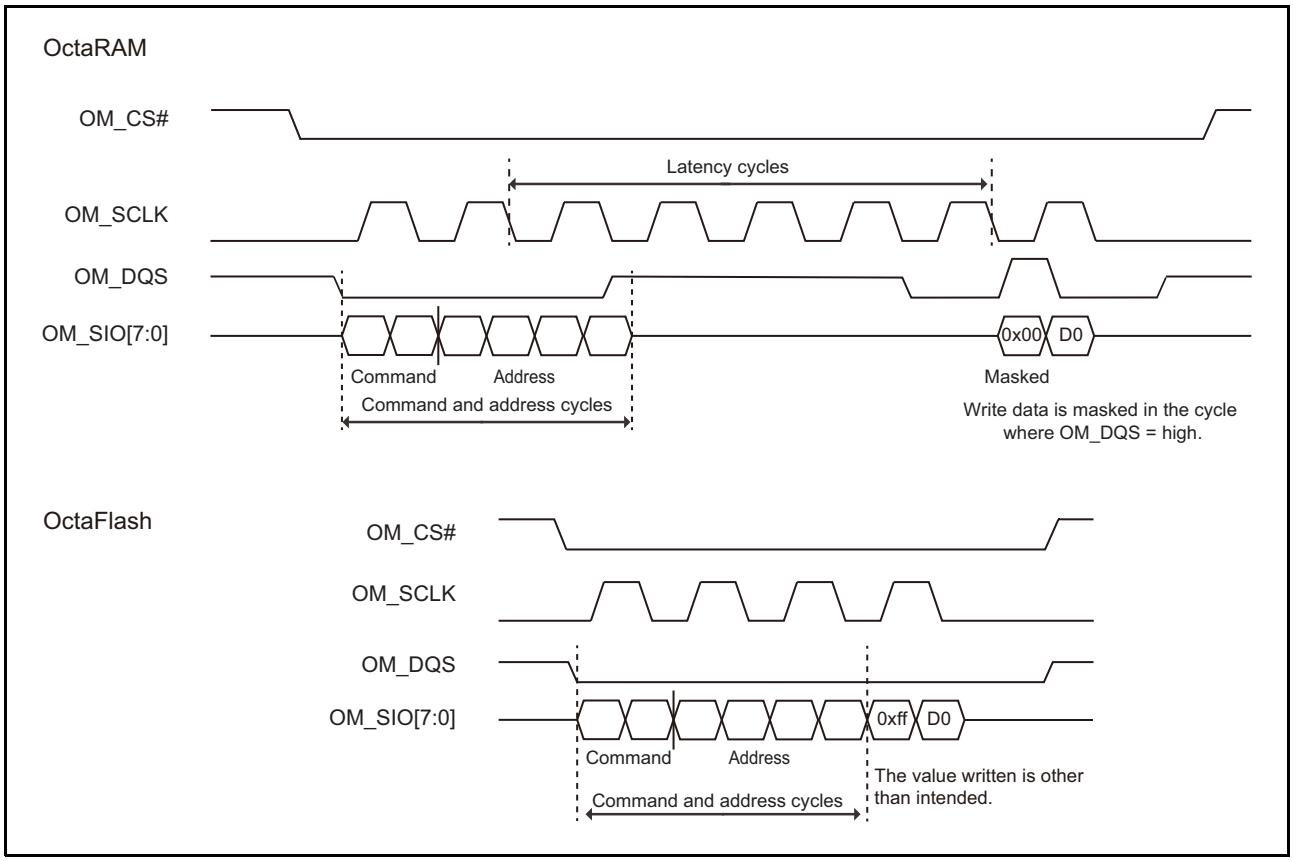


Figure 22.13 Waveforms in One-Byte Write Access in the DOPI Mode

22.5.6 Operation Flows

This section describes the flows of Octa memory controller operations for various purposes.
The following only shows sample settings, and appropriate commands and data should be specified in accordance with the specifications of the target memory device.
The sample settings assume the following environment.
(1) Memory connections: CS0 (device 0) to OctaFlash and CS1 (device 1) to OctaRAM
(2) OM_SCLK (Octa memory clock) frequency: 132 MHz

22.5.6.1 Initial Settings

The following shows an example of initial settings of the Octa memory controller.
Before access to the external memory, set up the HOSEL bit in the HyperBus controller and Octa memory controller dedicated pin control register (PHMOM0) in the initial setting steps.
For the HOSEL bit, refer to section 51, GPIO.

Table 22.6 Example of Initial Settings of the Octa Memory Controller

Step	Read/Write	Address (Register)	Data	Data Length (Bytes)	Description
1	Write	DSR0	H'08000000	4	Specifies the type and size of device 0.
2	Write	DSR1	H'40800000	4	Specifies the type and size of device 1.
3	Write	MDTR	H'0740A640	4	Adjusts the memory access timings.
4	Write	DRCSTR	H'29002900	4	Specifies the read timing for each device.
5	Write	DWCSTR	H'1A001A00	4	Specifies the write timing for each device.
6	Write	DCSTR	H'00002900	4	Specifies the timings of the chip select signals.
7	Write	CDSR	H'0000000A	4	Sets up the controller functions. (1) Memory pre-cycle is enabled or disabled. (2) Transfer type is specified (SPI, SOPI, or DOPI).
8	Write	MDLR	H'05050E0E	4	Specifies number of dummy cycles (latency).
9	Write	MRWCR0	H'12EDEE11	4	Specifies the read and write commands for device 0.
10	Write	MRWCR1	H'2000A000	4	Specifies the read and write commands for device 1.
11	Write	MRWCSR	H'54545454	4	Sets up the memory-mapped reading and writing. (1) The length (bytes) of a command is specified. (2) The length (bytes) of an address is specified. (3) The order of bytes to be written is specified.

Note: Set up the HOSEL bit in the HyperBus controller and Octa memory controller dedicated pin control register (PHMOM0). For the HOSEL bit, refer to section 51, GPIO.

22.5.6.2 Basic Operation Settings

This section describes the basic operations of OctaFlash and OctaRAM.
The following only shows sample settings, and appropriate commands and data should be specified in accordance with the specifications of the target memory device.

(1) OctaFlash Operation Settings

1. Write Enable Settings

			SPI Mode		OPI Mode		
Step	Read/ Write	Register	Data	Data Length (Bytes)	Data	Data Length (Bytes)	Description
1	Write	DCR	H'00000006	4	H'000006F9	4	
2	Write	DCSR	H'00100000	4	H'00200000	4	
3	Write	CWNDR	Any value	4	Any value	4	

2. Read Status Register

			SPI Mode		OPI Mode		
Step	Read/ Write	Register	Data	Data Length (Bytes)	Data	Data Length (Bytes)	Description
1	Write	DCR	H'00000005	4	H'000005FA	4	
2	Write	DAR	—	—	H'00000000	4	
3	Write	DCSR	H'00100001	4	H'0C200401	4	
4	Read	CRR	H'000000xx	1	H'000000xx	1	The high-order 24 bits are all set to 0s, and the read data is stored in the low-order eight bits.

3. Read Configuration Register

			SPI Mode		OPI Mode		
Step	Read/ Write	Register	Data	Data Length (Bytes)	Data	Data Length (Bytes)	Description
1	Write	DCR	H'00000015	4	H'000015EA	4	
2	Write	DAR	—	—	H'00000001	4	
3	Write	DCSR	H'00100001	4	H'0C200401	4	
4	Read	CRR	H'000000xx	1	H'000000xx	1	The high-order 24 bits are all set to 0s, and the read data is stored in the low-order eight bits.

4. Write Configuration Register

			SPI Mode		OPI Mode		
Step	Read/ Write	Register	Data	Data Length (Bytes)	Data	Data Length (Bytes)	Description
1	Write	DCR	-	-	H'000001FE	4	
2	Write	DAR	-	-	H'00000001	4	
3	Write	DCSR	-	-	H'04200001	4	
4	Write	CWDR	-	-	H'000000xx	1	Set all the high-order 24 bits to 0s, and specify the write data in the low-order eight bits.

5. Read Configuration Register 2

			SPI Mode		OPI Mode		
Step	Read/ Write	Register	Data	Data Length (Bytes)	Data	Data Length (Bytes)	Description
1	Write	DCR	H'00000071	4	H'0000718E	4	
2	Write	DAR	Address	4	Address	4	
3	Write	DCSR	H'04100001	4	H'0C200401	4	
4	Read	CRR	H'000000xx	1	H'000000xx	1	The high-order 24 bits are all set to 0s, and the read data is stored in the low-order eight bits.

6. Write Configuration Register 2

			SPI Mode		OPI Mode		
Step	Read/ Write	Register	Data	Data Length (Bytes)	Data	Data Length (Bytes)	Description
1	Write	DCR	H'00000072	4	H'0000728D	4	
2	Write	DAR	Address	4	Address	4	
3	Write	DCSR	H'04100001	4	H'04200001	4	
4	Write	CWDR	H'000000xx	1	H'000000xx	1	Set all the high-order 24 bits to 0s, and specify the write data in the low-order eight bits.

7. Memory-Map Write and Read Command Settings

			SPI Mode		OPI Mode		
Step	Read/ Write	Register	Data	Data Length (Bytes)	Data	Data Length (Bytes)	Description
1	Write	MRWCR0	H'00120013	4	H'12EDEE11	4	
2	Write	MRWCSR	H'00000C0C	4	H'00005454	4	

8. Erase Sector

			SPI Mode		OPI Mode		
Step	Read/ Write	Register	Data	Data Length (Bytes)	Data	Data Length (Bytes)	Description
1	Write	DCR	H'00000021	4	H'000021DE	4	
2	Write	DAR	Address to be erased	4	Address to be erased	4	
3	Write	DCSR	H'04100000	4	H'04200000	4	
4	Write	CWNDR	Any value	4	Any value	4	

(2) OctaRAM Operation Settings

1. Read Configuration Register

Step	Read/ Write	Register	Data	Data Length (Bytes)	Description
1	Write	DCR	H'0000C000	4	
2	Write	DAR	H'00040000	4	
3	Write	DCSR	H'04A80502	4	
4	Read	CRR	H'0000xxxx	2	The high-order 16 bits are all set to 0s, and the read data is stored in the low-order 16 bits.

2. Write Configuration Register

Step	Read/ Write	Register	Data	Data Length (Bytes)	Description
1	Write	DCR	H'00004000	4	
2	Write	DAR	H'00040000	4	
3	Write	DCSR	H'04A80002	4	
4	Write	CWDR	H'0000xxxx	2	Set all the high-order 16 bits to 0s, and specify the write data in the low-order 16 bits.

22.6 Delaying OM_DQS and Auto-Calibration

22.6.1 Delaying OM_DQS

Depending on the characteristics of the DOPI mode, the controller might not be able to latch data on the rising and falling edges of the OM_DQS clock signal. OM_DQS will then require a delay to allow the latching of data with sufficient margins for the setup and hold times. The DV1DEL[7:0] and DV0DEL[7:0] bits in the memory delay trimming register (MDTR) can be used to specify the amount of delay.
Figure 22.14 shows how a delayed OM_DQS becomes asserted at a good time for latching of the data.

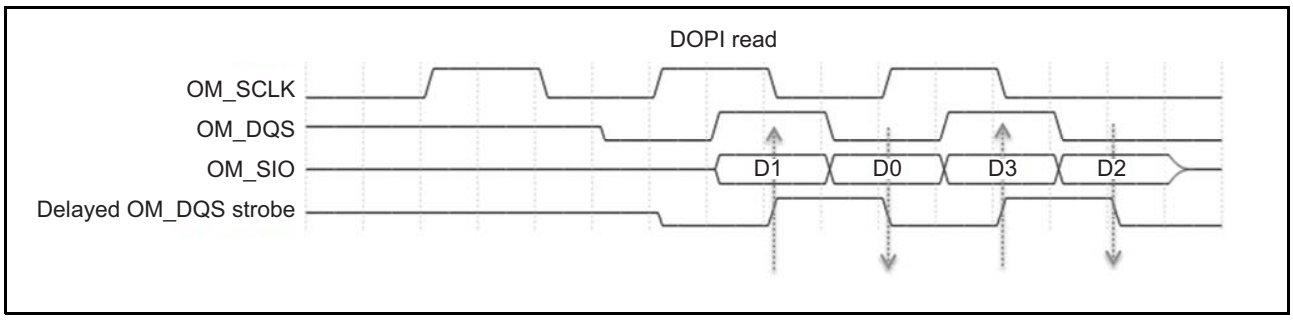


Figure 22.14 Example of Securing the Latching of Data by Delaying OM_DQS

22.6.2 OM_DQS Auto-Calibration

This LSI has an automatic calibration to prevent acquisition of wrong data due to the variation of voltage and temperature. When using the automatic calibration, set the following registers.

- Controller and device setting register (CDSR)
  - Automatic calibration mode ACMODE[1:0] bits setting
- Auto-calibration timer register (ACTR)
  - Automatic calibration cycle time setting
- Auto-calibration address registers 0 and 1 (ACAR0/ACAR1)
  - Preamble pattern store address setting

Automatic calibration requires 16 bytes user area to store the preamble pattern. Store the preamble pattern to the setting address of ACAR0 and ACAR1.

Preamble pattern: H'00F708F7 FF0000F7 0800FF00 FFFF0000
When MRWCSR.MRO1 or MRWCSR.MRO0 is 0,
H'00F708F7 FF0000F7 0800FF00 FFFF0000
When MRWCSR.MRO1 or MRWCSR.MRO0 is 1,
H'F700F708 00FFF700 000800FF FFFF0000
store in the flash memory or RAM device.

The operation proceeds as follows when auto-calibration is enabled.
Once the time specified in ACTR has elapsed, the Octa memory controller varies the delay time for OM_DQS and checks whether the preamble pattern can correctly be read from the address specified in ACAR0 or ACAR1.
The preamble pattern is read 48 times per auto-calibration.
The controller seeks the most appropriate timing of latching in this sequence of reading and updates the DV0DEL or DV1DEL bits in MDTR with the result. This re-starts the timer for auto-calibration.
Note that access to the OctaFlash memory or OctaRAM area by the CPU or other bus masters is held pending until reading of the preamble data for auto-calibration is completed.

Figure 22.15 describes the flow of automatic calibration settings for OctaFlash and OctaRAM devices.
Before starting settings, check that the Octa memory controller is selected in the HOSEL bit in the HyperBus controller and Octa memory controller dedicated pin control register (PHMOM0).

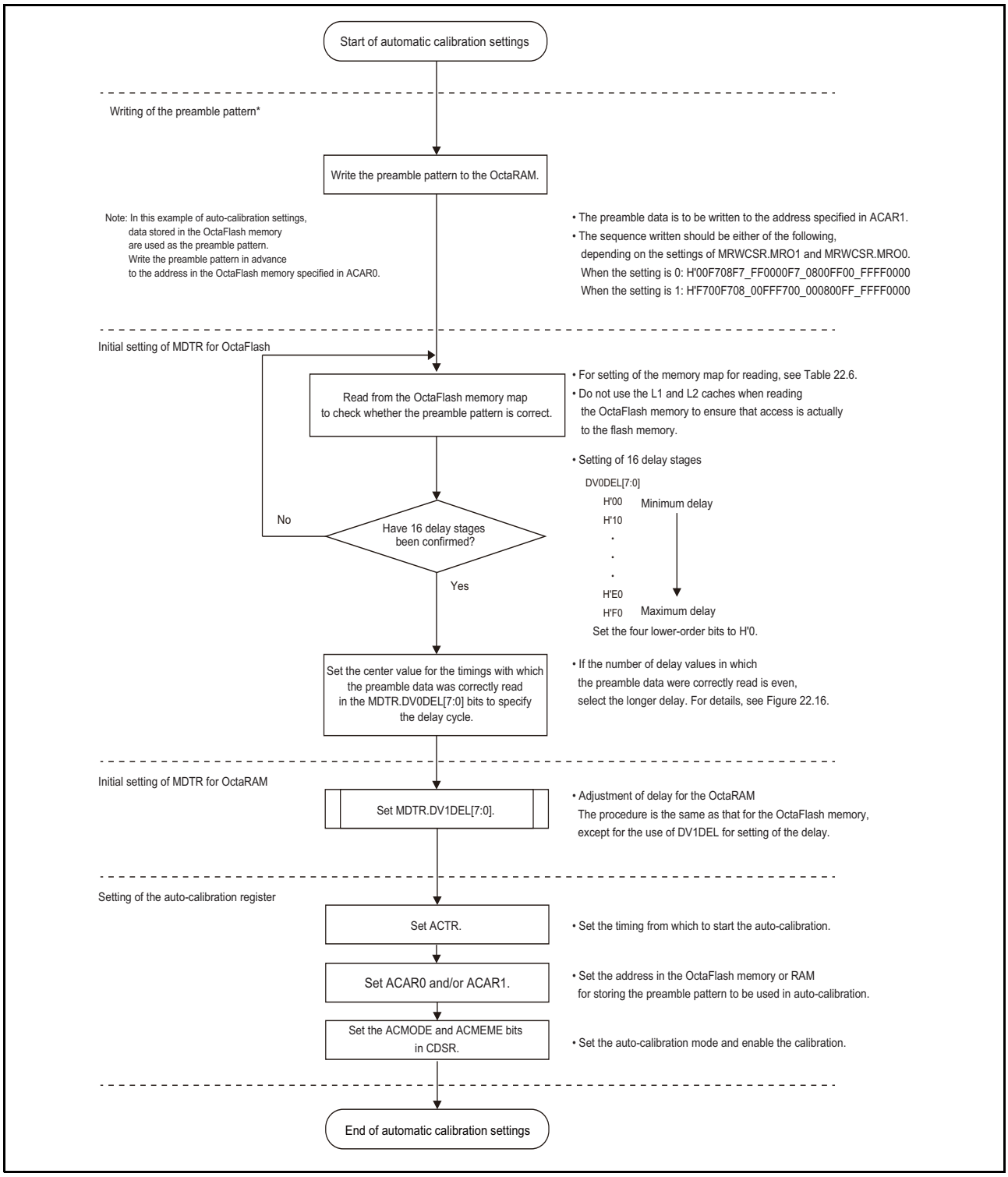


Figure 22.15 Flow of Settings for the Auto-Calibration Mode

Figure 22.16 shows an example of the initial setting of the DV0DEL or DV1DEL Bits in MDTR.

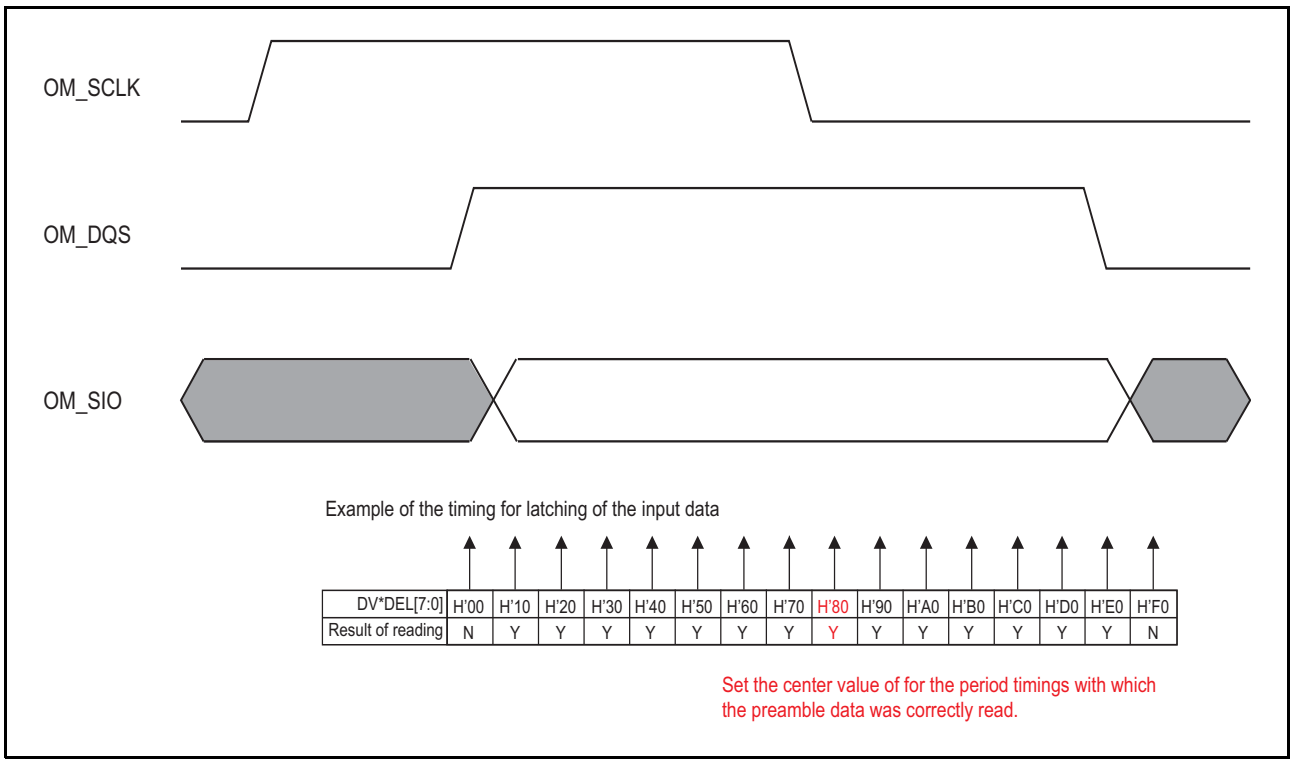


Figure 22.16 Example of the Initial Setting of the DV0DEL or DV1DEL Bits in MDTR

22.7 OM_DQS Enable Counter

During read command operation, the OM_DQS signal is used as the data latch clock by the controller. When the memory device is in the idle state, OM_DQS maintains a high impedance (Hi-Z) state, and asserts low before data is output. This Hi-Z to low transition may latch wrong data by driving the OM_DQS data latch logic of the controller.
Figure 22.15 shows the waveform when OM_DQS maintains a high-impedance (Hi-Z) state and asserts low before data is output.
Avoiding such Hi-Z to low transitions, this controller can bypass wrong data with memory delay trim register (MDTR). (MDTR.DQSEDOPI[3:0] for Flash DOPI mode, MDTR.DQSESOPI[3:0] for Flash SOPI mode, and MDTR.DQSERAM[3:0] for RAM mode)
Each of the bit fields can be set within the range from 0 to 15.

OM_DQS enable counter (MDTR) minimum value ≥ Number of the command cycles + number of the address cycles*1 + number of the stable OM_DQS cycles*2 + number of the OM_DQS delay cycles*3
OM_DQS enable counter (MDTR) maximum value ≤ Number of the command cycles + number of the address cycles*1 + number of the memory dummy cycles (latency) + number of the OM_DQS delay cycles*3 + pre-cycle setting*4

Note 1. Flash SOPI: Number of the command cycles + number of the address cycles = 2 + 4 = 6
Flash DOPI: Number of the command cycles + number of the address cycles = 1 + 2 = 3
RAM DOPI: number of command cycles + number of address cycles = 1 + 2 - 1 = 2 (start dummy (latency) calculation from the second cycle)
Note 2. Number of stable OM_DQS cycles: Maximum value is 2 cycles of OM_SCLK (Reference specification of OctaFlash / OctaRAM)
Note 3. Number of the OM_DQS delay cycles: OM_DQS clock delay due to external causes
When OM_DQS delay ≤ 0.25 OM_SCLK: OM_DQS delay cycles = -1
When 0.25 OM_SCLK < OM_DQS delay ≤ 1.25 OM_SCLK: OM_DQS delay cycles = 0
When 1.25 OM_SCLK < OM_DQS delay ≤ 2.25 OM_SCLK: OM_DQS delay cycles = 1
In the flash SOPI mode, read 0.25 as 0.5, 1.25 as 1.5, and 2.25 as 2.5 in the above expressions.
Note 4. Pre-cycle setting: OM_DQS pre-cycle on/off
OM_DQS pre-cycle off: Pre-cycle setting = 0
OM_DQS pre-cycle on: Pre-cycle setting = -1

Setting example: Flash DOPI mode, dummy cycles (latency) = 4, pre-cycle is off
Minimum value of OM_DQS enable counter (MDTR) ≥ 1 + 2 + 2 + 0 = 5
Maximum value of OM_DQS enable counter (MDTR) ≤ 1 + 2 + 4 + 0 = 7

From the above, setting MDTR .DQSEDOPI[3:0] from 5 to 7 can latch the correct data. If MDTR is too small, wrong data may be latched due to a transition from Hi-Z to Low. If MDTR is too large, no data is latched.

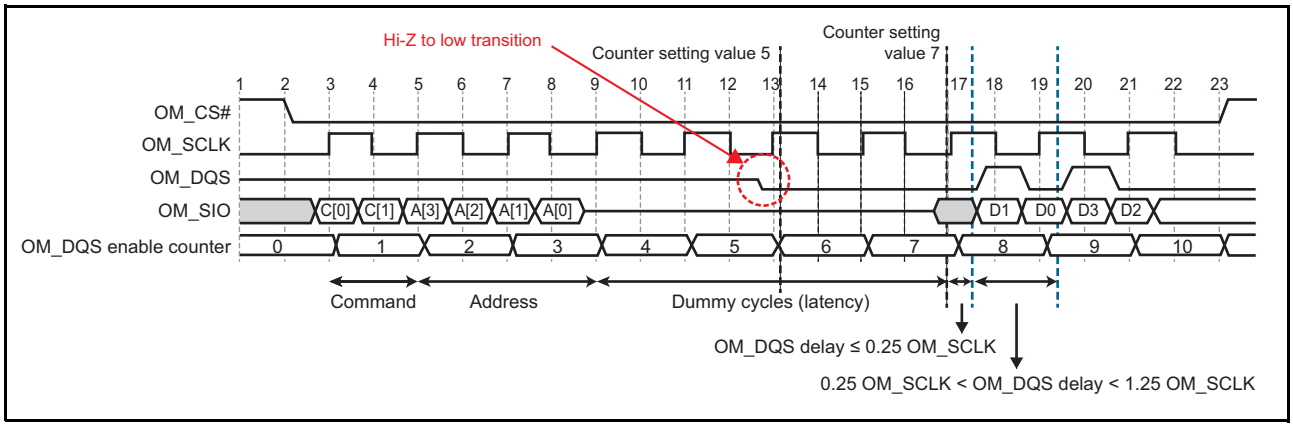


Figure 22.17 Timing of the OM_DQS Enable Counter in DOPI Mode for the Flash Memory with 4 as the Number of Dummy Cycles (Latency), and OM_DQS Pre-Cycle Off

23. I²C Bus Interface

This section gives an overall description of the I²C bus interface (RIIC).
The first section describes the features specific to this LSI, including the number of units and the register base addresses.
The subsequent sections describe the RIIC's functions and registers.

23.1 Features

23.1.1 Channels

This LSI has the following number of channels of the I²C bus interface (RIIC).

Table 23.1 Channels of RIIC

Item	Description
Number of channels	4
Name	RIICn (n = 0 to 3)

Table 23.2 Index

Index	Description
n	Throughout this section, the individual channels of the I²C bus interface are identified by the index "n" (n = 0 to 3); for example, RIICnCR1 for the I²C bus control register 1.

23.1.2 Register Base Addresses

The base address <RIICn_base> of each RIICn is listed in the following table.
All RIICn register addresses are given as values obtained by adding offsets to the register base address <RIICn_base> for each channel.

Table 23.3 Register Base Address

Channel	Base Address Name	Base Address
RIIC0	<RIIC0_base>	$E803\_A000_H$
RIIC1	<RIIC1_base>	$E803\_A400_H$
RIIC2	<RIIC2_base>	$E803\_A800_H$
RIIC3	<RIIC3_base>	$E803\_AC00_H$

23.1.3 External I/O Signals

The following table shows the external I/O signals of the RIIC.

Table 23.4 RIICn Pin Configuration

Channel	Alternative Port Pin Name	Function
RIIC0	RIIC0SCL	RIIC0 serial clock I/O pin
	RIIC0SDA	RIIC0 serial data I/O pin
RIIC1	RIIC1SCL	RIIC1 serial clock I/O pin
	RIIC1SDA	RIIC1 serial data I/O pin
RIIC2	RIIC2SCL	RIIC2 serial clock I/O pin
	RIIC2SDA	RIIC2 serial data I/O pin
RIIC3	RIIC3SCL	RIIC3 serial clock I/O pin
	RIIC3SDA	RIIC3 serial data I/O pin

23.2 Overview

23.2.1 Functional Overview

Communications format

- I²C bus format or SMBus format
- Master mode or slave mode selectable
- Automatic securing of the various set-up times, hold times, and bus-free times for the transfer rate

Transfer rate

Up to 1 Mbps

SCL clock

For master operation, the duty cycle of the SCL clock is selectable in the following range.
0% < Duty cycle < 100%.

Issuing and detecting conditions

- Start, restart, and stop conditions are automatically generated.
- Start conditions (including restart conditions) and stop conditions are detected.

Slave address

- Up to three slave-address settings can be made.
- Seven- and ten-bit address formats are supported (along with the use of both at once).
- General call addresses, device ID addresses, and SMBus host addresses are detected.

Acknowledgement

- For transmission, the acknowledge bit is automatically loaded
  - Transfer of the next data for transmission can be automatically suspended on detection of a not-acknowledge bit.
- For reception, the acknowledge bit is automatically transmitted
  - If a wait between the eighth and ninth clock cycles has been selected, software control of the value in the acknowledge field in response to the received value is possible.

Wait function

- In reception, the following periods of waiting can be obtained by holding the clock signal (SCL) at the low level:
  - Waiting between the eighth and ninth clock cycles
  - Waiting between the ninth clock cycle and the first clock cycle of the next transfer (WAIT function)

SDA output delay function

Timing of the output of transmitted data, including the acknowledge bit, can be delayed.

Arbitration

- For multi-master operation
  - Operation to synchronize the SCL (clock) signal in cases of conflict with the SCL signal from another master is possible.
  - When issuing the start condition would create conflict on the bus, loss of arbitration is detected by testing for non-matching between the internal signal for the SDA line and the level on the SDA line.
  - In master operation, loss of arbitration is detected by testing for non-matching between the signal on the SDA line and the internal signal for the SDA line.
- Loss of arbitration due to detection of the start condition while the bus is busy is detectable (to prevent the issuing of double start conditions).
- Loss of arbitration in transfer of a not-acknowledge bit due to the internal signal for the SDA line and the level on the SDA line not matching is detectable.
- Loss of arbitration due to non-matching of internal and line levels for data is detectable in slave transmission.

Timeout function

The internal time-out function is capable of detecting long-interval stop of the SCL (clock signal).

Noise removal

The interface incorporates analog noise filters and digital noise filters for input on the RIICnSCL and RIICnSDA pins, and the width for noise cancellation by the digital noise filters is adjustable by software.

Interrupt sources

- Eight sources:
  - Transmission complete
  - Receive-data-full
  - Transmit-data-empty
  - Detection of a stop condition
  - Detection of a start condition
  - Reception of a NACK
  - Arbitration lost
  - Timeout

Low power consumption function

Module-stop state can be set.

23.2.2 Block Diagram

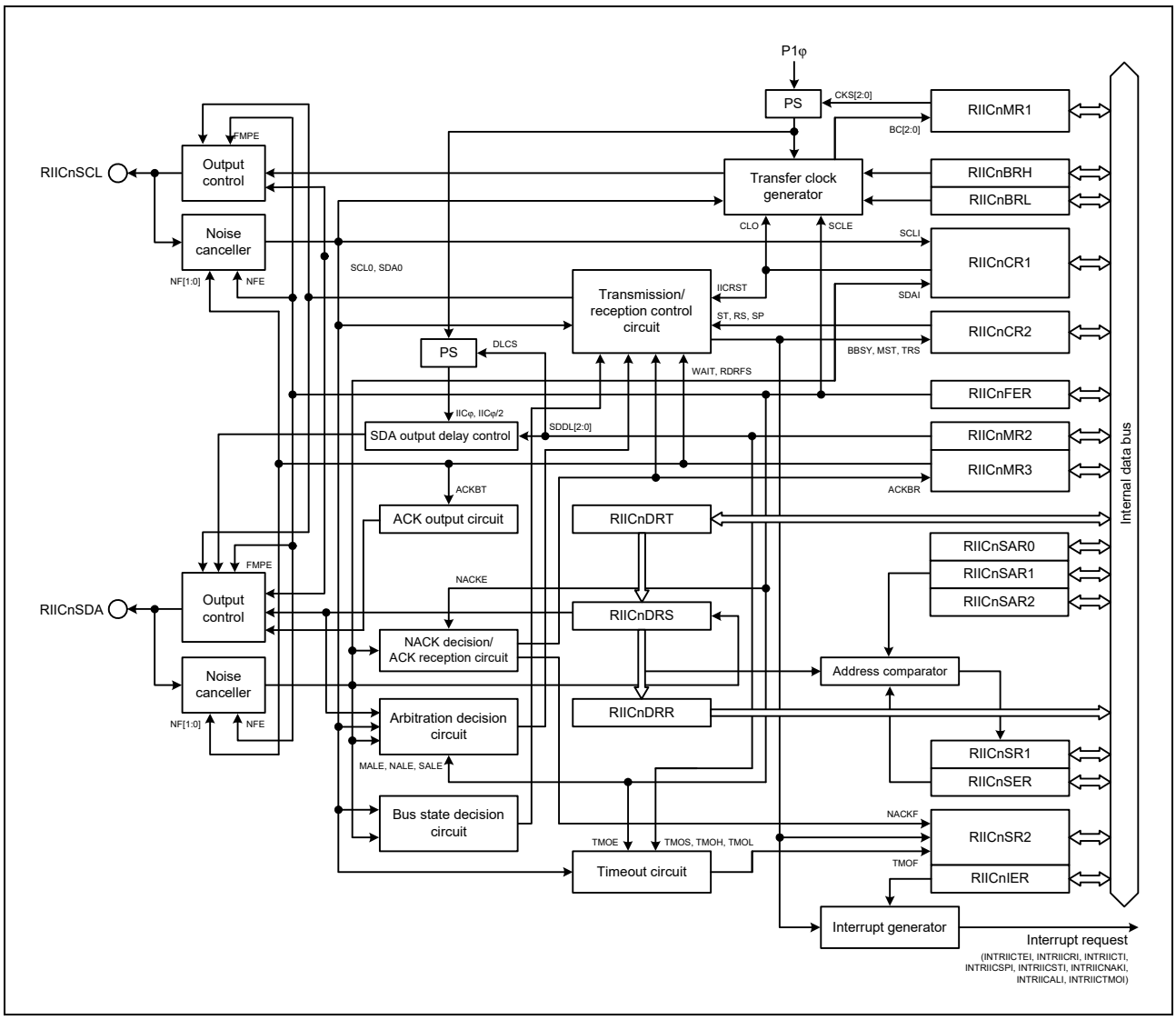


Figure 23.1 Block Diagram of RIIC

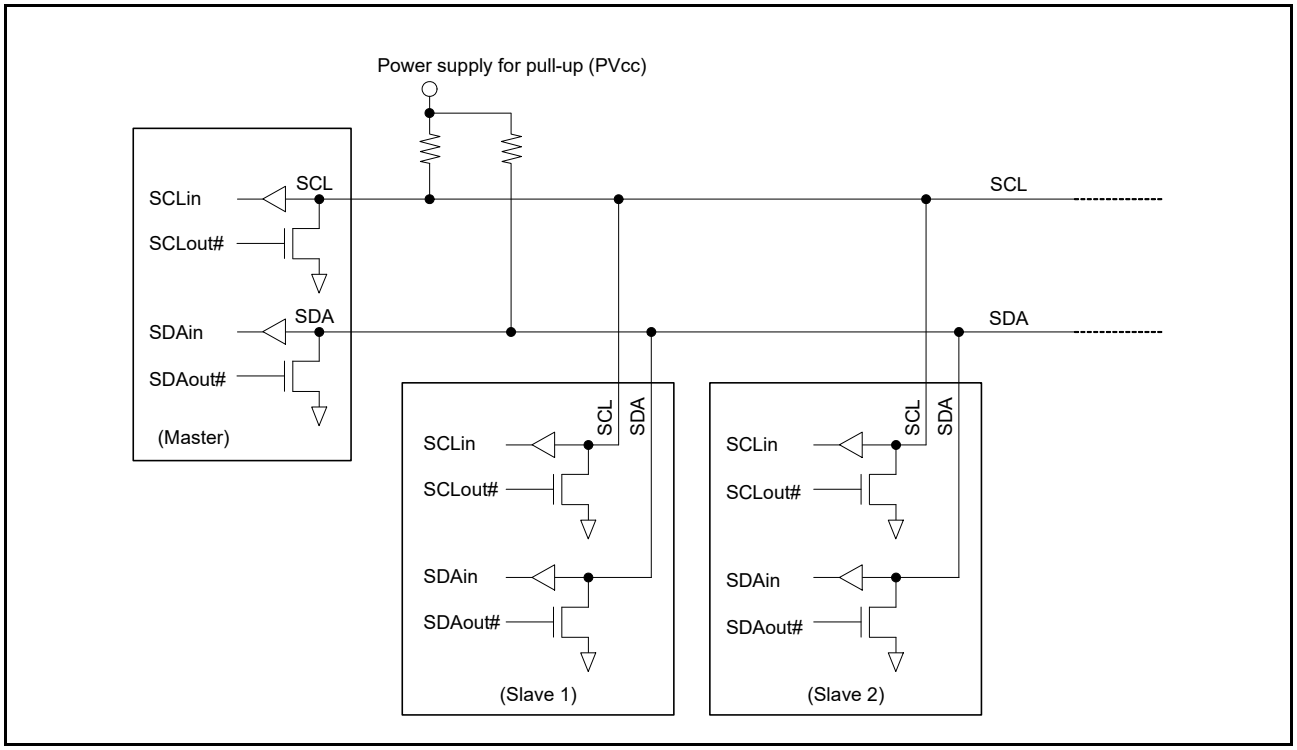


Figure 23.2 Connections to the External Circuit by the I/O Pins (I²C Bus Configuration Example)

RIICnSCL and RIICnSDA are Schmitt input/open-drain output pins for both master and slave operations. Because the output is open drain, an external pull-up resistor is required.

23.3 Registers

Channel	Register Name	Abbreviation	Address	Access Size
RIIC0	I²C bus control register 1	ICCR1	E803 A000h	8, 16, 32
	I²C bus control register 2	ICCR2	E803 A004h	8, 16, 32
	I²C bus mode register 1	ICMR1	E803 A008h	8, 16, 32
	I²C bus mode register 2	ICMR2	E803 A00Ch	8, 16, 32
	I²C bus mode register 3	ICMR3	E803 A010h	8, 16, 32
	I²C bus function enable register	ICFER	E803 A014h	8, 16, 32
	I²C bus status enable register	ICSER	E803 A018h	8, 16, 32
	I²C bus interrupt enable register	ICIER	E803 A01Ch	8, 16, 32
	I²C bus status register 1	ICSR1	E803 A020h	8, 16, 32
	I²C bus status register 2	ICSR2	E803 A024h	8, 16, 32
	I²C slave address register 0	ICSAR0	E803 A028h	8, 16, 32
	I²C slave address register 1	ICSAR1	E803 A02Ch	8, 16, 32
	I²C slave address register 2	ICSAR2	E803 A030h	8, 16, 32
	I²C bus bit rate low-level register	ICBRL	E803 A034h	8, 16, 32
	I²C bus bit rate high-level register	ICBRH	E803 A038h	8, 16, 32
	I²C bus transmit data register	ICDRT	E803 A03Ch	8, 16, 32
	I²C bus receive data register	ICDRR	E803 A040h	8, 16, 32
RIIC1	I²C bus control register 1	ICCR1	E803 A400h	8, 16, 32
	I²C bus control register 2	ICCR2	E803 A404h	8, 16, 32
	I²C bus mode register 1	ICMR1	E803 A408h	8, 16, 32
	I²C bus mode register 2	ICMR2	E803 A40Ch	8, 16, 32
	I²C bus mode register 3	ICMR3	E803 A410h	8, 16, 32
	I²C bus function enable register	ICFER	E803 A414h	8, 16, 32
	I²C bus status enable register	ICSER	E803 A418h	8, 16, 32
	I²C bus interrupt enable register	ICIER	E803 A41Ch	8, 16, 32
	I²C bus status register 1	ICSR1	E803 A420h	8, 16, 32
	I²C bus status register 2	ICSR2	E803 A424h	8, 16, 32
	I²C slave address register 0	ICSAR0	E803 A428h	8, 16, 32
	I²C slave address register 1	ICSAR1	E803 A42Ch	8, 16, 32
	I²C slave address register 2	ICSAR2	E803 A430h	8, 16, 32
	I²C bus bit rate low-level register	ICBRL	E803 A434h	8, 16, 32
	I²C bus bit rate high-level register	ICBRH	E803 A438h	8, 16, 32
	I²C bus transmit data register	ICDRT	E803 A43Ch	8, 16, 32
	I²C bus receive data register	ICDRR	E803 A440h	8, 16, 32
RIIC2	I²C bus control register 1	ICCR1	E803 A800h	8, 16, 32
	I²C bus control register 2	ICCR2	E803 A804h	8, 16, 32
	I²C bus mode register 1	ICMR1	E803 A808h	8, 16, 32
	I²C bus mode register 2	ICMR2	E803 A80Ch	8, 16, 32
	I²C bus mode register 3	ICMR3	E803 A810h	8, 16, 32
	I²C bus function enable register	ICFER	E803 A814h	8, 16, 32
	I²C bus status enable register	ICSER	E803 A818h	8, 16, 32
	I²C bus interrupt enable register	ICIER	E803 A81Ch	8, 16, 32
	I²C bus status register 1	ICSR1	E803 A820h	8, 16, 32
	I²C bus status register 2	ICSR2	E803 A824h	8, 16, 32

Channel	Register Name	Abbreviation	Address	Access Size
RIIC2	I²C slave address register 0	ICSAR0	E803 A828h	8, 16, 32
	I²C slave address register 1	ICSAR1	E803 A82Ch	8, 16, 32
	I²C slave address register 2	ICSAR2	E803 A830h	8, 16, 32
	I²C bus bit rate low-level register	ICBRL	E803 A834h	8, 16, 32
	I²C bus bit rate high-level register	ICBRH	E803 A838h	8, 16, 32
	I²C bus transmit data register	ICDRT	E803 A83Ch	8, 16, 32
	I²C bus receive data register	ICDRR	E803 A840h	8, 16, 32
RIIC3	I²C bus control register 1	ICCR1	E803 AC00h	8, 16, 32
	I²C bus control register 2	ICCR2	E803 AC04h	8, 16, 32
	I²C bus mode register 1	ICMR1	E803 AC08h	8, 16, 32
	I²C bus mode register 2	ICMR2	E803 AC0Ch	8, 16, 32
	I²C bus mode register 3	ICMR3	E803 AC10h	8, 16, 32
	I²C bus function enable register	ICFER	E803 AC14h	8, 16, 32
	I²C bus status enable register	ICSER	E803 AC18h	8, 16, 32
	I²C bus interrupt enable register	ICIER	E803 AC1Ch	8, 16, 32
	I²C bus status register 1	ICSR1	E803 AC20h	8, 16, 32
	I²C bus status register 2	ICSR2	E803 AC24h	8, 16, 32
	I²C slave address register 0	ICSAR0	E803 AC28h	8, 16, 32
	I²C slave address register 1	ICSAR1	E803 AC2Ch	8, 16, 32
	I²C slave address register 2	ICSAR2	E803 AC30h	8, 16, 32
	I²C bus bit rate low-level register	ICBRL	E803 AC34h	8, 16, 32
	I²C bus bit rate high-level register	ICBRH	E803 AC38h	8, 16, 32
	I²C bus transmit data register	ICDRT	E803 AC3Ch	8, 16, 32
	I²C bus receive data register	ICDRR	E803 AC40h	8, 16, 32

23.3.1 RIICnCR1 — I²C Bus Control Register 1

Access: RIICnCR1 is a 32-bit readable/writable register.
RIICnCR1L and RIICnCR1H are 16-bit readable/writable registers.
RIICnCR1LL, RIICnCR1LH, RIICnCR1HL, and RIICnCR1HH are 8-bit readable/writable registers.

Address: RIICnCR1: <RIICn_base> + 0000_H
RIICnCR1L: <RIICn_base> + 0000_H, RIICnCR1H: <RIICn_base> + 0002_H
RIICnCR1LL: <RIICn_base> + 0000_H, RIICnCR1LH: <RIICn_base> + 0001_H, RIICnCR1HL: <RIICn_base> + 0002_H,
RIICnCR1HH: <RIICn_base> + 0003_H

Initial Value: 0000 $001F_H$. This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI
Initial value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Table 23.5 RIICnCR1 register contents (1/2)

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are read as 0. The write value should be 0.
7	ICE	I²C Bus Interface Enable 0: Output to the RIICnSCL and RIICnSDA pins is disabled. (Input to the RIICnSCL and RIICnSDA pins is enabled.) 1: Enabled (RIICnSCL and RIICnSDA pins are in the driving state) (An RIIC reset or an internal reset is selected according to the combination of this bit and IICRST bit settings.)
6	IICRST	I²C Bus Interface Internal Reset 0: Clears the RIIC reset or internal reset. 1: Initiates the RIIC reset or internal reset. (Clears the bit counter and the SCL/SDA output latch)
5	CLO	Extra SCL Clock Cycle Output 0: Does not output an extra SCL clock cycle (default). 1: Outputs an extra SCL clock cycle. (The CLO bit is cleared automatically after one clock cycle is output.)
4	SOWP*2	SCLO/SDAO Write Protect 0: Allows the SCLO and SDAO bits to be rewritten. (This bit is read as 1.)
3	SCLO*1,*2	SDA Output Control • Read: 0: RIICnSCL pin output is at a low level. 1: RIICnSCL pin is in a high-impedance state. • Write: 0: Changes the RIICnSCL pin output to a low level. 1: Changes the RIICnSCL pin in a high-impedance state. (High level output is achieved through an external pull-up resistor.)

Table 23.5 RIICnCR1 register contents (2/2)

Bit Position	Bit Name	Function
2	SDAO*1,*2	SDA Output Control • Read: 0: RIICnSDA pin output is at a low level. 1: RIICnSDA pin is in a high-impedance state. • Write: 0: Changes the RIICnSDA pin output to a low level. 1: Changes the RIICnSDA pin in a high-impedance state. (High level output is achieved through an external pull-up resistor.)
1	SCLI	SCL Bus Input Monitor 0: RIICnSCL pin input is at a low level. 1: RIICnSCL pin input is at a high level.
0	SDAI	SDA Bus Input Monitor 0: RIICnSDA pin input is at a low level. 1: RIICnSDA pin input is at a high level.

Note 1. Do not write to these bits during communication. Changing a value during communication may cause a transmission or reception failure or an AL error.
Note 2. To change the SDAO and SCLO bits, set the SOWP bit to 0 at the same timing to set the SDAO and SCLO bits to 0.

CLO Bit (Extra SCL Clock Cycle Output)

This bit is used to output an extra SCL clock cycle for debugging or error processing.
Normally, set the bit to 0. Setting the bit to 1 in a normal communication state causes a communication error.
For details on this function, see section 23.13.2, Extra SCL Clock Cycle Output Function.

IICRST Bit (I²C Bus Interface Internal Reset)

This bit is used to reset the internal states of the RIIC.
Setting this bit to 1 initiates an RIIC reset or internal reset.
Whether an RIIC reset or internal reset is initiated is determined according to the combination with the ICE bit. Table 23.6 lists the resets of the RIIC.
The RIIC reset resets all registers including the RIICnCR2.BBSY flag (except ICE and IICRST) and internal states of the RIIC, and the internal reset resets the bit counter (RIICnMR1.BC[2:0] bits), the I²C bus shift register (RIICnDRS), and the I²C bus status registers (RIICnSR1 and RIICnSR2) as well as the internal states of the RIIC. For the reset conditions for each register, see section 23.15, Reset Function of RIIC.
An internal reset initiated with the IICRST bit set to 1 during operation (with the ICE bit set to 1) resets the internal states of the RIIC without initializing the port settings and the control and setting registers of the RIIC when the bus or RIIC hangs up due to a communication error.
If the RIIC hangs up in a low level output state, resetting the internal states cancels the low level output state and releases the bus with the RIICnSCL pin and RIICnSDA pin at a high impedance.

Caution: If an internal reset is initiated using the IICRST bit for a bus hang-up occurred during communication with the master device in slave mode, the states may become different between the slave device and the master device (due to the difference in the bit counter information). For this reason, do not initiate an internal reset in slave mode, but initiate restoration processing from the master device. If an internal reset is necessary because the RIIC hangs up with the SCL line in a low level output state in slave mode, initiate an internal reset and then issue a restart condition from the master device or resume communication from the start condition issuance after issuing a stop condition. If communication is restarted by initiating a reset solely in the slave device without issuing a start condition or restart condition from the master device, synchronization will be lost because the master and slave devices operate asynchronously.

Table 23.6 RIIC Resets

IICRST	ICE	State	Specifications
1	0	RIIC reset	Resets all registers (except ICE and IICRST) and internal states of the RIIC.
	1	Internal reset	Reset the RIICnMR1.BC[2:0] bits, and the RIICnSR1, RIICnSR2, RIICnDRS registers and the internal states of the RIIC.

ICE Bit (I2C Bus Interface Enable)

The ICE bit selects driving or non-driving of the RIICnSCL and RIICnSDA pins. Moreover, this bit can perform two types of reset in combination with the IICRST bit. For the types of reset, see Table 23.6.

Set the ICE bit to 1 when using RIIC. Setting the ICE bit to 1 selects driving of the RIICnSCL and RIICnSDA pins.

Set the ICE bit to 0 when RIIC is not to be used. Clearing the ICE bit to 0 disables output from the RIICnSCL and RIICnSDA pins.

23.3.2 RIICnCR2 — I²C Bus Control Register 2

Access: RIICnCR2 is a 32-bit readable/writable register.
RIICnCR2L and RIICnCR2H are 16-bit readable/writable registers.
RIICnCR2LL, RIICnCR2LH, RIICnCR2HL, and RIICnCR2HH are 8-bit readable/writable registers.

Address: RIICnCR2: <RIICn_base> + 0004_H
RIICnCR2L: <RIICn_base> + 0004_H, RIICnCR2H: <RIICn_base> + 0006_H
RIICnCR2LL: <RIICn_base> + 0004_H, RIICnCR2LH: <RIICn_base> + 0005_H, RIICnCR2HL: <RIICn_base> + 0006_H,
RIICnCR2HH: <RIICn_base> + 0007_H

Initial Value: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	BBSY	MST	TRS	—	SP	RS	ST	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R

Table 23.7 RIICnCR2 register contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are read as 0. The write value should be 0.
7	BBSY	Bus Busy Detection Flag 0: The I²C bus is released (bus free state). 1: The I²C bus is occupied (bus busy state or in the bus free state).
6	MST	Master/Slave Mode 0: Slave mode 1: Transmit mode
5	TRS	Transmit/Receive Mode 0: Receive mode 1: Transmit mode
4	—	Reserved These bits are read as 0. The write value should be 0.
3	SP	Stop Condition Issuance Request 0: Does not request to issue a stop condition. 1: Requests to issue a stop condition.
2	RS	Restart Condition Issuance Request 0: Does not request to issue a restart condition. 1: Requests to issue a restart condition.
1	ST	Start Condition Issuance Request 0: Does not request to issue a start condition. 1: Requests to issue a start condition.
0	—	Reserved These bits are read as 0. The write value should be 0.

ST Bit (Start Condition Issuance Request)

This bit is used to request transition to master mode and issuance of a start condition.
When this bit is set to 1 to request to issue a start condition, a start condition is issued when the BBSY flag is set to 0 (bus free).
For details on the start condition issuance, see section 23.12, Start Condition/Restart Condition/Stop Condition Issuing Function.
[Setting condition]
When 1 is written to the ST bit
[Clearing conditions]

- When 0 is written to the ST bit
- When a start condition has been issued
- When the RIICnSR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

Caution: Set the ST bit to 1 (start condition issuance request) when the BBSY flag is set to 0 (bus free).
Note that arbitration may be lost if the ST bit is set to 1 (start condition issuance request) when the BBSY flag is set to 1 (bus busy).

RS Bit (Restart Condition Issuance Request)

This bit is used to request that a restart condition be issued in master mode.
When this bit is set to 1 to request to issue a restart condition, a restart condition is issued when the BBSY flag is set to 1 (bus busy) and the MST bit is set to 1 (master mode).
For details on the restart condition issuance, see section 23.12, Start Condition/Restart Condition/Stop Condition Issuing Function.
[Setting condition]
When 1 is written to the RS bit with the RIICnCR2.BBSY flag set to 1
[Clearing conditions]

- When 0 is written to the RS bit
- When a restart condition has been issued or a start condition is detected
- When a stop condition is detected
- When the RIICnCR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

Caution:1. Do not set the RS bit to 1 while issuing a stop condition.
Caution:2. It is commended to issue a restart condition in master transmit mode. If the RS bit is set to 1 (restart condition issuance request) in mode other than master mode, the restart condition is not issued in this mode but the restart condition issuance request bit remains set. If the operating mode changes to master mode with the bit not being cleared, the restart condition may be issued.

SP Bit (Stop Condition Issuance Request)

This bit is used to request that a stop condition be issued in master mode.
When this bit is set to 1 to request to issue a stop condition, a stop condition is issued when the BBSY flag is set to 1 (bus busy) and the MST bit is set to 1 (master mode).
For details on the stop condition issuance, see section 23.12, Start Condition/Restart Condition/Stop Condition Issuing Function.
[Setting condition]
When 1 is written to the SP bit with both the RIICnCR2.BBSY flag and the RIICnCR2.MST bit set to 1
[Clearing conditions]

- When 0 is written to the SP bit
- When a stop condition has been issued or a stop condition is detected
- When the RIICnSR2.AL (arbitration-lost) flag is set to 1
- When a start condition and a restart condition are detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

Caution:1. Wring to the SP bit is not possible while the setting of the BBSY flag is 0 (bus free).
Caution:2. Do not set the SP bit to 1 while a restart condition is being issued.

TRS Bit (Transmit/Receive Mode)

This bit indicates transmit or receive mode.
The RIIC is in receive mode when the TRS bit is set to 0 and is in transmit mode when the bit is set to 1. Combination of this bit and the MST bit indicates the operating mode of the RIIC.
The value of the TRS bit is automatically changed to the value for transmission mode or reception mode by detection or issuing of a start condition, setting or clearing of the R/W# bit, etc.
[Setting conditions]

- When a start condition is issued normally according to the start condition issuance request (when a start condition is detected with the ST bit set to 1)
- When the R/W# bit added to the slave address is set to 0 in master mode
- When the address received in slave mode matches the address enabled in RIICnSER, with the R/W# bit set to 1

[Clearing conditions]

- When a stop condition is detected
- The RIICnSR2.AL (arbitration-lost) flag being set to 1
- In master mode, reception of a slave address to which an R/W# bit with the value 1 is appended
- In slave mode, a match between the received address and the address enabled in RIICnSER when the value of the received R/W# bit is 0 (including cases where the received address is the general call address)
- In slave transmit mode, a restart condition is detected (a restart condition is detected with RIICnCR2.BBSY = 1 and RIICnCR2.MST = 0)
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

MST Bit (Master/Slave Mode)

This bit indicates master or slave mode.
The RIIC is in slave mode when the MST bit is set to 0 and is in master mode when the bit is set to 1. Combination of this bit and the TRS bit indicates the operating mode of the RIIC.
The value of the MST bit is automatically changed to the value for master mode or slave mode by detection or issuing of a start condition, etc.
[Setting conditions]

- When a start condition is issued normally according to the start condition issuance request (when a start condition is detected with the ST bit set to 1)

[Clearing conditions]

- When a stop condition is detected
- When the RIICnSR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

BBSY Flag (Bus Busy Detection)

The BBSY flag indicates whether the I²C bus is occupied (bus busy) or released (bus free).
This bit is set to 1 when the SDA line changes from high to low under the condition of SCL = high, assuming that a start condition has been issued.
When the SDA line changes from low to high under the condition of SCL = high, this bit is cleared to 0 after the bus free time (specified in RIICnBRL) start condition is not detected, assuming that a stop condition has been issued.
[Setting condition]

- When a start condition is detected

[Clearing conditions]

- When the bus free time (specified in RIICnBRL) start condition is not detected after detecting a stop condition
- When 1 is written to the RIICnCR1.IICRST bit with the RIICnCR1.ICE bit set to 0 (RIIC reset)

23.3.3 RIICnMR1 — I²C Bus Mode Register 1

Access: RIICnMR1 is a 32-bit readable/writable register.
RIICnMR1L and RIICnMR1H are 16-bit readable/writable registers.
RIICnMR1LL, RIICnMR1LH, RIICnMR1HL, and RIICnMR1HH are 8-bit readable/writable registers.

Address: RIICnMR1: <RIICn_base> + 0008_H
RIICnMR1L: <RIICn_base> + 0008_H, RIICnMR1H: <RIICn_base> + $000A_H$
RIICnMR1LL: <RIICn_base> + 0008_H, RIICnMR1LH: <RIICn_base> + 0009_H, RIICnMR1HL: <RIICn_base> + $000A_H$,
RIICnMR1HH: <RIICn_base> + $000B_H$

Initial Value: 0000 0008_H This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CKS[2:0]			BCWP	BC[2:0]		
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.8 RIICnMR1 register contents

Bit Position	Bit Name	Function
31 to 7	—	Reserved These bits are read as 0. The write value should be 0.
6 to 4	CKS[2:0]	Internal Reference Clock (IICφ) Selection b6 b4 0 0 0: IICφ = P1φ/1 0 0 1: IICφ = P1φ/2 0 1 0: IICφ = P1φ/4 0 1 1: IICφ = P1φ/8 1 0 0: IICφ = P1φ/16 1 0 1: IICφ = P1φ/32 1 1 0: IICφ = P1φ/64 1 1 1: IICφ = P1φ/128
3	BCWP*1	BC Write Protect 0: Enables a value to be written in the BC[2:0] bits. (This bit is read as 1.)
2 to 0	BC[2:0]	Bit Counter b2 b0 0 0 0: 9 bits 0 0 1: 2 bits 0 1 0: 3 bits 0 1 1: 4 bits 1 0 0: 5 bits 1 0 1: 6 bits 1 1 0: 7 bits 1 1 1: 8 bits

Note 1. When rewriting the BC[2:0] bits, write 0 to the BCWP bit simultaneously.

BC[2:0] Bits (Bit Counter)

These bits function as a counter that indicates the number of bits remaining to be transferred at the detection of a rising edge on the SCL line. Although these bits are writable and readable, it is not necessary to access these bits under normal conditions.

To write to these bits, specify the number of bits to be transferred plus one (data is transferred with an additional acknowledge bit) between transferred frames when the SCL line is at a low level.

The values of the BC[2:0] bits return to 000_B at the end of a data transfer including the acknowledge bit or when a start condition including a restart condition is detected.

23.3.4 RIICnMR2 — I²C Bus Mode Register 2

Access: RIICnMR2 is a 32-bit readable/writable register.
RIICnMR2L and RIICnMR2H are 16-bit readable/writable registers.
RIICnMR2LL, RIICnMR2LH, RIICnMR2HL, and RIICnMR2HH are 8-bit readable/writable registers.

Address: RIICnMR2: <RIICn_base> + $000C_H$
RIICnMR2L: <RIICn_base> + $000C_H$, RIICnMR2H: <RIICn_base> + $000E_H$
RIICnMR2LL: <RIICn_base> + $000C_H$, RIICnMR2LH: <RIICn_base> + $000D_H$, RIICnMR2HL: <RIICn_base> + $000E_H$,
RIICnMR2HH: <RIICn_base> + $000F_H$

Initial Value: $0000\ 0006_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DLCS	SDDL[2:0]			—	TMOH	TMOL	TMOS
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 23.9 RIICnMR2 register contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are read as 0. The write value should be 0.
7	DLCS	SDA Output Delay Clock Source Selection 0: The internal reference clock (IICφ) is selected as the clock source of the SDA output delay counter. 1: The internal reference clock divided by 2 (IICφ/2) is selected as the clock source of the SDA output delay counter.*1
6 to 4	SDDL[2:0]	SDA Output Delay Counter • When RIICnMR2.DLCS = 0 (IICφ) b6 b4 0 0 0: No output delay 0 0 1: 1 IICφ cycle 0 1 0: 2 IICφ cycles 0 1 1: 3 IICφ cycles 1 0 0: 4 IICφ cycles 1 0 1: 5 IICφ cycles 1 1 0: 6 IICφ cycles 1 1 1: 7 IICφ cycles • When RIICnMR2.DLCS = 1 (IICφ/2) b6 b4 0 0 0: No output delay 0 0 1: 1 or 2 IICφ cycles 0 1 0: 3 or 4 IICφ cycles 0 1 1: 5 or 6 IICφ cycles 1 0 0: 7 or 8 IICφ cycles 1 0 1: 9 or 10 IICφ cycles 1 1 0: 11 or 12 IICφ cycles 1 1 1: 13 or 14 IICφ cycles
3	—	Reserved These bits are read as 0. The write value should be 0.
2	TMOH	Timeout H Count Control 0: Count is disabled while the SCL line is at a high level. 1: Count is enabled while the SCL line is at a high level.

Table 23.9 RIICnMR2 register contents

Bit Position	Bit Name	Function
1	TMOL	Timeout L Count Control 0: Count is disabled while the SCL line is at a low level. 1: Count is enabled while the SCL line is at a low level.
0	TMOS	Timeout Detection Time Selection 0: Long mode is selected. 1: Short mode is selected.

Note 1. The setting DLCS = 1 (IICφ/2) only becomes valid when SCL is at the low level. When SCL is at the high level, the setting DLCS = 1 becomes invalid and the clock source becomes the internal reference clock (IICφ).

TMOS Bit (Timeout Detection Time Selection)

This bit is used to select long mode or short mode for the timeout detection time when the timeout function is enabled (RIICnFER.TMOE bit = 1). When this bit is set to 0, long mode is selected. When this bit is set to 1, short mode is selected. In long mode, the timeout detection internal counter functions as a 16 bit-counter. In short mode, the counter functions as a 14 bit-counter. While the SCL line is in the state that enables this counter as specified by bits TMOH and TMOL, the counter counts up in synchronization with the internal reference clock (IICφ) as a count source.
For details on the timeout function, see section 23, Timeout Function.

TMOL Bit (Timeout L Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL line is held low when the timeout function is enabled (RIICnFER.TMOE bit = 1).

TMOH Bit (Timeout H Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL line is held high when the timeout function is enabled (RIICnFER.TMOE bit = 1).

SDDL[2:0] Bits (SDA Output Delay Setup Counter)

The SDA output can be delayed by the SDDL[2:0] setting. This counter works with the clock source selected by the DLCS bit. The setting of this function can be used for all types of SDA output, including the transmission of the acknowledge bit.
For details on this function, see section 23.7, Facility for Delaying SDA Output.

Caution: Set the SDA output delay time to meet the I²C bus standard (within the data enable time/acknowledge enable time*1) or the SMBus standard (within the data hold time: 300 [ns] or more, and SCL-clock low-level period - the data setup time: 250 [ns]). Note that, if a value outside the standard is set, communication with communication devices may malfunction or it may seemingly become a start condition or stop condition depending on the bus state.

Note 1. Data enable time/acknowledge enable time
3,450 [ns] (0 to 100 [kbps]: standard mode (Sm))
900 [ns] (0 to 400 [kbps]: fast mode (Fm))
450 [ns] (0 to 1 [Mbps]: fast mode plus (Fm+))

23.3.5 RIICnMR3 — I²C Bus Mode Register 3

Access: RIICnMR3 is a 32-bit readable/writable register.
RIICnMR3L and RIICnMR3H are 16-bit readable/writable registers.
RIICnMR3LL, RIICnMR3LH, RIICnMR3HL, and RIICnMR3HH are 8-bit readable/writable registers.

Address: RIICnMR3: <RIICn_base> + 0010_H
RIICnMR3L: <RIICn_base> + 0010_H, RIICnMR3H: <RIICn_base> + 0012_H
RIICnMR3LL: <RIICn_base> + 0010_H, RIICnMR3LH: <RIICn_base> + 0011_H, RIICnMR3HL: <RIICn_base> + 0012_H,
RIICnMR3HH: <RIICn_base> + 0013_H

Initial Value: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SMBE	WAIT	RDRFS	ACKWP	ACKBT	ACKBR	NF[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	W	R/W	R	R/W	R/W

Table 23.10 RIICnMR3 register contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are read as 0. The write value should be 0.
7	SMBE	SMBus/I²C Bus Selection 0: I²C bus is selected. 1: SMBus is selected.
6	WAIT*2	WAIT 0: No WAIT (The period between ninth clock cycle and first clock cycle is not held low.) 1: WAIT (The period between ninth clock cycle and first clock cycle is held low.) Low-hold is released by reading RIICnDRR.
5	RDRFS*2	RDRF Flag Set Timing Selection 0: The RDRF flag is set at the rising edge of the ninth SCL clock cycle. (The SCL line is not held low at the falling edge of the eighth clock cycle.) 1: The RDRF flag is set at the rising edge of the eighth SCL clock cycle. (The SCL line is held low at the falling edge of the eighth clock cycle.) Low-hold is released by writing a value to the ACKBT bit.
4	ACKWP*1	ACKBT Write Protect 0: Modification of the ACKBT bit is disabled. 1: Modification of the ACKBT bit is enabled.
3	ACKBT*1	Transmit Acknowledge 0: A 0 is sent as the acknowledge bit (ACK transmission). 1: A 1 is sent as the acknowledge bit (NACK transmission).
2	ACKBR	Receive Acknowledge 0: A 0 is received as the acknowledge bit (ACK reception). 1: A 1 is received as the acknowledge bit (NACK reception).
1, 0	NF[1:0]	Noise Filter Stage Selection b1 b0 0 0: Noise of up to one IICφ cycle is filtered out (single-stage filter). 0 1: Noise of up to two IICφ cycles is filtered out (2-stage filter). 1 0: Noise of up to three IICφ cycles is filtered out (3-stage filter). 1 1: Noise of up to four IICφ cycles is filtered out (4-stage filter).

Note 1. If it is attempted to write 1 to both ACKWP and ACKBT bits, the ACKBT bit cannot be set to 1.
Note 1. The WAIT and RDRFS bits are valid only in receive mode (invalid in transmit mode).

NF[1:0] Bits (Noise Filter Stage Selection)

These bits are used to select the width of noise that can be removed from the signals input to RIICnSCL or RIICnSDA pin.

ACKBR Bit (Receive Acknowledge)

This bit is used to store the acknowledge bit information received from the receive device in transmit mode.
[Setting condition]
When 1 is received as the acknowledge bit with the RIICnCR2.TRS bit set to 1
[Clearing conditions]

- When 0 is received as the acknowledge bit with the RIICnCR2.TRS bit set to 1
- When 1 is written to the RIICnCR1.IICRST bit while the RIICnCR1.ICE bit is 0 (RIIC reset)

ACKBT Bit (Transmit Acknowledge)

This bit is used to set the bit to be sent at the acknowledge timing in receive mode.
[Setting condition]
When 1 is written to this bit with the ACKWP bit set to 1
[Clearing conditions]

- When 0 is written to this bit with the ACKWP bit set to 1
- When stop condition issuance is detected
- When 1 is written to the RIICnCR1.IICRST bit while the RIICnCR1.ICE bit is 0 (RIIC reset)

Caution: The ACKBT bit must be written to while the ACKWP bit is 1. If the ACKBT bit is written to with the ACKWP bit cleared to 0, writing to the ACKBT bit is disabled.

ACKWP Bit (ACKBT Write Protect)

This bit is used to control the modification of the ACKBT bit.

RDRFS Bit (RDRF Flag Set Timing Selection)

This bit is used to select the RDRF flag set timing in receive mode and also to select whether to hold the SCL line low at the falling edge of the eighth SCL clock cycle.
When the RDRFS bit is 0, the SCL line is not held low at the falling edge of the eighth SCL clock cycle, and the RDRF flag is set to 1 at the rising edge of the ninth SCL clock cycle.
When the RDRFS bit is 1, the RDRF flag is set to 1 at the rising edge of the eighth SCL clock cycle and the SCL line is held low at the falling edge of the eighth SCL clock cycle. The low-hold of the SCL line is released by writing a value to the ACKBT bit.
After data is received with this setting, the SCL line is automatically held low before the acknowledge bit is sent. This enables processing to send ACK (ACKBT = 0) or NACK (ACKBT = 1) according to receive data.

WAIT Bit (WAIT)

This bit is used to control whether to hold the period between the ninth SCL clock cycle and the first SCL clock cycle low until the receive data buffer (RIICnDRR) is completely read each time single-byte data is received in receive mode. When the WAIT bit is 0, the receive operation is continued without holding the period between the ninth and the first SCL clock cycle low. When both the RDRFS and WAIT bits are 0, continuous receive operation is enabled with the double buffer.

When the WAIT bit is 1, the SCL line is held low from the falling edge of the ninth clock cycle until the RIICnDRR value is read each time single-byte data is received. This enables receive operation in byte units.

Caution: When the value of the WAIT bit is to be read, be sure to read the RIICnDRR beforehand.

SMBE Bit (SMBus Select)

Setting this bit to 1 enables the RIICnSER.HOAE bit.

23.3.6 RIICnFER — I²C Bus Function Enable Register

Access: RIICnFER is a 32-bit readable/writable register.
RIICnFERL and RIICnFERH are 16-bit readable/writable registers.
RIICnFERLL, RIICnFERLH, RIICnFERHL, and RIICnFERHH are 8-bit readable/writable registers.

Address: RIICnFER: <RIICn_base> + 0014_H
RIICnFERL: <RIICn_base> + 0014_H, RIICnFERH: <RIICn_base> + 0016_H
RIICnFERLL: <RIICn_base> + 0014_H, RIICnFERLH: <RIICn_base> + 0015_H, RIICnFERHL: <RIICn_base> + 0016_H,
RIICnFERHH: <RIICn_base> + 0017_H

Initial Value: $0000\ 0072_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	FMPE	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE
Initial value	0	0	0	0	0	0	0	0	0	1	1	1	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.11 RIICnFER register contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are read as 0. The write value should be 0.
7	FMPE	Fast-mode Plus Enable 0: No fm+ slope control circuit is used for the SCLn pin and SDAn pin 1: An fm+ slope control circuit is used for the SCLn pin and SDAn pin
6	SCLE	SCL Synchronous Circuit Enable 0: No SCL synchronous circuit is used. 1: An SCL synchronous circuit is used.
5	NFE	Digital Noise Filter Circuit Enable 0: No digital noise filter circuit is used. 1: A digital noise filter circuit is used.
4	NACKE	NACK Reception Transfer Suspension Enable 0: Transfer operation is not suspended during NACK reception (transfer suspension disabled). 1: Transfer operation is suspended during NACK reception (transfer suspension enabled).
3	SALE	Slave Arbitration-Lost Detection Enable 0: Slave arbitration-lost detection is disabled. 1: Slave arbitration-lost detection is enabled.
2	NALE	NACK Transmission Arbitration-Lost Detection Enable 0: NACK transmission arbitration-lost detection is disabled. 1: NACK transmission arbitration-lost detection is enabled.
1	MALE	Master Arbitration-Lost Detection Enable 0: Master arbitration-lost detection is disabled. (Disables the arbitration-lost detection function and does not clear the RIICnCR2.MST and TRS bits automatically when arbitration is lost.) 1: Master arbitration-lost detection is enabled. (Enables the arbitration-lost detection function and clears the RIICnCR2.MST and TRS bits automatically when arbitration is lost.)
0	TMOE	Timeout Function Enable 0: The timeout function is disabled. 1: The timeout function is enabled.

TMOE Bit (Timeout Function Enable)

This bit is used to enable or disable the timeout function.
For details on the timeout function, see section 23.13.1, Timeout Function.

MALE Bit (Master Arbitration-Lost Detection Enable)

This bit is used to specify whether to use the arbitration-lost detection function in master mode. Normally, set this bit to 1.

NALE Bit (NACK Transmission Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when ACK is detected during transmission of NACK in receive mode (such as when slaves with the same address exist on the bus or when two or more masters select the same slave device simultaneously with different number of receive bytes).

SALE Bit (Slave Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when a value different from the value being transmitted is detected on the bus in slave transmit mode (such as when slaves with the same address exist on the bus or when a mismatch with the transmit data occurs due to noise).

NACKE Bit (NACK Reception Transfer Suspension Enable)

This bit is used to specify whether to continue or discontinue the transfer operation when NACK is received in transmit mode. Normally, set this bit to 1.
When NACK is received with the NACKE bit set to 1, the next transfer operation is suspended.
When the NACKE bit is 0, the next transfer operation is continued regardless of the received acknowledge content.

SCLE Bit (SCL Synchronous Circuit Enable)

This bit is used to specify whether to synchronize the SCL clock with the SCL input clock. Normally, set this bit to 1.
When the SCLE bit is cleared to 0 (SCL synchronous circuit not used), the RIIC does not synchronize the SCL clock with the SCL input clock. In this setting, the RIIC outputs the SCL clock with the transfer rate set in RIICnBRH and RIICnBRL regardless of the SCL line state. For this reason, if the bus load of the I^{2}C bus line is much larger than the specification value or if the SCL clock output overlaps in multiple masters, the short-cycle SCL clock that does not meet the specification may be output. When no SCL synchronous circuit is used, it also affects the issuance of a start condition, restart condition, and stop condition, and the continuous output of extra SCL clock cycles.
This bit must not be cleared to 0 except for checking the output of the transfer rate.

FMPE Bit (Fast-mode Plus Enable)

This bit is used to specify whether to use a slope control circuit for Fast-mode Plus [fm+].
When this bit is set to 1, a slope control circuit conforming to the Fast-mode Plus [fm+] slope control standard (tof) of the I^{2}C bus is selected. When this bit is cleared to 0, a slope control circuit conforming to the Standard-mode [Sm] and Fast-mode [fm] slope control standard (tof) of the I^{2}C bus is selected.
Set this bit to 1 when using the transmission rate within a range up to 1 Mbps (Fast-mode Plus[fm+]) of the I^{2}C bus standard. Clear this bit to 0 when using the transmission rate at other rates (up to 100 kbps[Sm], up to 400 kbps[fm]) or for SMBus (10 to 100 kbps).

23.3.7 RIICnSER — I²C Bus Status Enable Register

Access: RIICnSER is a 32-bit readable/writable register.
RIICnSERL and RIICnSERH are 16-bit readable/writable registers.
RIICnSERLL, RIICnSERLH, RRIICnSERHL, and RIICnSERHH are 8-bit readable/writable registers.

Address: RIICnSER: <RIICn_base> + 0018_H
RIICnSERL: <RIICn_base> + 0018_H, RIICnSERH: <RIICn_base> + $001A_H$
RIICnSERLL: <RIICn_base> + 0018_H, RIICnSERLH: <RIICn_base> + 0019_H, RIICnSERHL: <RIICn_base> + $001A_H$,
RIICnSERHH: <RIICn_base> + $001B_H$

Initial Value: $0000\ 0009_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HOAE	—	DIDE	—	GCE	SAR2	SAR1	SAR0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R	R/W	R/W	R/W	R/W

Table 23.12 RIICnSER register contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are read as 0. The write value should be 0.
7	HOAE	Host Address Enable 0: Host address detection is disabled. 1: Host address detection is enabled.
6	—	Reserved This bit is read as 0. The write value should be 0.
5	DIDE	Device-ID Address Detection Enable 0: Device-ID address detection is disabled. 1: Device-ID address detection is enabled.
4	—	Reserved This bit is read as 0. The write value should be 0.
3	GCE	General Call Address Enable 0: General call address detection is disabled. 1: General call address detection is enabled.
2	SAR2	Slave Address Register 2 Enable 0: Slave address in RIICnSAR2 is disabled. 1: Slave address in RIICnSAR2 is enabled.
1	SAR1	Slave Address Register 1 Enable 0: Slave address in RIICnSAR1 is disabled. 1: Slave address in RIICnSAR1 is enabled.
0	SAR0	Slave Address Register 0 Enable 0: Slave address in RIICnSAR0 is disabled. 1: Slave address in RIICnSAR0 is enabled.

SARy Bit (Slave Address Register y Enable) (y = 0 to 2)

This bit is used to enable or disable the received slave address and the slave address set in RIICnSARy.
When this bit is set to 1, the slave address set in RIICnSARy is enabled and is compared with the received slave address.
When this bit is cleared to 0, the slave address set in RIICnSARy is disabled and is ignored even if it matches the received slave address.

GCE Bit (General Call Address Enable)

This bit is used to specify whether to ignore the general call address ($0000\ 000_B$ + 0 [W]: All 0) when it is received.
When this bit is set to 1, if the received slave address matches the general call address, the RIIC recognizes the received slave address as the general call address independently of the slave addresses set in RIICnSARy (y = 0 to 2) and performs data receive operation.
When this bit is cleared to 0, the received slave address is ignored even if it matches the general call address.

DIDE Bit (Device-ID Address Detection Enable)

This bit is used to specify whether to recognize and execute the Device-ID address when a device ID ($1111\ 100_B$) is received in the first frame after a start condition or restart condition is detected.
When this bit is set to 1, if the received first frame matches the device ID, the RIIC recognizes that the Device-ID address has been received. When the following R/W# bit is 0 [W], the RIIC recognizes the second and the following frames as slave addresses and continues the receive operation.
When this bit is cleared to 0, the RIIC ignores the received first frame even if it matches the device ID address and recognizes the first frame as a normal slave address.
For details on the device-ID address detection, see section 23.9.3, Device-ID Address Detection.

HOAE Bit (Host Address Enable)

This bit is used to specify whether to ignore received host address ($0001\ 000_B$) when the RIICnMR3.SMBS bit is 1.
When this bit is set to 1 while the RIICnMR3.SMBS bit is 1, if the received slave address matches the host address, the RIIC recognizes the received slave address as the host address independently of the slave addresses set in RIICnSARy (y = 0 to 2) and performs the receive operation.
When the RIICnMR3.SMBS bit or the HOAE bit is cleared to 0, the received slave address is ignored even if it matches the host address.

23.3.8 RIICnIER — I²C Bus Interrupt Enable Register

Access: RIICnIER is a 32-bit readable/writable register.
RIICnIERL and RIICnIERH are 16-bit readable/writable registers.
RIICnIERLL, RIICnIERLH, RIICnIERHL, and RIICnIERHH are 8-bit readable/writable registers.

Address: RIICnIER: <RIICn_base> + $001C_H$
RIICnIERL: <RIICn_base> + $001C_H$, RIICnIERH: <RIICn_base> + $001E_H$
RIICnIERLL: <RIICn_base> + $001C_H$, RIICnIERLH: <RIICn_base> + $001D_H$, RIICnIERHL: <RIICn_base> + $001E_H$,
RIICnIERHH: <RIICn_base> + $001F_H$

Initial Value: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.13 RIICnIER register contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are read as 0. The write value should be 0.
7	TIE	Transmit Data Empty Interrupt Enable 0: Transmit data empty interrupt request (INTRIICTI) is disabled. 1: Transmit data empty interrupt request (INTRIICTI) is enabled.
6	TEIE	Transmit End Interrupt Enable 0: Transmit end interrupt request (INTRIICTEI) is disabled. 1: Transmit end interrupt request (INTRIICTEI) is enabled.
5	RIE	Receive Data Full Interrupt Enable 0: Receive data full interrupt request (INTRIICRI) is disabled. 1: Receive data full interrupt request (INTRIICRI) is enabled.
4	NAKIE	NACK Reception Interrupt Enable 0: NACK reception interrupt request (INTRIICNAKI) is disabled. 1: NACK reception interrupt request (INTRIICNAKI) is enabled.
3	SPIE	Stop Condition Detection Interrupt Enable 0: Stop condition detection interrupt request (INTRIICSPI) is disabled. 1: Stop condition detection interrupt request (INTRIICSPI) is enabled.
2	STIE	Start Condition Detection Interrupt Enable 0: Start condition detection interrupt request (INTRIICSTI) is disabled. 1: Start condition detection interrupt request (INTRIICSTI) is enabled.
1	ALIE	Arbitration-Lost Interrupt Enable 0: Arbitration-lost interrupt request (INTRIICALI) is disabled. 1: Arbitration-lost interrupt request (INTRIICALI) is enabled.
0	TMOIE	Timeout Interrupt Enable 0: Timeout interrupt request (INTRIICTMOI) is disabled. 1: Timeout interrupt request (INTRIICTMOI) is enabled.

TMOIE Bit (Timeout Interrupt Enable)

This bit is used to enable or disable timeout interrupt requests (INTRIICTMOI) when the RIICnSR2.TMOF flag is set to 1. An INTRIICTMOI interrupt request is canceled by clearing the TMOF flag or the TMOIE bit to 0.

ALIE Bit (Arbitration-Lost Interrupt Enable)

This bit is used to enable or disable arbitration-lost interrupt requests (INTRIICALII) when the RIICnSR2.AL flag is set to 1. An INTRIICALII interrupt request is canceled by clearing the AL flag or the ALIE bit to 0.

STIE Bit (Start Condition Detection Interrupt Enable)

This bit is used to enable or disable start condition detection interrupt requests (INTRIICSTI) when the RIICnSR2.START flag is set to 1. An INTRIICSTI interrupt request is canceled by clearing the START flag or the STIE bit to 0.

SPIE Bit (Stop Condition Detection Interrupt Enable)

This bit is used to enable or disable stop condition detection interrupt requests (INTRIICSPI) when the RIICnSR2.STOP flag is set to 1. An INTRIICSPI interrupt request is canceled by clearing the STOP flag or the SPIE bit to 0.

NAKIE Bit (NACK Reception Interrupt Enable)

This bit is used to enable or disable NACK reception interrupt requests (INTRIICNAKI) when the RIICnSR2.NACKF flag is set to 1. An INTRIICNAKI interrupt request is canceled by clearing the NACKF flag or the NAKIE bit to 0.

RIE Bit (Receive Data Full Interrupt Enable)

This bit is used to enable or disable receive data full interrupt requests (INTRIICRI) when the RIICnSR2.RDRF flag in ICSR2 is set to 1. An INTRIICRI interrupt request is canceled by clearing the RDRF flag or the RIE bit to 0.

TEIE Bit (Transmit End Interrupt Enable)

This bit is used to enable or disable transmit end interrupts (INTRIICTEI) when the RIICnSR2.TEND flag is set to 1. An INTRIICTEI interrupt request is canceled by clearing the TEND flag or the TEIE bit to 0.

TIE Bit (Transmit Data Empty Interrupt Enable)

This bit is used to enable or disable transmit data empty interrupts (INTRIICTI) when the RIICnSR2.TDRE flag is set to 1.

23.3.9 RIICnSR1 — I²C Bus Status Register 1

Access: RIICnSR1 is a 32-bit readable/writable register.
RIICnSR1L and RIICnSR1H are 16-bit readable/writable registers.
RIICnSR1LL, RIICnSR1LH, RIICnSR1HL, and RIICnSR1HH are 8/1-bit readable/writable registers.

Address: RIICnSR1: <RIICn_base> + 0020_H
RIICnSR1L: <RIICn_base> + 0020_H, RIICnSR1H: <RIICn_base> + 0022_H
RIICnSR1LL: <RIICn_base> + 0020_H, RIICnSR1LH: <RIICn_base> + 0021_H, RIICnSR1HL: <RIICn_base> + 0022_H,
RIICnSR1HH: <RIICn_base> + 0023_H

Initial Value: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HOA	—	DID	—	GCA	AAS2	AAS1	AAS0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R(/W) *1	R	R(/W) *1	R	R/W *1	R(/W) *1	R(/W) *1	R(/W) *1

Note 1. Only 0 can be written to this bit.

Table 23.14 RIICnSR1 register contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are read as 0. The write value should be 0.
7	HOA	Host Address Detection Flag 0: Host address is not detected. 1: Host address is detected.
6	—	Reserved This bit is read as 0. The write value should be 0.
5	DID	Device-ID Address Detection Flag 0: Device-ID command is not detected. 1: Device-ID command is detected.
4	—	Reserved This bit is read as 0. The write value should be 0.
3	GCA	General Call Address Detection Flag 0: General call address is not detected. 1: General call address is detected.
2	AAS2	Slave Address 2 Detection Flag 0: Slave address 2 is not detected. 1: Slave address 2 is detected.
1	AAS1	Slave Address 1 Detection Flag 0: Slave address 1 is not detected. 1: Slave address 1 is detected.
0	AAS0	Slave Address 0 Detection Flag 0: Slave address 0 is not detected. 1: Slave address 0 is detected.

AASy Flag (Slave Address y Detection) (y = 0 to 2)

[Setting conditions]

<For 7-bit address format: RIICnSARy.FSy = 0>

When the received slave address matches the RIICnSARy.SVA[7:1] value with the RIICnSER.SARy bit set to 1 (slave address y detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

<For 10-bit address format: RIICnSARy.FSy = 1>

When the received slave address matches a value of (1111 0_B + RIICnSARy.SVA[9:8]) and the following address matches the RIICnSARy.SVA[7:0] value with the RIICnSER.SARy bit set to 1 (slave address y detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the AASy bit after reading AASy = 1
- When a stop condition is detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

<For 7-bit address format: RIICnSARy.FSy = 0>

- When the received slave address does not match the RIICnSARy.SVA[7:1] value with the RIICnSER.SARy bit set to 1 (slave address y detection enabled)

  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.

<For 10-bit address format: RIICnSARy.FSy = 1>

- When the received slave address does not match a value of (1111 0_B + RIICnSARy.SVA[9:8]) with the RIICnSER.SARy bit set to 1 (slave address y detection enabled)

  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When the received slave address matches a value of (1111 0_B + RIICnSARy.SVA[9:8]) and the following address does not match the RIICnSARy.SVA[7:0] value with the RIICnSER.SARy bit set to 1 (slave address y detection enabled)

  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.

GCA Flag (General Call Address Detection)

[Setting condition]

When the received slave address matches the general call address (0000 000_B + 0 [W]) with the RIICnSER.GCE bit set to 1 (general call address detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the GCA bit after reading GCA = 1
- When a stop condition is detected
- When the received slave address does not match the general call address (0000 000_B + 0 [W]) with the RIICnSER.GCE bit set to 1 (general call address detection enabled)

  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

DID Flag (Device-ID Address Detection)

[Setting condition]

- When the first frame received immediately after a start condition or restart condition is detected matches a value of (device ID (1111 100_B) + 0 [W]) with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled)

  This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.
- When a restart condition is detected after a match with the device ID address and the device ID address (1111 100_B) + 1 [R] has matched with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the DID bit after reading DID = 1
- When a stop condition is detected
- When the first frame received immediately after a start condition or restart condition is detected does not match a value of (device ID (1111 100_B)) with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When the first frame received immediately after a start condition or restart condition is detected matches a value of (device ID (1111 100_B) + 0 [W]) and the second frame does not match any of slave addresses 0 to 2 with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

HOA Flag (Host Address Detection)

[Setting condition]

When the received slave address matches the host address (0001 000_B) while the RIICnMR3.SMBE bit and RIICnSER.HOAE bit are set to 1 (host address detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the HOA bit after reading HOA = 1
- When a stop condition is detected
- When 0 is written to the RIICnMR3.SMBS bit in ICMR3 or the RIICnSER.HOAE bit
- When the received slave address does not match the host address (0001 000_B) with the RIICnSER.HOAE bit set to 1 (host address detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the RIICnCR1.IICRST bit 1 to apply an RIIC reset or an internal reset

23.3.10 RIICnSR2 — I²C Bus Status Register 2

Access: RIICnSR2 is a 32-bit readable/writable register.
RIICnSR2L and RIICnSR2H are 16-bit readable/writable registers.
RIICnSR2LL, RIICnSR2LH, RIICnSR2HL, and RIICnSR2HH are 8/1-bit readable/writable registers.

Address: RIICnSR2: <RIICn_base> + 0024_H
RIICnSR2L: <RIICn_base> + 0024_H, RIICnSR2H: <RIICn_base> + 0026_H
RIICnSR2LL: <RIICn_base> + 0024_H, RIICnSR2LH: <RIICn_base> + 0025_H, RIICnSR2HL: <RIICn_base> + 0026_H,
RIICnSR2HH: <RIICn_base> + 0027_H

Initial Value: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TDRE	TEND	RDRF	NACKF	STOP	START	AL	TMOF
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R(/W) *1	R(/W) *1	R(/W) *1	R(/W) *1	R(/W) *1	R(/W) *1	R(/W) *1

Note 1. Only 0 can be written to this bit.

Table 23.15 RIICnSR2 register contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are read as 0. The write value should be 0.
7	TDRE	Transmit Data Empty Flag 0: RIICnDRT contains transmit data. 1: RIICnDRT contains no transmit data.
6	TEND	Transmit End Flag 0: Data is being transmitted. 1: Data has been transmitted.
5	RDRF	Receive Data Full Flag 0: RIICnDRR contains no receive data. 1: RIICnDRR contains receive data.
4	NACKF	NACK Reception Flag 0: NACK is not received. 1: NACK is received.
3	STOP	Stop Condition Detection Flag 0: Stop condition is not detected. 1: Stop condition is detected.
2	START	Start Condition Detection Flag 0: Start condition is not detected. 1: Start condition is detected.
1	AL	Arbitration-Lost Flag 0: Arbitration is not lost. 1: Arbitration is lost.
0	TMOF	Timeout Flag 0: No timeout has occurred. 1: Timeout has occurred.

TMOF Flag (Timeout)

This flag is set to 1 when the RIIC recognizes timeout after the SCL line state remains unchanged for a certain period.
[Setting condition]
The timeout function is enabled when the RIICnFER.TMOE bit is 1. It detects an abnormal bus state that the SCL line is held low or high during the following conditions:

- The bus is busy (RIICnCR2.BBSY = 1) in master mode (RIICnCR2.MST = 1).
- The slave address matches that of this module (RIICnSR1 register is not 00_H) and the bus is busy (RIICnCR2.BBSY = 1) in slave mode (RIICnCR2.MST = 0).
- Issuing of a start condition is being requested (RIICnCR2.ST = 1) and the bus is free (RIICnCR2.BBSY = 0).

[Clearing conditions]

- When 0 is written to the TMOF bit after reading TMOF = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

AL Flag (Arbitration-Lost)

This flag shows that bus mastership has been lost (loss in arbitration) due to a bus conflict or some other reason when a start condition is issued or an address and data are transmitted. The RIIC monitors the level on the SDA line during transmission and, if the level on the line does not match the value of the bit being output, sets the value of the AL bit to 1 to indicate that the bus is occupied by another device.
The RIIC can also set the flag to indicate the detection of loss of arbitration during NACK transmission in receive mode or during data transmission in slave mode.
[Setting conditions]
<When master arbitration-lost detection is enabled: RIICnFER.MALE = 1>

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data (including slave address) transmission in master transmit mode (when the SDA line is driven low while the internal SDA output is at a high level (the SDA pin is in the high-impedance state))
- When a start condition is detected while the RIICnCR2.ST bit is 1 (start condition issuance request) or the internal SDA output state does not match the SDA line level
- When the RIICnCR2.ST bit is set to 1 (start condition issuance request) with the RIICnCR2.BBSY flag set to 1.

<When NACK arbitration-lost detection is enabled: RIICnFER.NALE = 1>
When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock in the ACK period during NACK transmission in receive mode
<When slave arbitration-lost detection is enabled: RIICnFER.SALE = 1>
When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data transmission in slave transmit mode
[Clearing conditions]

- When 0 is written to the AL bit after reading AL = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

Table 23.16 Relationship between Arbitration-Lost Generation Sources and Arbitration-Lost Enable Functions

RIICnFER			RIICnSR2		
MALE	NALE	SALE	AL	Error	Arbitration-Lost Generation Source
1	x	x	1	Start condition issuance error	When internal SDA output state does not match SDA line level when a start condition is detected while the RIICnCR2.ST bit is 1
					When RIICnCR2.ST is set to 1 with RIICnCR2.BBSY set to 1
			1	Transmit data mismatch	When transmit data (including slave address) does not match the bus state in master transmit mode
x	1	x	1	NACK transmission mismatch	When ACK is detected during transmission of NACK in master receive mode or slave receive mode
x	x	1	1	Transmit data mismatch	When transmit data does not match the bus state in slave transmit mode

x: Don't care

START Flag (Start Condition Detection)

[Setting condition]

When a start condition (or a restart condition) is detected

[Clearing conditions]

- When 0 is written to the START bit after reading START = 1
- When a stop condition is detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

STOP Flag (Stop Condition Detection)

[Setting condition]

When a stop condition is detected

[Clearing conditions]

- When 0 is written to the STOP bit after reading STOP = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

NACKF Flag (NACK Reception)

[Setting condition]

When acknowledge is not received (NACK is received) from the receive device in transmit mode with the RIICnFER.NACKE bit set to 1 (transfer suspension enabled)

[Clearing conditions]

- When 0 is written to the NACKF bit after reading NACKF = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

Caution: When the NACKF flag is set to 1, the RIIC suspends data transmission/reception. Writing to RIICnDRT in transmit mode or reading from RIICnDRR in receive mode with the NACKF flag set to 1 does not enable data transmit/receive operation. To restart data transmission/reception, clear the NACKF flag to 0.

RDRF Flag (Receive Data Full)

[Setting conditions]

- Slave receive mode
  - When the received slave address matches and the RIICnCR2.TRS bit is cleared to 0 after a start condition (or a restart condition) is detected
  - At the rising edge of the eighth or ninth SCL clock cycle (selected by the RIICnMR3.RDRFS bit) after receive data is transferred from RIICnDRS to RIICnDRR
- Master receive mode
  - When the slave address and the data direction are transmitted and the receive mode is entered (the RIICnCR2.TRS bit is set to 1) after a start condition (or a restart condition) is issued
  - At the rising edge of the eighth or ninth SCL clock cycle (selected by the RIICnMR3.RDRFS bit) after receive data is transferred from RIICnDRS to RIICnDRR

[Clearing conditions]

- When 0 is written to the RDRF bit after reading RDRF = 1
- When data is read from RIICnDRR
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

TEND Flag (Transmit End)

[Setting condition]

At the rising edge of the ninth SCL clock cycle while the TDRE flag is 1

[Clearing conditions]

- When 0 is written to the TEND bit after reading TEND = 1
- When data is written to RIICnDRT
- When a stop condition is detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

TDRE Flag (Transmit Data Empty)

[Setting conditions]

- When data has been transferred from RIICnDRT to RIICnDRS and RIICnDRT becomes empty
- When the RIICnCR2.TRS bit is set to 1
  - When the RIICnCR2.MST bit is set to 1 after a start condition (or a restart condition) is detected
  - When the RIIC enters transmit mode from receive mode
- When the received slave address matches while the TRS bit is 1

[Clearing conditions]

- When data is written to RIICnDRT
- When the RIICnCR2.TRS bit is cleared to 0
  - When a stop condition is detected
  - When the RIIC enters receive mode from transmit mode
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

Caution: When the NACKF flag is set to 1 while the RIICnFER.NACKE bit is 1, the RIIC suspends data transmission/reception. Here, if the TDRE flag is 0 (next transmit data has been written), data is transferred to the RIICnDRS register and the RIICnDRT register becomes empty at the rising edge of the ninth clock cycle, but the TDRE flag is not set to 1.

23.3.11 RIICnSARy — I²C Slave Address Register y (y = 0 to 2)

Access: RIICnSARy is a 32-bit readable/writable register.
RIICnSARyL and RIICnSARyH are 16-bit readable/writable registers.
RIICnSARyLL, RIICnSARyLH, RIICnSARyHL, and RIICnSARyHH are 8-bit readable/writable registers.

Address: RIICnSAR0: <RIICn_base> + 0028_H
RIICnSAR0L: <RIICn_base> + 0028_H, RIICnSAR0H: <RIICn_base> + $002A_H$
RIICnSAR0LL: <RIICn_base> + 0028_H, RIICnSAR0LH: <RIICn_base> + 0029_H, RIICnSAR0HL: <RIICn_base> + $002A_H$,
RIICnSAR0HH: <RIICn_base> + $002B_H$
RIICnSAR1: <RIICn_base> + $002C_H$
RIICnSAR1L: <RIICn_base> + $002C_H$, RIICnSAR1H: <RIICn_base> + $002E_H$
RIICnSAR1LL: <RIICn_base> + $002C_H$, RIICnSAR1LH: <RIICn_base> + $002D_H$, RIICnSAR1HL: <RIICn_base> + $002E_H$,
RIICnSAR1HH: <RIICn_base> + $002F_H$
RIICnSAR2: <RIICn_base> + 0030_H
RIICnSAR2L: <RIICn_base> + 0030_H, RIICnSAR2H: <RIICn_base> + 0032_H
RIICnSAR2LL: <RIICn_base> + 0030_H, RIICnSAR2LH: <RIICn_base> + 0031_H, RIICnSAR2HL: <RIICn_base> + 0032_H,
RIICnSAR2HH: <RIICn_base> + 0033_H

Initial Value: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FSy	—	—	—	—	—	SVA[9:1]									SVA0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.17 RIICnSARy register contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are read as 0. The write value should be 0.
15	FSy	7-Bit/10-Bit Address Format Selection 0: The 7-bit address format is selected. 1: The 10-bit address format is selected.
14 to 10	—	Reserved These bits are read as 0. The write value should be 0.
9 to 1	SVA[9:1]	7-Bit Address/10-Bit Address Upper Bits A slave address is set. • When the FSy bit is 0 (7-bit address format), the SVA[7:1] bits are Valid and form a 7-bit slave address. • When the FSy bit is 1 (10-bit address format), SVA[9:1] bits form a 10-bit slave address (combined with the SVA0 bit).
0	SVA0	10-Bit Address LSB The least significant bit (LSB) of a 10-bit slave address is set. • When the FSy bit is 0 (7-bit address format), this bit is invalid. • When the FSy bit is 1 (10-bit address format), this bit is a 10-bit slave address (combined with the SVA[9:1] bits).

SVA0 Bit (10-Bit Address LSB)

When the 10-bit address format is selected (RIICnSARy.FSy = 1), this bit functions as the LSB of a 10-bit address and forms a 10-bit address in combination with the SVA[9:1] bits.
When the RIICnSER.SARy bit is set to 1 (RIICnSARy enabled) and the RIICnSARy.FSy bit is 1, this bit is valid. While the RIICnSARy.FSy bit or SARy bit is 0, the setting of this bit is ignored.

SVA[9:1] Bits (7-Bit Address/10-Bit Address Upper Bits)

When the 7-bit address format is selected (RIICnSARy.FSy = 0), these bits function as a 7-bit address. When the 10-bit address format is selected (RIICnSARy.FSy = 1), these bits function as a 10-bit address in combination with the SVA0 bit.
While the RIICnSER.SARy bit is 0, the setting of these bits is ignored.

FSy Bit (7-Bit/10-Bit Address Format Selection)

This bit is used to select 7-bit address or 10-bit address for slave address y (in RIICnSARy).
When the RIICnSER.SARy bit is set to 1 (RIICnSARy enabled) and the RIICnSARy.FSy bit is 0, the 7-bit address format is selected for slave address y, the RIICnSARy.SVA[7:1] setting is valid, and the settings of the SVA[9:8] bits and the RIICnSARy.SVA0 bit are ignored.
When the RIICnSER.SARy bit is set to 1 (RIICnSARy enabled) and the RIICnSARy.FSy bit is 1, the 10-bit address format is selected for slave address y and the settings of the SVA[9:1] bits and the SVA0 bit are valid.
While the RIICnSER.SARy bit is 0 (RIICnSARy disabled), the setting of the RIICnSARy.FSy bit is invalid.

23.3.12 RIICnBRL — I²C Bus Bit Rate Low-Level Register

Access: RIICnBRL is a 32-bit readable/writable register.
RIICnBRLL and RIICnBRLH are 16-bit readable/writable registers.
RIICnBRLLL, RIICnBRLLH, RIICnBRLHL, and RIICnBRLHH are 8-bit readable/writable registers.

Address: RIICnBRL: <RIICn_base> + 0034_H
RIICnBRLL: <RIICn_base> + 0034_H, RIICnBRLH: <RIICn_base> + 0036_H
RIICnBRLLL: <RIICn_base> + 0034_H, RIICnBRLLH: <RIICn_base> + 0035_H, RIICnBRLHL: <RIICn_base> + 0036_H,
RIICnBRLHH: <RIICn_base> + 0037_H

Initial Value: 0000 $00FF_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	BRL[4:0]				
Initial value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 23.18 RIICnBRL register contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are read as 0. The write value should be 0.
7 to 5	—	Reserved These bits are read as 1. The write value should be 1.
4 to 0	BRL[4:0]	Bit Rate Low-Level Period Low-level period of SCL clock

The RIICnBRL register is a 5-bit register that is used to set the width at low level for the SCL clock.
It also works to generate the data setup time for automatic SCL low-hold operation (see section 23.10, Automatically Low-Hold Function for SCL); when the RIIC is used only in slave mode, this register needs to be set to a value at least the same as the data setup time*1.
RIICnBRL counts the low-level period with the internal reference clock source (IICφ) specified by the RIICnMR1.CKS[2:0] bits.

Note 1. Data setup time ($t_{SU: DAT}$)
250 [ns] (0 to 100 [kbps]: standard mode (Sm))
100 [ns] (0 to 400 [kbps]: fast mode (Fm))
50 [ns] (0 to 1 [Mbps]: fast mode plus (Fm+))

23.3.13 RIICnBRH — I²C Bus Bit Rate High-Level Register

Access: RIICnBRH is a 32-bit readable/writable register.
RIICnBRHL and RIICnBRHH are 16-bit readable/writable registers.
RIICnBRHLL, RIICnBRHLH, RIICnBRHHL, and RIICnBRHHH are 8-bit readable/writable registers.

Address: RIICnBRH: <RIICn_base> + 0038_H
RIICnBRHL: <RIICn_base> + 0038_H, RIICnBRHH: <RIICn_base> + $003A_H$
RIICnBRHLL: <RIICn_base> + 0038_H, RIICnBRHLH: <RIICn_base> + 0039_H, RIICnBRHHL: <RIICn_base> + $003A_H$, RIICnBRHHH:<RIICn_base> + $003B_H$

Initial Value: 0000 $00FF_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	BRH[4:0]				
Initial value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 23.19 RIICnBRH register contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are read as 0. The write value should be 0.
7 to 5	—	Reserved These bits are read as 1. The write value should be 1.
4 to 0	BRH[4:0]	Bit Rate High-Level Period High-level period of SCL clock

RIICnBRH is a 5-bit register to set the high-level period of SCL clock. RIICnBRH is valid in master mode. If the RIIC is used only in slave mode, this register need not to set the high-level period.
RIICnBRH counts the high-level period with the internal reference clock source (IICφ) specified by the RIICnMR1.CKS[2:0] bits in ICMR1.
The frequency and duty cycle are calculated using one of the following expressions (1) to (5) according to the register settings.

Caution: The minimum value that can be specified in RIICnBRL and RIICnBRH is determined according to the values of the SCLE and NFE bits in RIICnFER and the NF bit in RIICnMR3. For details of the minimum specifiable value, see Table 23.20.

(1) When SCLE = 0
Frequency = 1 / {[(BRH + 1) + (BRL + 1)] / IICφ + tr + tf}
Duty cycle = {tr + (BRH + 1) / IICφ} / {tr + tf + [(BRH + 1) + (BRL + 1)] / IICφ}

(2) When SCLE = 1, NFE = 0, CKS = 000 (IICφ = P1φ)
Frequency = 1 / {[(BRH + 3) + (BRL + 3)] / IICφ + tr + tf}
Duty cycle = {tr + (BRH + 3) / IICφ} / {tr + tf + [(BRH + 3) + (BRL + 3)] / IICφ}

(3) When SCLE = 1, NFE = 1, CKS = 000 (IICφ = P1φ)
Frequency = 1 / {[(BRH + 3 + nf) + (BRL + 3 + nf)] / IICφ + tr + tf}
Duty cycle = {tr + (BRH + 3 + nf) / IICφ} / {tr + tf + [(BRH + 3 + nf) + (BRL + 3 + nf)] / IICφ}

(4) When SCLE = 1, NFE = 0, CKS ¹ 000 (IICφ< P1φ)
Frequency = 1 / {[(BRH + 2) + (BRL + 2)] / IICφ + tr + tf}
Duty cycle = {tr + (BRH + 2) / IICφ} / {tr + tf + [(BRH + 2) + (BRL + 2)] / IICφ}

(5) When SCLE = 1, NFE = 1, CKS ¹ 000 (IICφ < P1φ)
Frequency = 1 / {[(BRH + 2 + nf) + (BRL + 2 + nf)] / IICφ + tr + tf}
Duty cycle = {tr + (BRH + 2 + nf) / IICφ} / {tr + tf + [(BRH + 2 + nf) + (BRL + 2 + nf)] / IICφ}

Symbols in the expressions
SCLE: RIICnFER.SCLE bit
BRH: RIICnBRH.BRH[4:0] bits
BRL: RIICnBRL.BRL[4:0] bits
CKS: RIICnMR1.CKS bits
NFE: RIICnFER.NFE bit
IICφ: Internal reference clock selected by the CKS bits
tf: SCL signal falling time [s]*1
tr: SCL signal rising time [s]*1
nf: Number of digital noise filter stages specified in the RIICnMR3.NF[1:0] bits

Note 1. The rising time (tr) and falling time (tf) of the SCL signal depend on the total capacitance of the bus line (Cb) and pull-up resistor (Rp). For details, see I²C Bus Standard from NXP Semiconductors.

Table 23.20 Minimum Specifiable Value for RIICnBRL and RIICnBRH

SCLE	NFE	nf	Minimum Pulse Width that Passes through Digital Filter	Minimum Specifiable Value for BRH and BRL	Pulse Width when Minimum Value is Specified
0	0	—	1 × IICφ	1	2 × IICφ
0	1	1	2 × IICφ	2	3 × IICφ
0	1	2	3 × IICφ	3	4 × IICφ
0	1	3	4 × IICφ	4	5 × IICφ
0	1	4	5 × IICφ	5	6 × IICφ
IICφ cycle > P1φ cycle (CKS ≠ 000)					
1	0	—	1 × IICφ	0	2 × IICφ
1	1	1	2 × IICφ	1	4 × IICφ
1	1	2	3 × IICφ	2	6 × IICφ
1	1	3	4 × IICφ	3	8 × IICφ
1	1	4	5 × IICφ	4	10 × IICφ
IICφ cycle = P1φ cycle (CKS = 000)					
1	0	—	2 × P1φ	0	3 × IICφ
1	1	1	3 × P1φ	1	5 × IICφ
1	1	2	4 × P1φ	2	7 × IICφ
1	1	3	5 × P1φ	3	9 × IICφ
1	1	4	6 × P1φ	4	11 × IICφ

Table 23.21 and Table 23.22 list examples of RIICnBRH/RIICnBRL settings.

Table 23.21 Examples of RIICnBRH/RIICnBRL Settings for Transfer Rate (when RIICnFER.SCLE = 1 and RIICnFER.NFE= 0)

Transfer Rate [kbps]	Peripheral Clock Operating Frequency P1φ [MHz]								
	55			60			66		
	RIICnMR1. CKS[2:0]	RIICnBRH. BRH	RIICnBRL. BRL	RIICnMR1. CKS[2:0]	RIICnBRH. BRH	RIICnBRL. BRL	RIICnMR1. CKS[2:0]	RIICnBRH. BRH	RIICnBRL. BRL
10	111_B	18 ($F2_H$)	21 ($F5_H$)	111_B	18 ($F2_H$)	25 ($F9_H$)	111_B	24 ($F8_H$)	24 ($FF8_H$)
50	101_B	15 (EF_H)	16 ($F0_H$)	101_B	13 (ED_H)	20 ($F4_H$)	101_B	15 (EF_H)	22 ($F6_H$)
100	100_B	15 (EF_H)	16 ($F0_H$)	100_B	16 ($F0_H$)	18 ($F2_H$)	100_B	18 ($F2_H$)	20 ($F4_H$)
400	010_B	13 (ED_H)	18 ($F2_H$)	010_B	16 ($F0_H$)	18 ($F2_H$)	010_B	17 ($F1_H$)	21 ($F5_H$)
1000	001_B	7 ($E7_H$)	15 (EF_H)	001_B	10 (EA_H)	14 (EE_H)	001_B	12 (EC_H)	15 (EF_H)

Table 23.22 Examples of RIICnBRH/RIICnBRL Settings for Transfer Rate (when RIICnFER.SCLE = 1, RIICnFER.NFE= 1, and Number of NF Stages = 4)

Transfer Rate [kbps]	Peripheral Clock Operating Frequency P1φ [MHz]								
	55			60			66		
	RIICnMR1. CKS[2:0]	RIICnBRH. BRH	RIICnBRL. BRL	RIICnMR1. CKS[2:0]	RIICnBRH. BRH	RIICnBRL. BRL	RIICnMR1. CKS[2:0]	RIICnBRH. BRH	RIICnBRL. BRL
10	111_B	12 (EC_H)	19 ($F3_H$)	111_B	11 (EB_H)	24 ($F8_H$)	111_B	12 (EC_H)	28 (FC_H)
50	101_B	10 (EA_H)	13 (ED_H)	101_B	7 ($E7_H$)	19 ($F3_H$)	101_B	10 (EA_H)	20 ($F4_H$)
100	100_B	10 (EA_H)	13 (ED_H)	100_B	12 (EC_H)	14 (EE_H)	100_B	14 (EE_H)	16 ($F0_H$)
400	010_B	9 ($E9_H$)	14 (EE_H)	010_B	10 (EA_H)	16 ($F0_H$)	010_B	12 (EC_H)	18 ($F2_H$)
1000	001_B	6 ($E6_H$)	8 ($E8_H$)	001_B	6 ($E6_H$)	10 (EA_H)	001_B	7 ($E7_H$)	12 (EC_H)

23.3.14 RIICnDRT — I²C Bus Transmit Data Register

Access: RIICnDRT is a 32-bit readable/writable register.
RIICnDRTL and RIICnDRTH are 16-bit readable/writable registers.
RIICnDRTLL, RIICnDRTLH, RIICnDRTHL, and RIICnDRTHH are 8-bit readable/writable registers.

Address: RIICnDRT: <RIICn_base> + $003C_H$
RIICnDRTL: <RIICn_base> + $003C_H$, RIICnDRTH: <RIICn_base> + $003E_H$
RIICnDRTLL: <RIICn_base> + $003C_H$, RIICnDRTLH: <RIICn_base> + $003D_H$, RIICnDRTHL: <RIICn_base> + $003E_H$,
RIICnDRTHH: <RIICn_base> + $003F_H$

Initial Value: 0000 $00FF_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DRT[7:0]							
Initial value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When RIICnDRT detects a space in the I²C bus shift register (RIICnDRS), it transfers the transmit data that has been written to RIICnDRT to RIICnDRS and starts transmitting data in transmit mode.
The double-buffer structure of RIICnDRT and RIICnDRS allows continuous transmit operation if the next transmit data has been written to RIICnDRT while the RIICnDRS data is being transmitted.
RIICnDRT can always be read and written. Write transmit data to RIICnDRT once when a transmit data empty interrupt (INTRIICTI) request is generated. When writing to bits 8 to 15, be sure to write 0 to these bits.

23.3.15 RIICnDRR — I²C Bus Receive Data Register

Access: RIICnDRR is a 32-bit readable/writable register.
RIICnDRRL and RIICnDRRH are 16-bit readable/writable registers.
RIICnDRRLL, RIICnDRRLH, RIICnDRRHL, and RIICnDRRHH are 8-bit readable/writable registers.

Address: RIICnDRR: <RIICn_base> + 0040_H
RIICnDRRL: <RIICn_base> + 0040_H, RIICnDRRH: <RIICn_base> + 0042_H
RIICnDRRLL: <RIICn_base> + 0040_H, RIICnDRRLH: <RIICn_base> + 0041_H, RIICnDRRHL: <RIICn_base> + 0042_H,
RIICnDRRHH: <RIICn_base> + 0043_H

Initial Value: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DRR[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

When 1 byte of data has been received, the received data is transferred from the I²C bus shift register (RIICnDRS) to RIICnDRR to enable the next data to be received.
The double-buffer structure of RIICnDRS and RIICnDRR allows continuous receive operation if the received data has been read from RIICnDRR while RIICnDRS is receiving data.
RIICnDRR cannot be written. Read data from RIICnDRR once when a receive data full interrupt (INTRIICRI) request is generated.
If DRR receives the next receive data before the current data is read from RIICnDRR (while the RIICnSR2.RDRF flag is 1), the RIIC automatically holds the SCL clock low one cycle before the RDRF flag is set to 1 next.

23.3.16 RIICnDRS — I²C Bus Shift Register

Access: This register is not accessible.
Address: —
Initial Value: 0000 00FF$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DRS[7:0]							
Initial value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

RIICnDRS is an 8-bit shift register to transmit and receive data.
During transmission, transmit data is transferred from RIICnDRT to RIICnDRS and is sent from the SDA pin. During reception, data is transferred from RIICnDRS to RIICnDRR after 1 byte of data has been received.
RIICnDRS cannot be accessed directly.

23.4 Interrupt Sources

The RIIC issues eight types of interrupt request: transmit end, receive data full, transmit data empty, stop condition detection, start condition detection, NACK reception, arbitration-lost, and timeout.

Table 23.23 lists details of the several interrupt requests. The receive data full and transmit data empty sources are both capable of launching data transfer by the DMAC.

Table 23.23 Interrupt Sources

Symbol	Interrupt Source	Interrupt Flag	DMAC Launching	Priority*1	Interrupt Condition
INTRIICTEI	Transmission complete	TEND	Not possible	High	TEND = 1 • TEIE = 1
INTRIICRI	Receive-data-full	RDRF	Possible	↑	RDRF = 1 • RIE = 1
INTRIICTI	Transmit-data-empty	TDRE	Possible		TDRE = 1 • TIE = 1
INTRIICSPI	Detection of a stop condition	STOP	Not possible		STOP = 1 • SPIE = 1
INTRIICSTI	Detection of a start condition	START	Not possible		START = 1 • STIE = 1
INTRIICNAKI	Reception of a NACK	NACKF	Not possible		NACKF = 1 • NAKIE = 1
INTRIICALI	Arbitration lost	AL	Not possible		AL = 1 • ALIE = 1
INTRIICTMOI	Timeout	TMOF	Not possible	Low	TMOF = 1 • TMOIE = 1

Note 1. When the interrupt priority register (ICDIPRn) setting is the same

Clear or mask the each flag during interrupt handling.

Caution:1. There is a latency (delay) between the execution of a write instruction for a peripheral module by the CPU and actual writing to the module. Thus, when an interrupt flag has been cleared or masked, read the relevant flag again to check whether clearing or masking has been completed, and then return from interrupt processing. Returning from interrupt processing without checking that writing to the module has been completed creates a possibility of repeated processing of the same interrupt.
Caution:2. Since INTRIICRI and INTRIICTI are edge-detected interrupts, they do not require clearing.
Caution:3. When using the INTRIICTEI interrupt, clear the RIICnSR2.TEND flag in the INTRIICTEI interrupt processing.
Caution:4. When using the INTRIICSPI interrupt, clear the RIICnSR2.STOP flag in the INTRIICSPI interrupt processing.
Caution:5. When using the INTRIICSTI interrupt, clear the RIICnSR2.START flag in the INTRIICSTI interrupt processing.
Caution:6. When using the INTRIICNAKI interrupt, clear the RIICnSR2.NACKF flag in the INTRIICNAKI interrupt processing.
Caution:7. When using the INTRIICALI interrupt, clear the RIICnSR2.AL flag in the INTRIICALI interrupt processing.
Caution:8. When using the INTRIICTMOI interrupt, clear the RIICnSR2.TMOF flag in the INTRIICTMOI interrupt processing.

23.5 Operation

23.5.1 Communication Data Format

The I²C bus format consists of 8-bit data and 1-bit acknowledge (one frame). After a start condition or restart condition is issued, the master device sends the slave address and data direction in the first frame. The specified slave is valid until a stop condition is issued or a new slave is specified by a restart condition.
Figure 23.3 shows the I²C bus format, and Figure 23.4 shows the I²C bus timing.

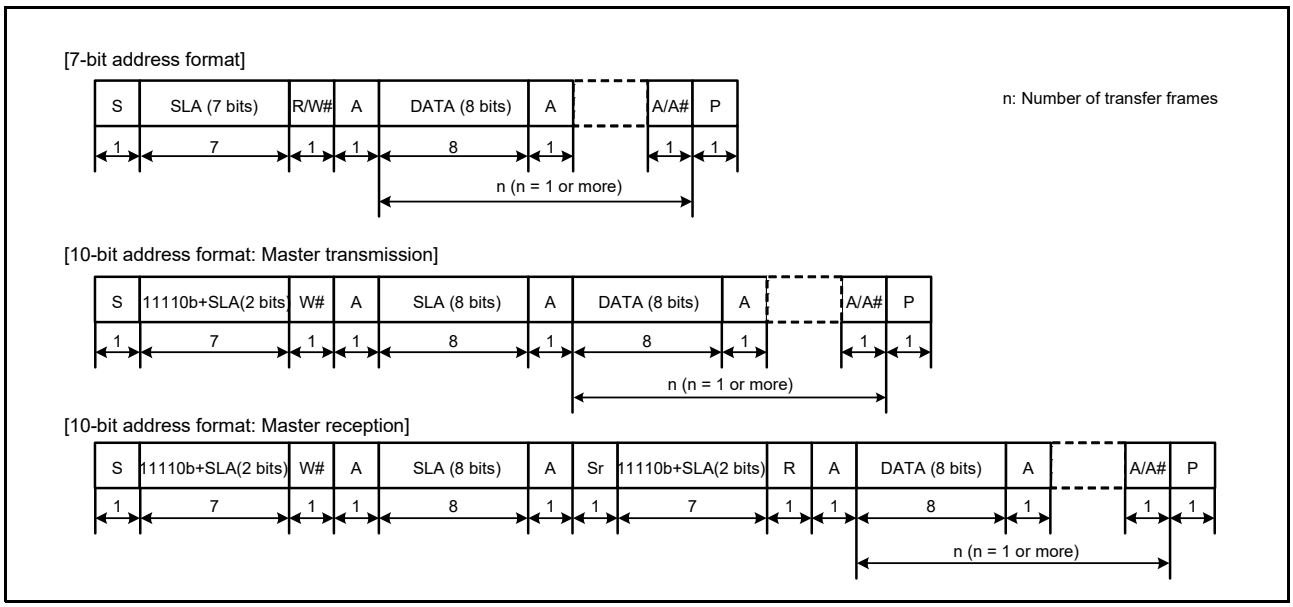


Figure 23.3 I²C Bus Format

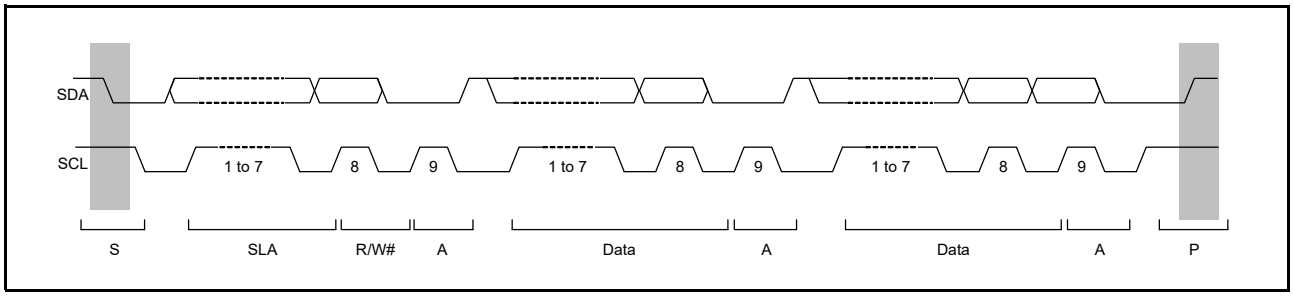


Figure 23.4 I²C Bus Timing (SLA = 7 Bits)

S: Start condition. The master device drives the SDA line low from high level while the SCL line is at a high level.

SLA: Slave address, by which the master device selects a slave device.

R/W#: Indicates the direction of data transfer: from the slave device to the master device when R/W is 1, or from the master device to the slave device when R/W is 0.

A: Acknowledge. The receive device drives the SDA line low. (In master transmit mode, the slave device returns acknowledge. In master receive mode, the master device returns acknowledge.)

A#: Not-acknowledge. The receiving device has not returned a response or is not present so the SDA line has remained at the high level.

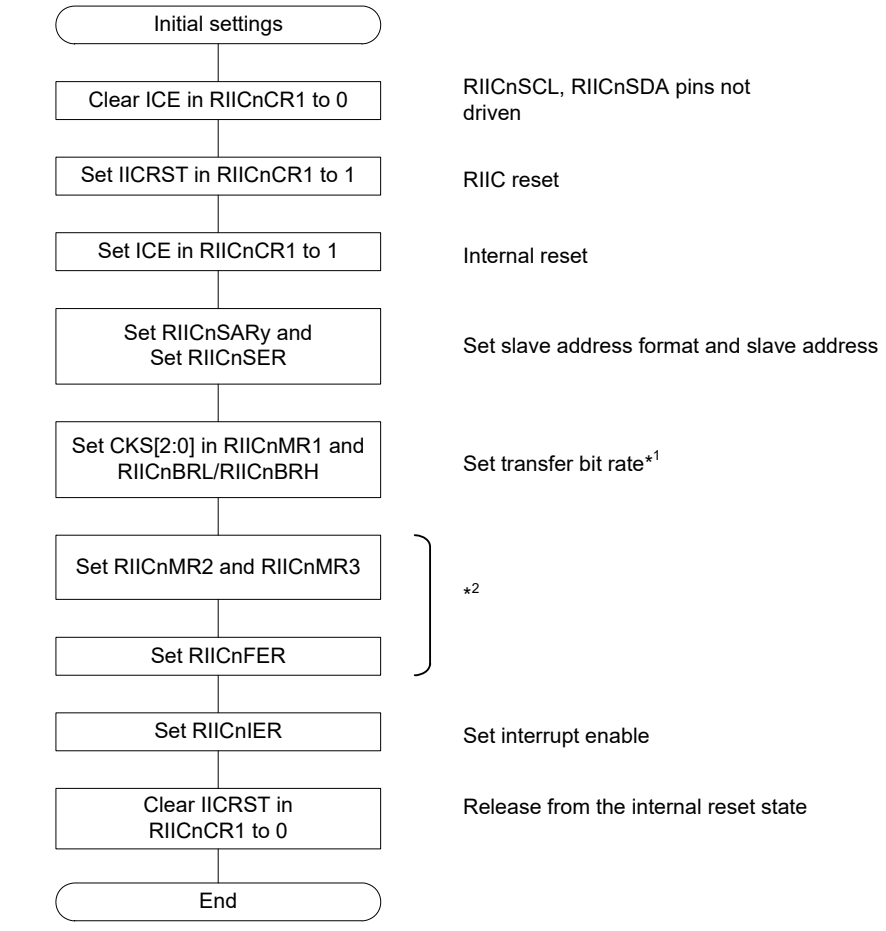
Sr: Restart condition. The master device drives the SDA line low from the high level after the setup time has elapsed with the SCL line at the high level.

DATA: Transmitted or received data

P: Stop condition. The master device drives the SDA line high from low level while the SCL line is at a high level.

23.5.2 Initial Settings

Before starting data transmission and reception, initialize the RIIC according to the procedure in Figure 23.5.



Note 1. When the RIIC is used only in slave mode, set the RIICnBRL register to a value at least the same as the data setup time.
Note 2. Set these registers as necessary.

Figure 23.5 Example of RIIC Initialization Flowchart

23.5.3 Master Transmit Operation

In master transmit operation, the RIIC outputs the SCL (clock) and transmitted data signals as the master device, and the slave device returns acknowledgements. Figure 23.6 shows an example of usage of master transmission and Figure 23.7 to Figure 23.9 show the timing of operations in master transmission.

The following describes the procedure and operations for master transmission.

(1) Set the RIICnCR1.IICRST bit 1 to 1 (RIIC reset) and then set the RIICnCR1.ICE bit to 1 (internal reset) with the RIICnCR1.ICE bit cleared to 0 (RIICnSCL and RIICnSDA pins not driven). This initializes the internal state and the various flags of RIICnSR1. After that, set registers RIICnSARy, RIICnSER, RIICnMR1, RIICnBRH, and RIICnBRL (y = 0 to 2), and set the other registers as necessary (for initial settings of the RIIC, see Figure 23.5). When the necessary register settings have been completed, set the RIICnCR1.IICRST bit to 0 (for release from the reset state). This step is not necessary if initialization of the RIIC has already been completed.

(2) Read the RIICnCR2.BBSY flag to check that the bus is open, and then set the RIICnCR2.ST bit to 1 (start condition issuance request). Upon receiving the request, the RIIC issues a start condition. At the same time, the BBSY flag and the RIICnSR2.START flag are automatically set to 1 and the ST bit is automatically cleared to 0. At this time, if the start condition is detected and the internal levels for the SDA output state and the levels on the SDA line have matched while the ST bit is 1, the RIIC recognizes that issuing of the start condition as requested by the ST bit has been successfully completed, and the RIICnCR2.MST and TRS bits are automatically set to 1, placing the RIIC in master transmit mode. The RIICnSR2.TDRE flag is also automatically set to 1 in response to setting of the TRS bit to 1.

(3) Check that the RIICnSR2.TDRE flag is 1, and then write the value for transmission (the slave address and the R/W# bit) to RIICnDRT. Once the data for transmission are written to RIICnDRT, the TDRE flag is automatically cleared to 0, the data are transferred from RIICnDRT to RIICnDRS, and the TDRE flag is again set to 1. After the byte containing the slave address and R/W# bit has been transmitted, the value of the TRS bit is automatically updated to select master transmit or master receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 0, the RIIC continues in master transmit mode.
Since the RIICnSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the RIICnCR2.SP bit to issue a stop condition.
For data transmission with an address in the 10-bit format, start by writing 1111 0_B, the two higher-order bits of the slave address, and W# to RIICnDRT as the first address transmission. Then, as the second address transmission, write the eight lower-order bits of the slave address to RIICnDRT.

(4) After confirming that the RIICnSR2.TDRE flag is 1, write the data for transmission to the RIICnDRT register. The RIIC automatically holds the SCL line low until the data for transmission are ready or a stop condition is issued.

(5) After the last byte of the data to be transmitted is written to the RIICnDRT register, wait until the value of the RIICnSR2.TEND flag returns to 1, and then set the RIICnCR2.SP bit to 1 (stop condition issuance request). Upon receiving a stop condition issuance request, the RIIC issues the stop condition.

(6) Upon detecting the stop condition, the RIIC automatically clears the RIICnCR2.MST and TRS bits to 00_B and enters slave receive mode. Furthermore, it automatically clears the RIICnSR2.TDRE and TEND flags to 0, and sets the RIICnSR2.STOP flag in to 1.

(7) Clear the RIICnSR2.NACKF and STOP flags to 0.

Caution: Operations for transfer start if the RIICnSR2.NACKF flag is cleared to 0 before RIICnSR2.STOP is set to 1. Be sure to confirm that RIICnSR2.STOP is set to 1 before clearing RIICnSR2.NACKF to 0. In particular, when the NACK receive interrupt (INTRIICNAKI) is in use, take care not to clear the NACKF flag to 0 before the STOP flag is set to 1 during interrupt processing.

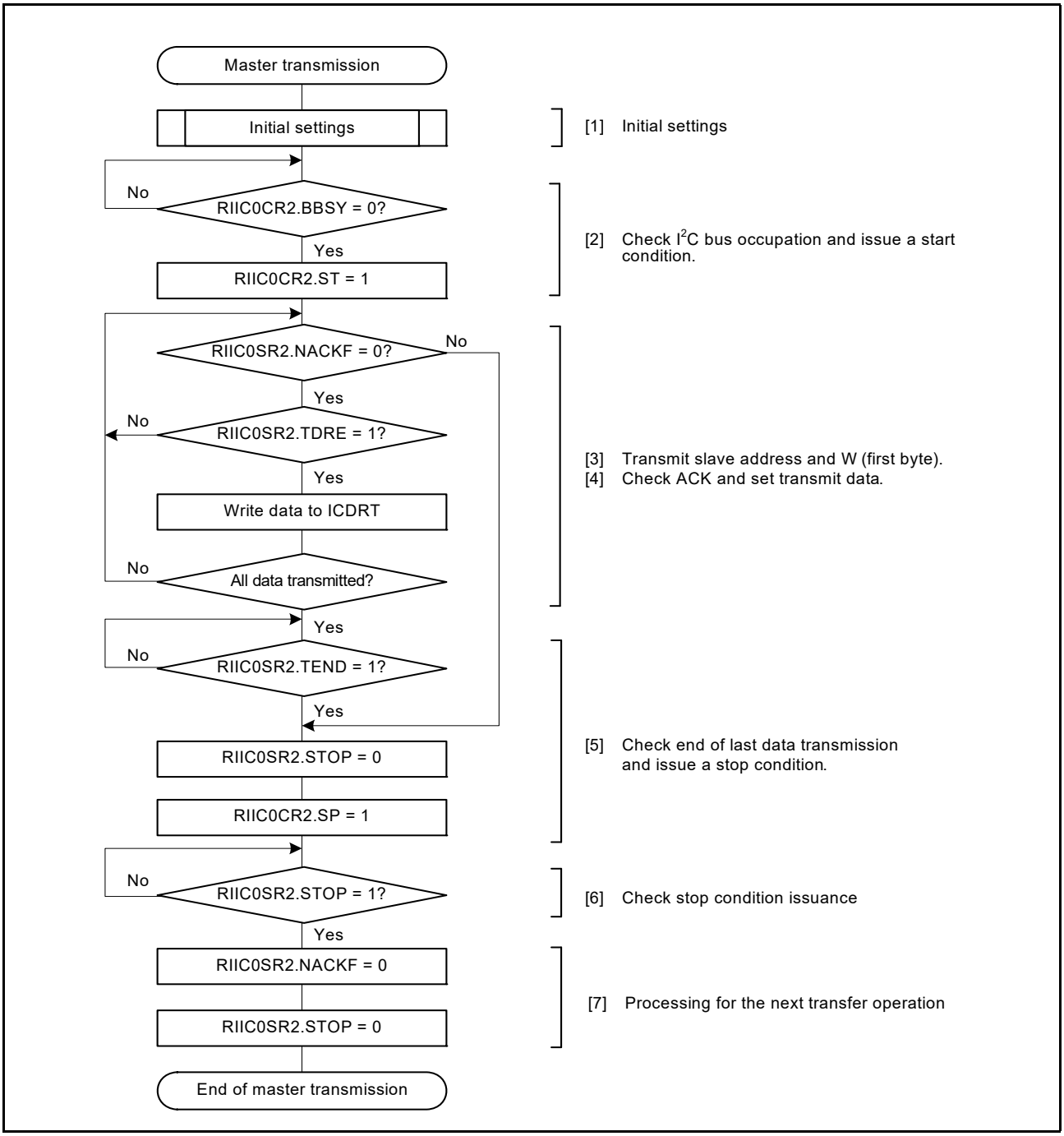


Figure 23.6 Example of Master Transmission Flowchart

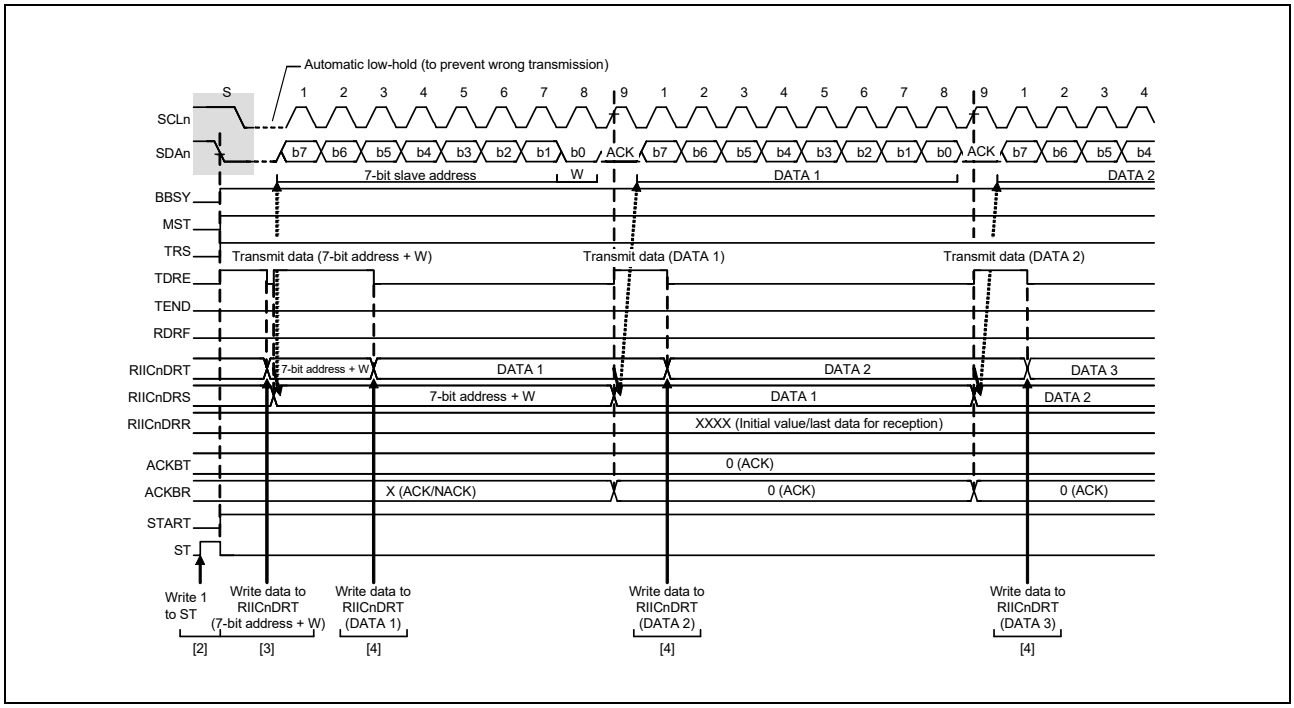


Figure 23.7 Master Transmit Operation Timing (1) (7-Bit Address Format)

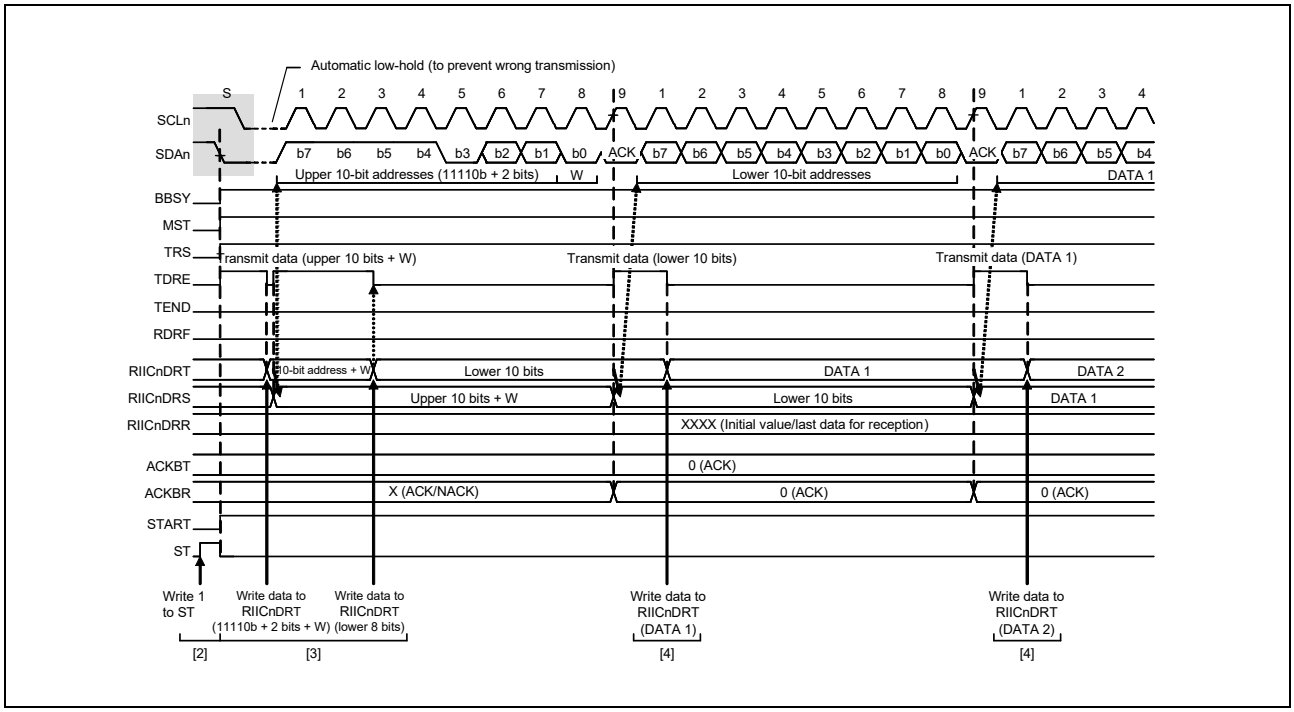


Figure 23.8 Master Transmit Operation Timing (2) (10-Bit Address Format)

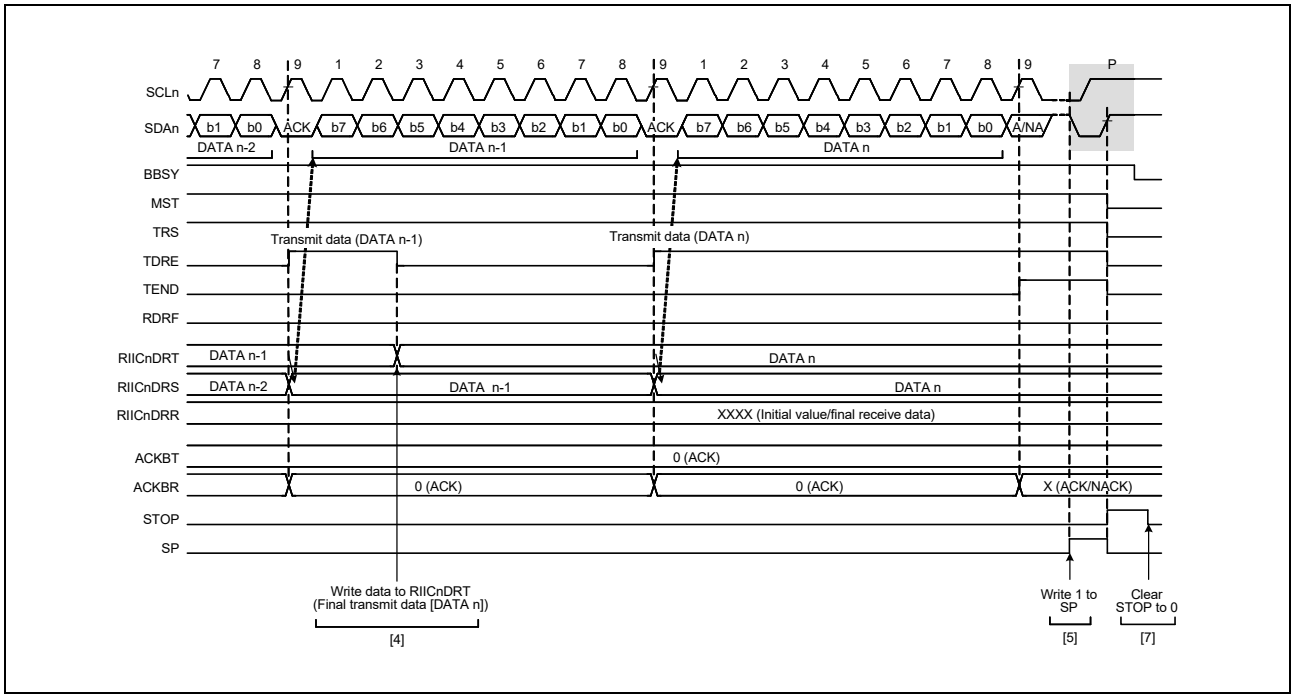


Figure 23.9 Master Transmit Operation Timing (3)

23.5.4 Master Receive Operation

In master receive operation, the RIIC as a master device outputs the SCL (clock) signal, receives data from the slave device, and returns acknowledgements. Since the RIIC must start by sending a slave address to the corresponding slave device, this part of the procedure is performed in master transmit mode, but the subsequent steps are in master receive mode.

Figure 23.10 shows an example of usage for the master reception of 3 or more bytes (7-bit address format), Figure 23.14 shows an example of usage for the master reception of 1 or 2 bytes (7-bit address format), and Figure 23.11 to Figure 23.13 show the timing of operations in master reception.

The following describes the procedure and operations for master reception.

(1) Set the RIICnCR1.IICRST bit to 1 (RIIC reset) and then set the RIICnCR1.ICE bit to 1 (internal reset) with the RIICnCR1.ICE bit cleared to 0 (RIICnSCL and RIICnSDA pins not driven). This initializes the internal state and the various flags of RIICnSR1. After that, set registers RIICnSARy, RIICnSER, RIICnMR1, RIICnBRH, and RIICnBRL (y = 0 to 2), and set the other registers as necessary (for initial settings of the RIIC, see Figure 23.5). When the necessary register settings have been completed, set the RIICnCR1.IICRST bit to 0 (for release from the reset state). This step is not necessary if initialization of the RIIC has already been completed.

(2) Read the RIICnCR2.BBSY flag to check that the bus is open, and then set the RIICnCR2.ST bit to 1 (start condition issuance request). Upon receiving the request, the RIIC issues a start condition. When the RIIC detects the start condition, the BBSY flag and the RIICnSR2.START flag are automatically set to 1 and the ST bit is automatically cleared to 0. At this time, if the start condition is detected and the levels for the SDA output and the levels on the SDA line have matched while the ST bit is 1, the RIIC recognizes that issuing of the start condition as requested by the ST bit has been successfully completed, and the RIICnCR2.MST and TRS bits are automatically set to 1, placing the RIIC in master transmit mode. The RIICnSR2.TDRE flag is also automatically set to 1 in response to setting of the TRS bit to 1.

(3) Check that the RIICnSR2.TDRE flag is 1, and then write the value for transmission (the first byte indicates the slave address and value of the R/W# bit) to RIICnDRT. Once the data for transmission are written to RIICnDRT, the TDRE flag is automatically cleared to 0, the data are transferred from RIICnDRT to RIICnDRS, and the TDRE flag is again set to 1. Once the byte containing the slave address and R/W# bit has been transmitted, the value of the RIICnCR2.TRS bit is automatically updated to select transmit or receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 1, the RIICnCR2.TRS bit is cleared to 0 on the rising edge of the ninth cycle of SCL (the clock signal), placing the RIIC in master receive mode. At this time, the TDRE flag is automatically cleared to 0 and the RIICnSR2.RDRF flag is automatically set to 1.
Since the RIICnSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the RIICnCR2.SP bit to issue a stop condition.
For master reception from a device with a 10-bit address, start by using master transmission to issue the 10-bit address, and then issue a restart condition. After that, transmitting 1111 0B, the two higher-order bits of the slave address, and the R bit places the RIIC in master receive mode.

(4) Dummy read RIICnDRR after confirming that the RIICnSR2.RDRF flag is 1; this makes the RIIC start output of the SCL (clock) signal and start data reception.

(5) After 1 byte of data has been received, the RIICnSR2.RDRF flag is set to 1 on the rising edge of the eighth or ninth cycle of SCL clock (the clock signal) as selected by the RIICnMR3.RDRFS bit. Reading out RIICnDRR at this time will produce the received data, and the RDRF flag is automatically cleared to 0 at the same time. Furthermore, the value of the acknowledgement field received during the ninth cycle of SCL clock is returned as the value set in the RIICnMR3.ACKBT bit. Furthermore, if the next byte to be received is the next to last byte, set the RIICnMR3.WAIT bit to 1 (for wait insertion) before reading the RIICnDRR (containing the second byte from last). As well as enabling NACK output even in the case of delays in processing to set the RIICnMR3.ACKBT bit to 1 (NACK) in step (6), due to other interrupts, etc., this fixes the SCL line to the low level on the rising edge of the ninth clock cycle in reception of the last byte, so the state is such that issuing a stop condition is possible.

(6) When the RIICnMR3.RDRFS bit is 0 and the slave device must be notified that it is to end transfer for data reception after transfer of the next (final) byte, set the RIICnMR3.ACKBT bit to 1 (NACK).

(7) After reading out the byte before last from the RIICnDRR register, if the value of the RIICnSR2.RDRF flag is confirmed to be 1, write 1 to the RIICnCR2.SP bit (stop condition issuance request) and then read the last byte from RIICnDRR. When RIICnDRR is read, the RIIC is released from the wait state and issues the stop condition after low-level output in the ninth clock cycle is completed or the SCL line is released from the low-hold state.

(8) Upon detecting the stop condition, the RIIC automatically clears the RIICnCR2.MST and TRS bits to 00B and enters slave receive mode. Furthermore, detection of the stop condition leads to setting of the RIICnSR2.STOP flag to 1.

(9) After checking that the RIICnSR2.STOP flag is 1, clear the RIICnSR2.NACKF and STOP flags to 0 for the next transfer operation.

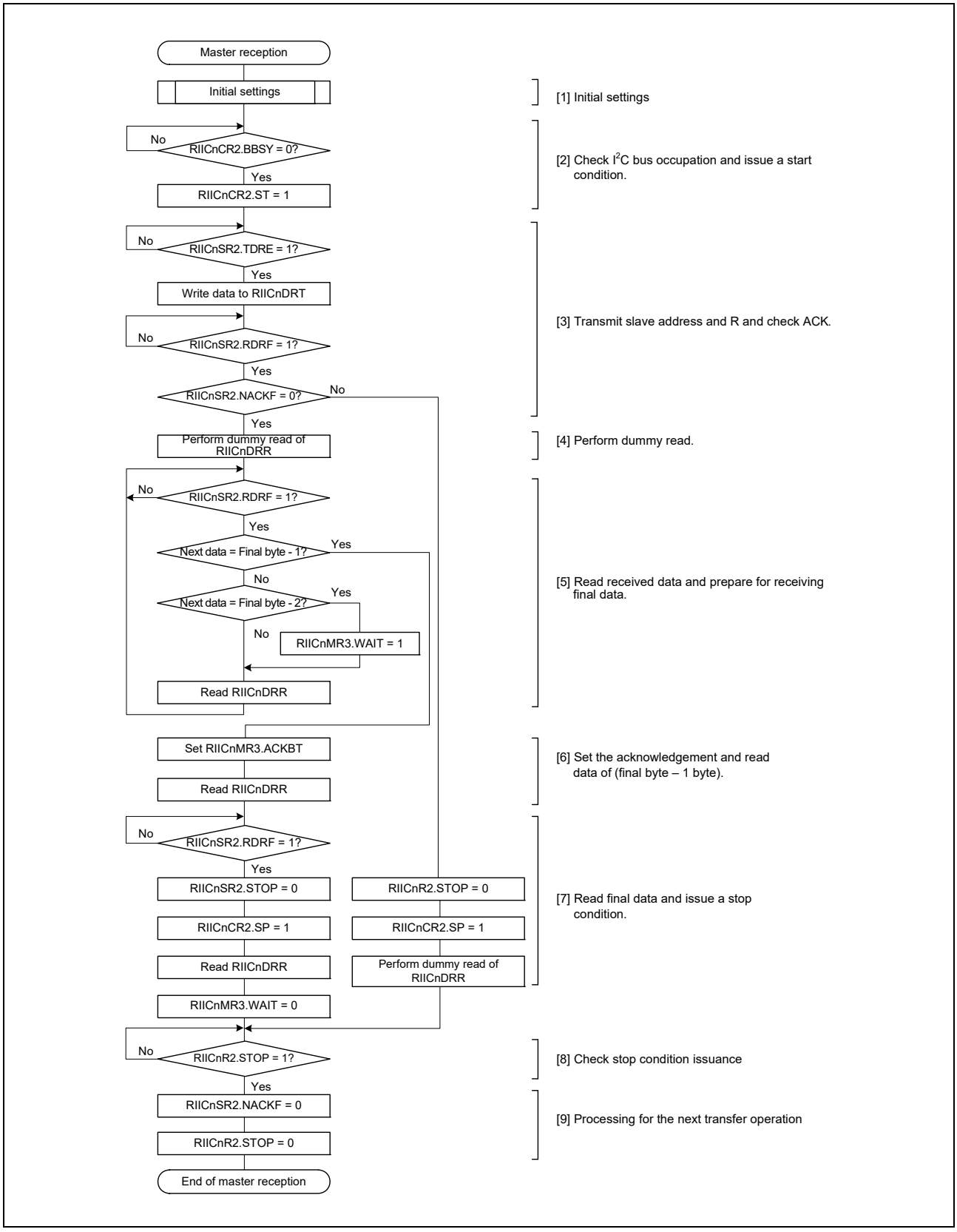


Figure 23.10 Example Flowchart for the Master Reception of 3 or More Bytes (7-Bit Address Format)

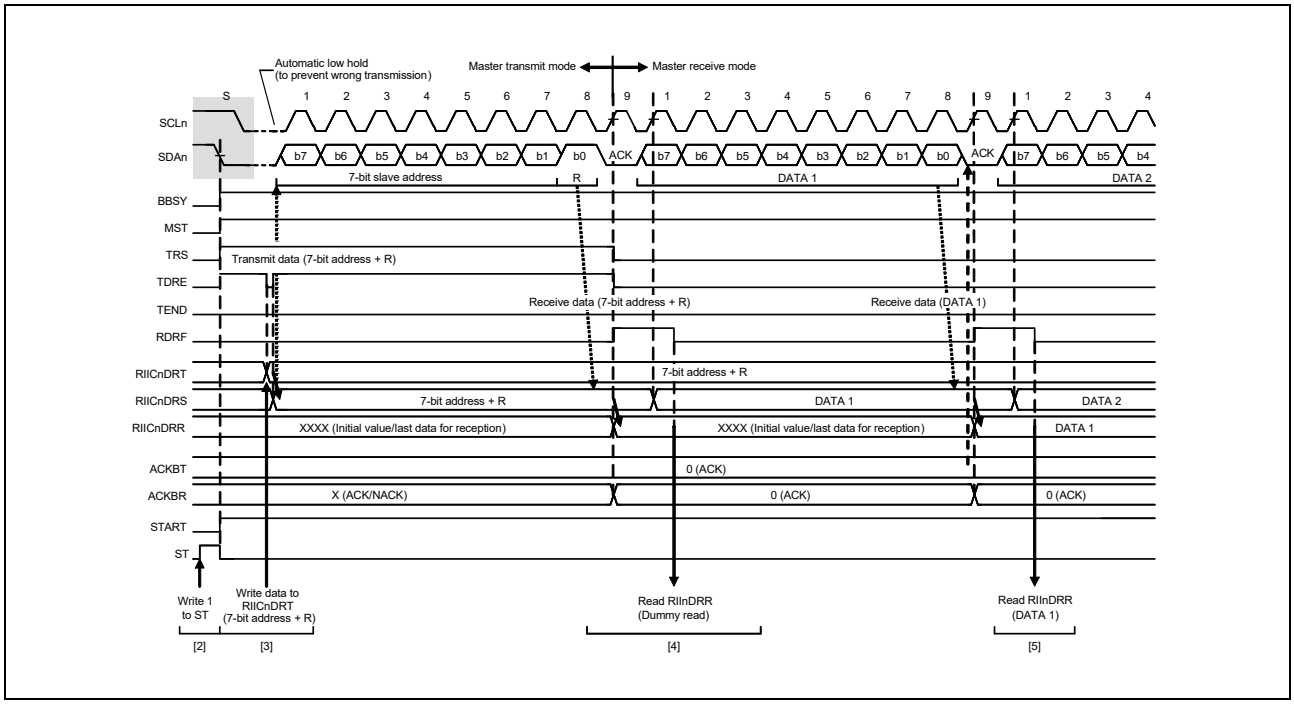


Figure 23.11 Master Receive Operation Timing (1) (7-Bit Address Format, when RDRFS = 0)

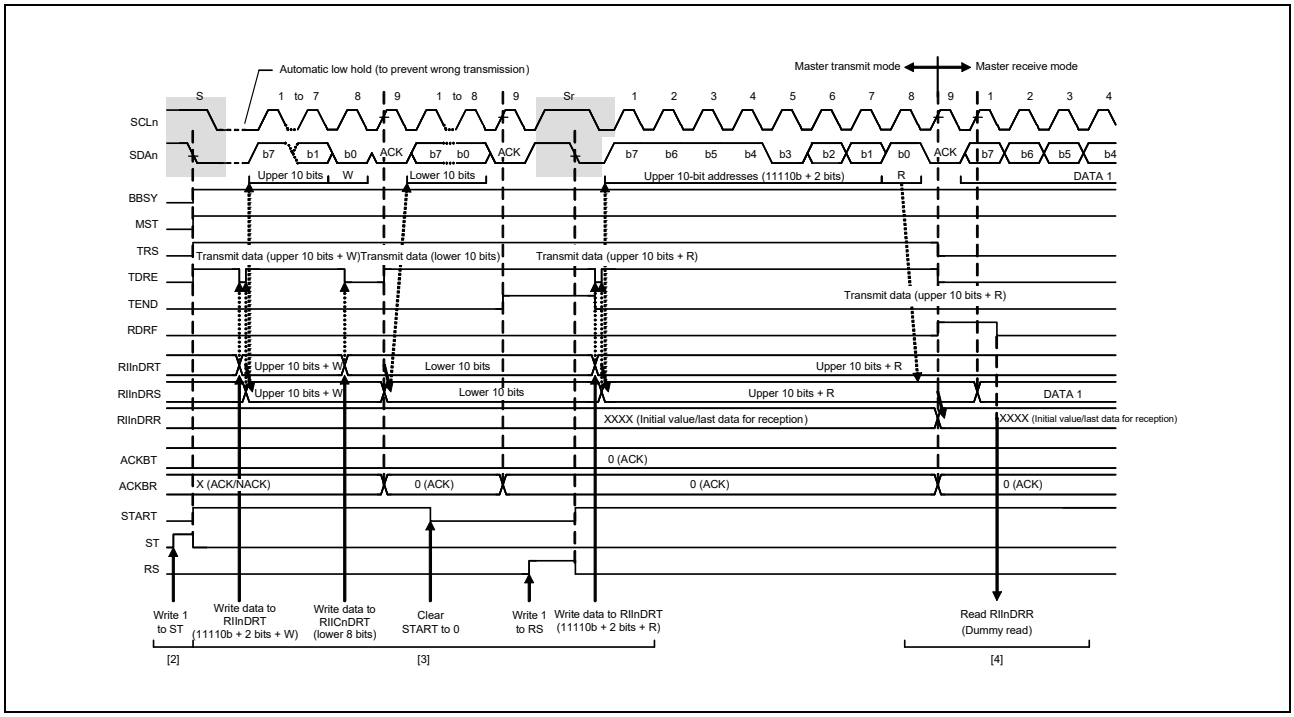


Figure 23.12 Master Receive Operation Timing (2) (10-Bit Address Format, when RDRFS = 0)

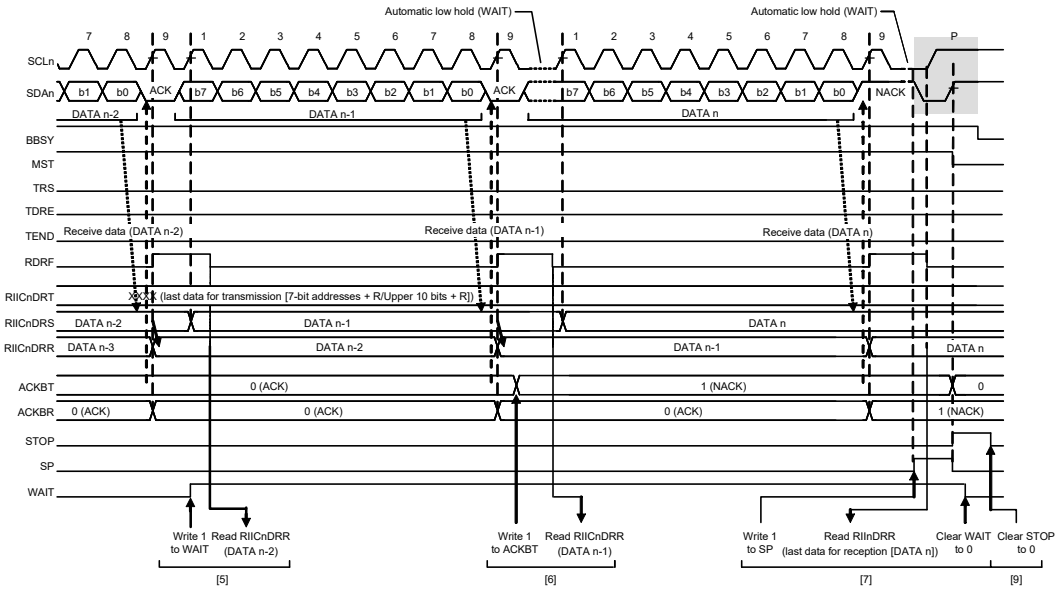


Figure 23.13 Master Receive Operation Timing (3) (when RDRFS = 0)

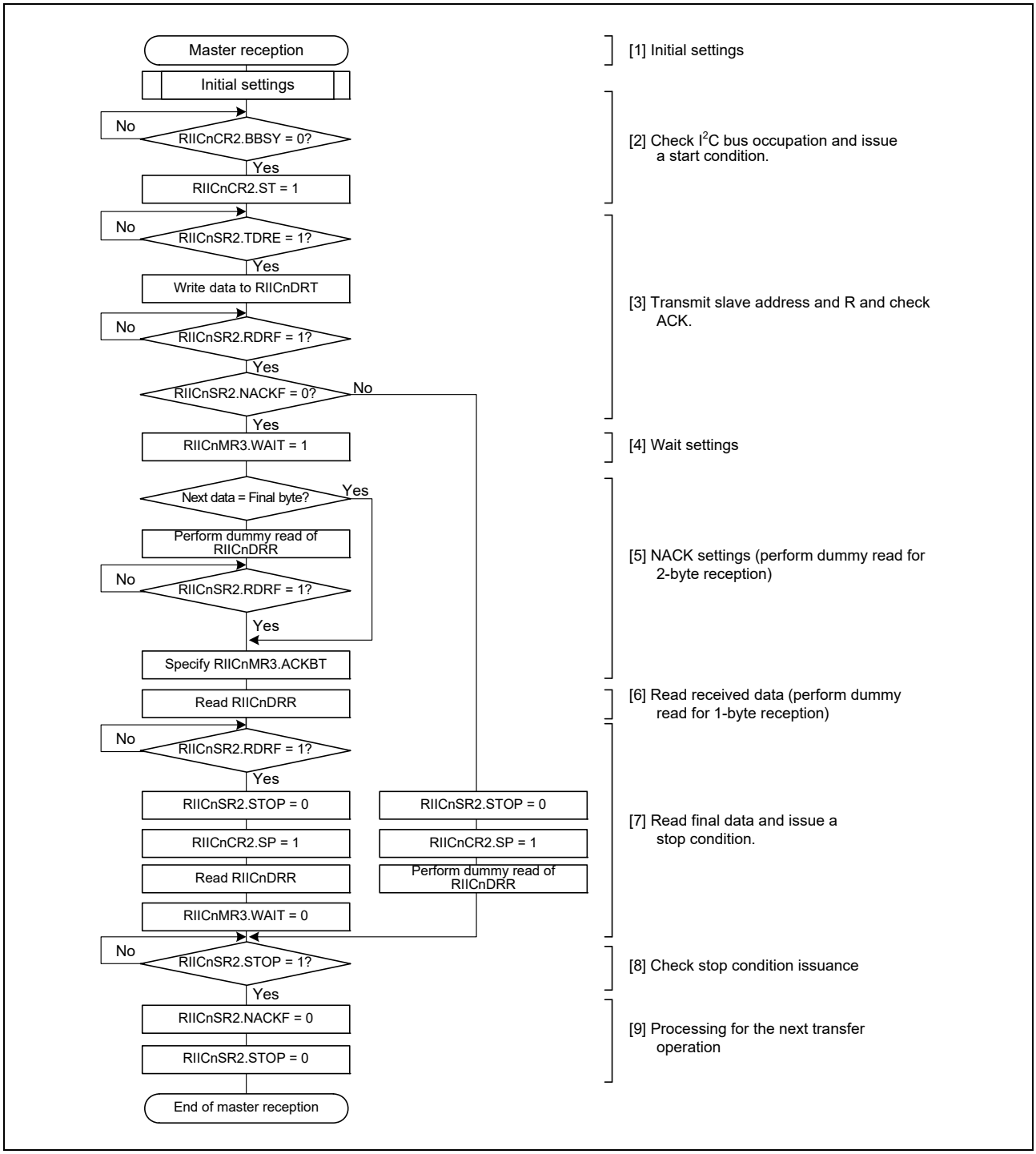


Figure 23.14 Example Flowchart for the Master Reception of 1 or 2 Bytes (7-Bit Address Format)

23.5.5 Slave Transmit Operation

In slave transmit operation, the master device outputs the SCL (clock) signal, the RIIC transmits data as a slave device, and the master device returns acknowledgements.

Figure 23.15 shows an example of usage of slave transmission and Figure 23.16 and Figure 23.17 show the timing of operations in slave transmission.

The following describes the procedure and operations for slave transmission.

(1) Follow the procedure in Figure 23.5 to make initial settings for the RIIC. This step is not necessary if initialization of the RIIC has already been completed. After initial settings, the RIIC will stay in the standby state until it receives a slave address that it matches.

(2) After receiving a matching slave address, the RIIC sets one of the corresponding bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the value set in the RIICnMR3.ACKBT bit to the acknowledge bit on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 1, the RIIC automatically places itself in slave transmit mode by setting both the RIICnCR2.TRS bit and the RIICnSR2.TDRE flag to 1.

(3) After the RIICnSR2.TEND flag is confirmed to be 1, write the data for transmission to the RIICnDRT register. At this time, if the RIIC receives no acknowledge from the master device (receives an NACK signal) while the RIICnFER.NACKE bit is 1, the RIIC suspends transfer of the next data.

(4) Wait unit the RIICnSR2.TEND flag is set to 1 while the RIICnSR2.TDRE flag is 1, after the RIICnSR2.NACKF flag is set to 1 or the last byte for transmission is written to the RIICnDRT register. When the RIICnSR2.NACKF flag or the TEND flag is 1, the RIIC drives the SCL line low on the ninth falling edge of SCL clock.

(5) When the RIICnSR2.NACKF flag or the RIICnSR2.TEND flag is 1, dummy read RIICnDRR to complete the processing. This releases the SCL line.

(6) Upon detecting the stop condition, the RIIC automatically clears bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2), flags RIICnSR2.TDRE and TEND, and the RIICnCR2.TRS bit to 0, and enters slave receive mode.

(7) Clear the RIICnSR2.NACKF and STOP flags to 0.

Caution: Operations for transfer start if the RIICnSR2.NACKF flag is cleared to 0 before RIICnSR2.STOP is set to 1. Be sure to confirm that RIICnSR2.STOP is set to 1 before clearing RIICnSR2.NACKF to 0. In particular, when the NACK receive interrupt (INTRIICNAKI) is in use, take care not to clear the NACKF flag to 0 before the STOP flag is set to 1 during interrupt processing.

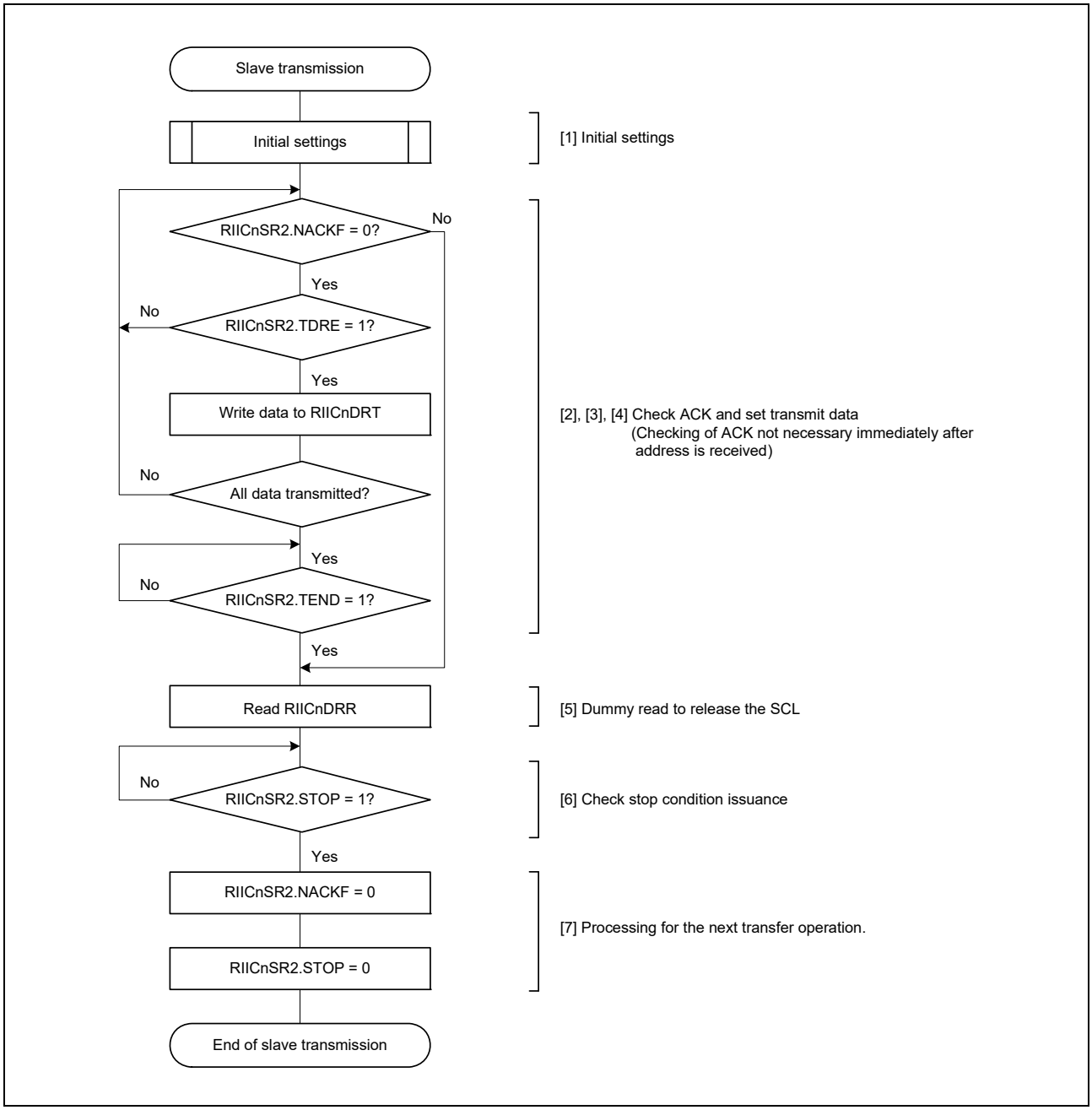


Figure 23.15 Example of Slave Transmission Flowchart

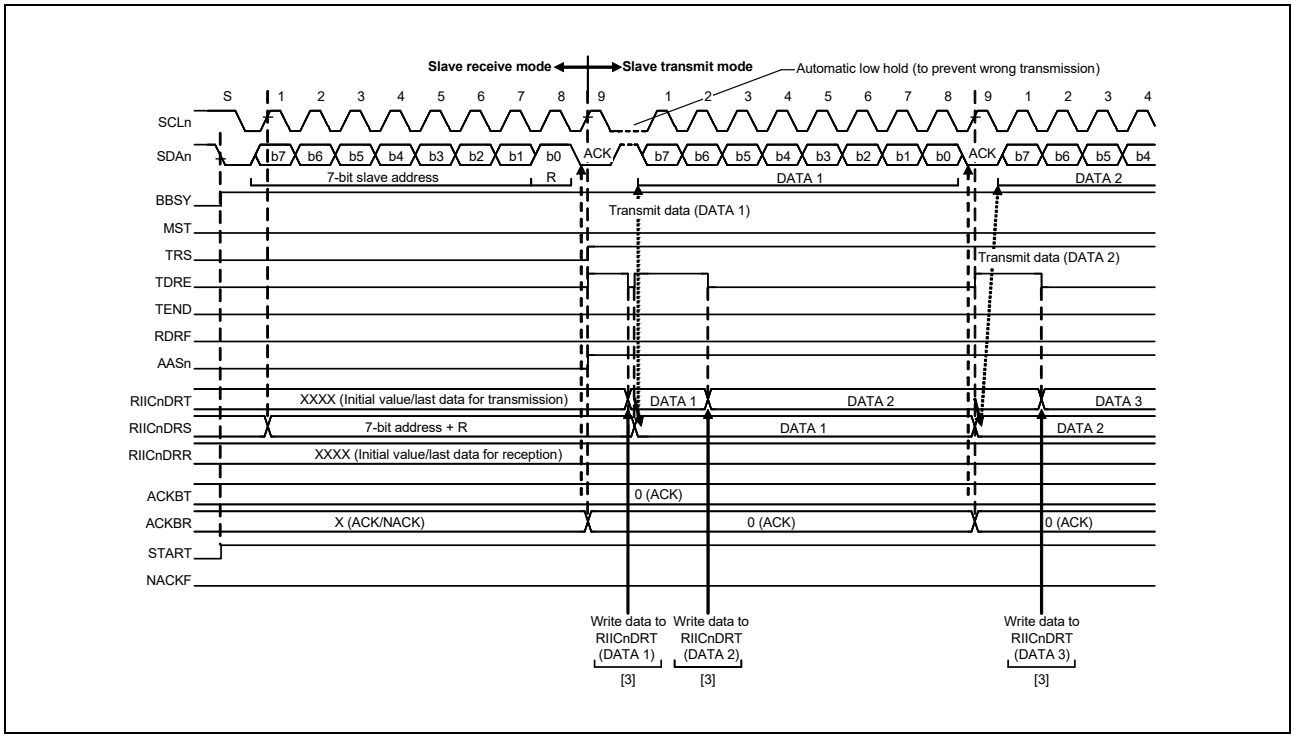


Figure 23.16 Slave Transmit Operation Timing (1) (7-Bit Address Format)

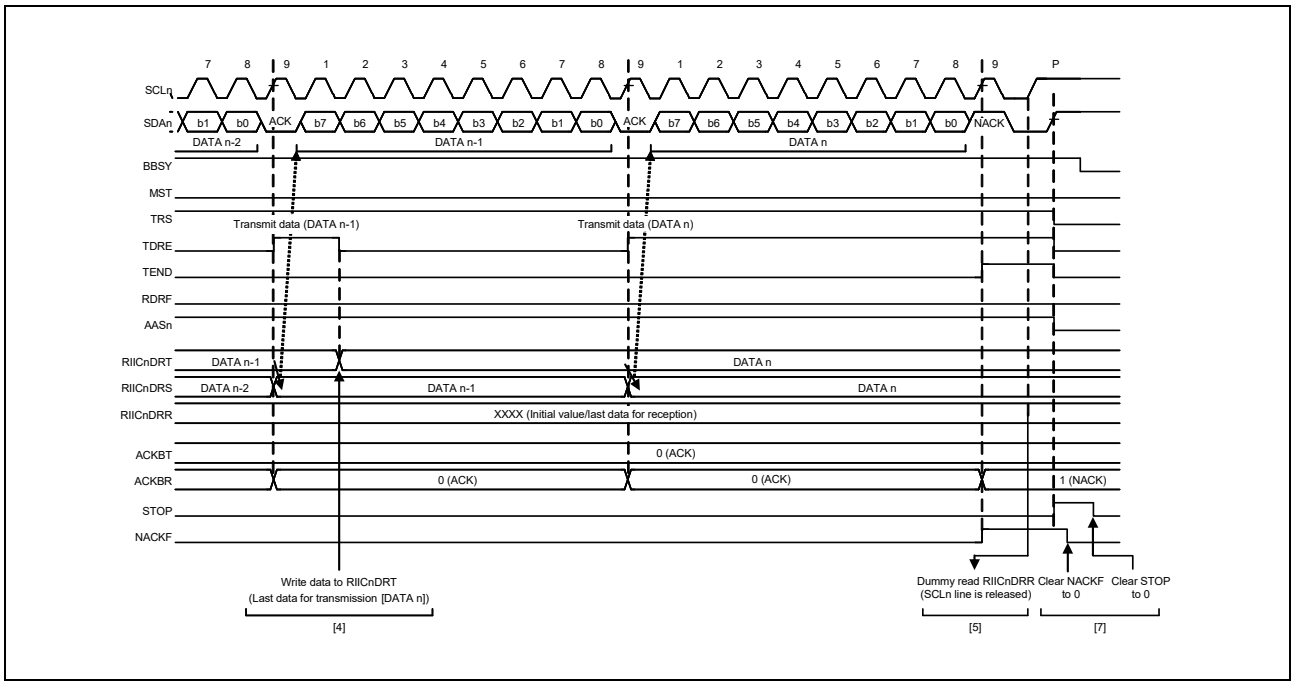


Figure 23.17 Slave Transmit Operation Timing (2)

23.5.6 Slave Receive Operation

In slave receive operation, the master device outputs the SCL clock and transmit data, and the RIIC returns acknowledgements as a slave device.
Figure 23.18 shows an example of usage of slave reception and Figure 23.19 and Figure 23.20 show the timing of operations in slave reception.
The following describes the procedure and operations for slave reception.

(1) Follow the procedure in Figure 23.5 to make initial settings for the RIIC. This step is not necessary if initialization of the RIIC has already been completed. After initial settings, the RIIC will stay in the standby state until it receives a slave address that it matches.
(2) After receiving a matching slave address, the RIIC sets one of the corresponding bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the value set in the RIICnMR3.ACKBT bit to the acknowledge bit on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 0, the RIIC continues to place itself in slave receive mode and sets the RIICnSR2.RDRF flag to 1.
(3) After the RIICnSR2.STOP flag is confirmed to be 0 and the RIICnSR2.RDRF flag to be 1, dummy read RIICnDRR (the dummy value consists of the slave address and R/W# bit when the 7-bit address format is selected, or the lower eight bits when the 10-bit address format is selected).
(4) When RIICnDRR is read, the RIIC automatically clears the RIICnSR2.RDRF flag to 0. If reading of RIICnDRR is delayed and a next byte is received while the RDRF flag is still set to 1, the RIIC holds the SCL line low from one SCL cycle before the timing with which RDRF should be set. In this case, reading RIICnDRR releases the SCL line from being held at the low level.
When the RIICnSR2.STOP flag is 1 and the RIICnSR2.RDRF flag is also 1, read RIICnDRR until all the data is completely received.
(5) Upon detecting the stop condition, the RIIC automatically clears bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2) to 0.
(6) After checking that the RIICnSR2.STOP flag is 1, clear the RIICnSR2.STOP flag to 0 for the next transfer operation.

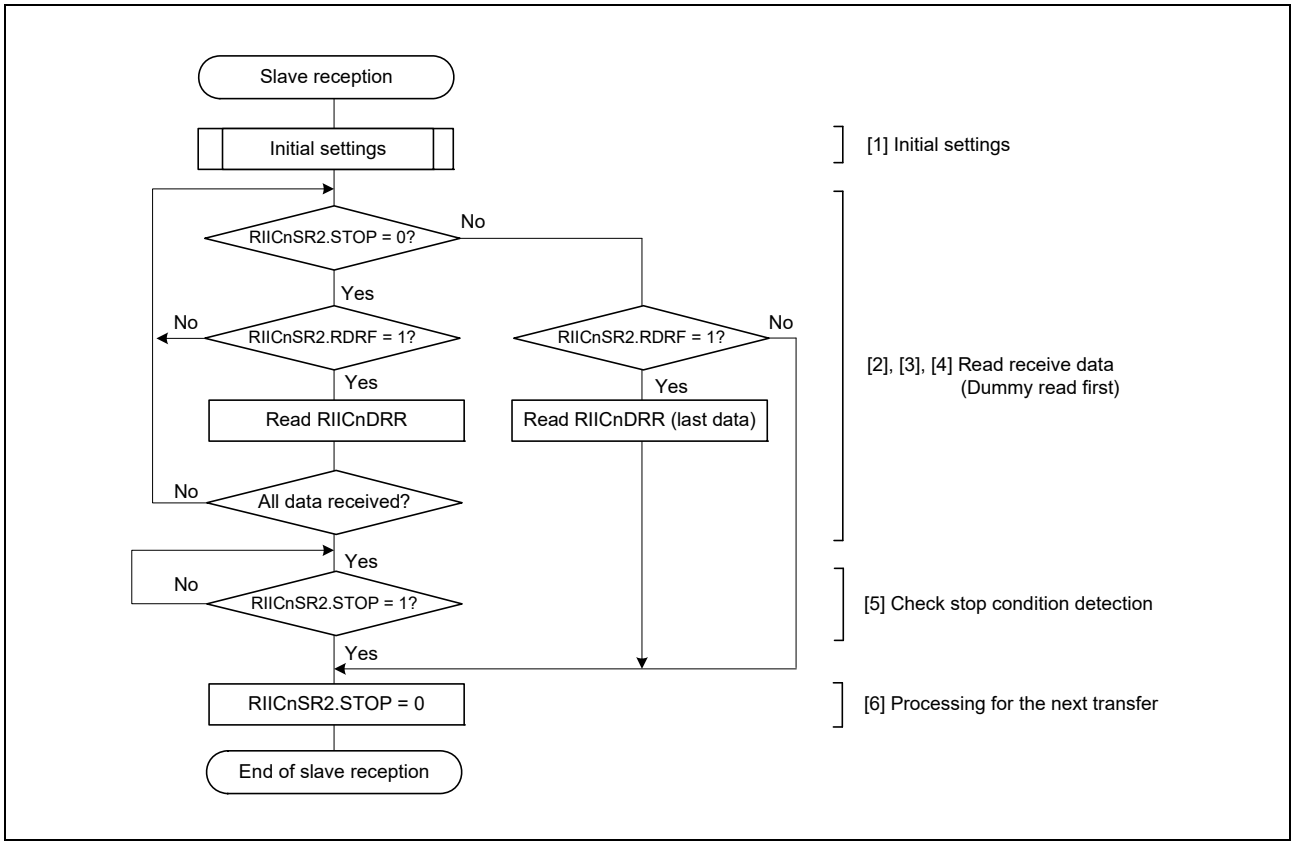


Figure 23.18 Example of Slave Reception Flowchart

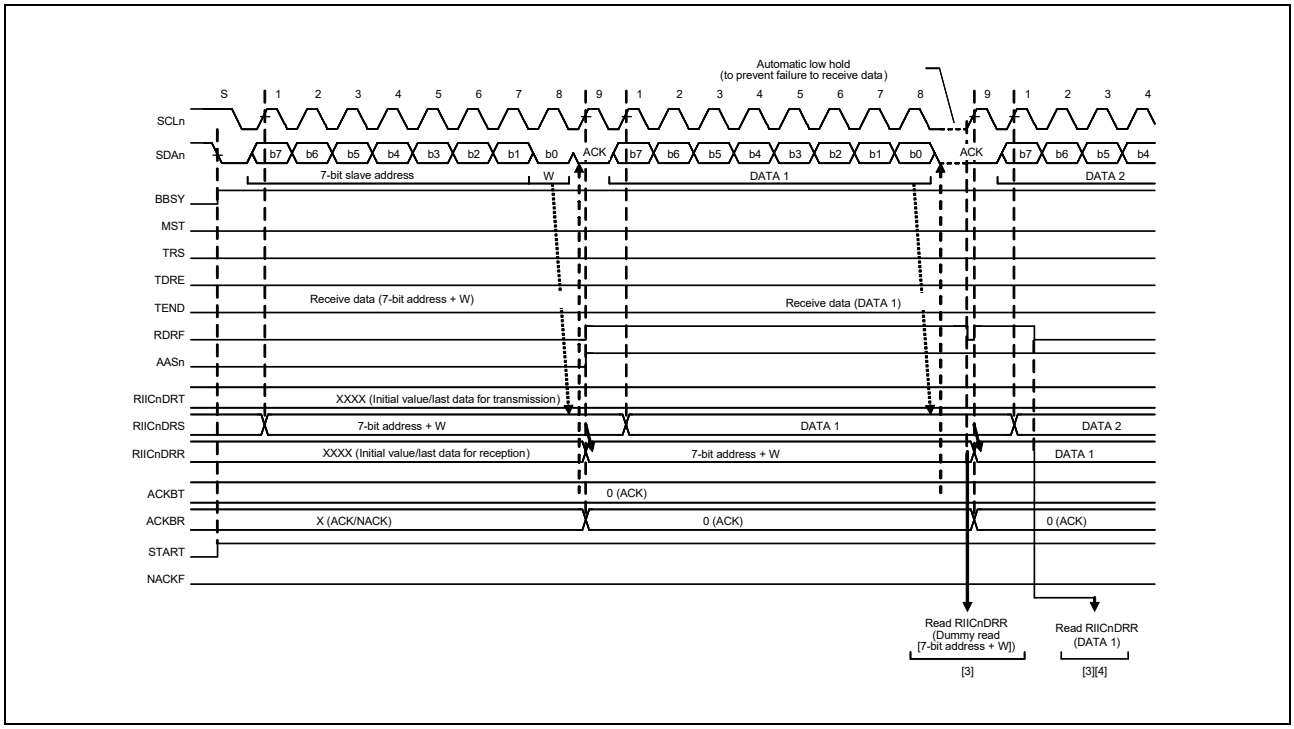


Figure 23.19 Slave Receive Operation Timing (1) (7-Bit Address Format, when RDRFS = 0)

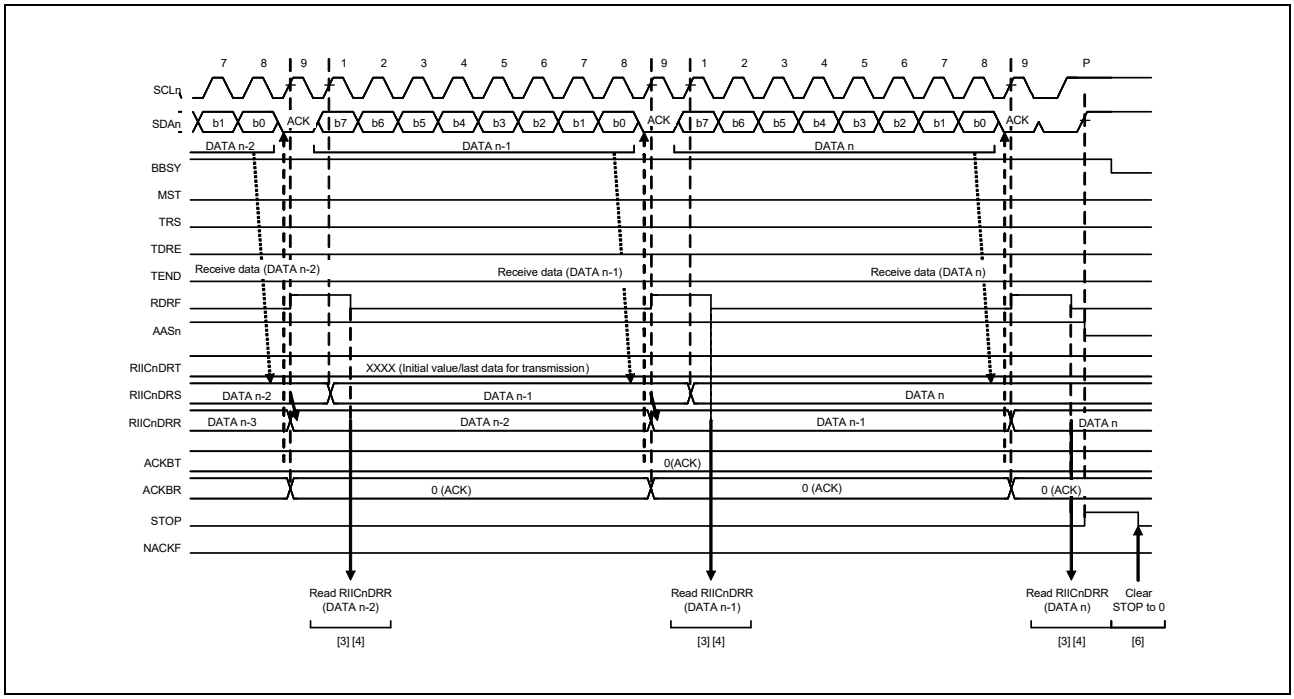


Figure 23.20 Slave Receive Operation Timing (2) (when RDRFS = 0)

23.6 SCL Synchronization Circuit

In generation of the SCL (clock) signal, the RIIC starts counting out the value for width at high level specified in RIICnBRH when it detects a rising edge on the SCL line and drives the SCL line low once counting of the width at high level is complete. When the RIIC detects the falling edge of the SCL line, it starts counting out the width at low level period specified in RIICnBRL, and then stops driving the SCL line (releases the line) once counting of the width at low level is complete. The SCL (clock) signal is thus generated.

If multiple master devices are connected to the I²C bus, a collision of SCL signals may arise due to contention with another master device. In such cases, the master devices have to synchronize their SCL signals. Since this synchronization of SCL signals must be bit by bit, the RIIC is equipped with a facility (the SCL synchronization circuit) to obtain bit-by-bit synchronization of the SCL clock signals by monitoring the SCL line during communication.

When the RIIC has detected a rising edge on the SCL line and thus started counting out the width at high level specified in RIICnBRH, and the level on the SCL line falls because an SCL signal is being generated by another master device, the RIIC stops counting when it detects the falling edge, drives the level on the SCL line low, and starts counting out the width at low level specified in RIICnBRL. When the RIIC finishes counting out the width at low level, it stops driving the SCL line to the low level (i.e. releases the line). At this time, if the width at low level of the SCL clock signal from the other master device is longer than the width at low level set in the RIIC, the width at low level of the SCL signal will be extended. Once the width at low level for the other master device has ended, the SCL signal rises because the SCL line has been released. When the RIIC finishes outputting the low-level period of the SCL clock, the SCL line is released and the SCL clock rises. That is, in cases of contention of SCL signals from more than one master, the width at high level of the SCL signal is synchronized with that of the clock having the narrower width, and the width at low level of the SCL signal is synchronized with that of the clock having the broader width. However, such synchronization of the SCL signal is only enabled when the RIICnFER.SCLE bit is set to 1.

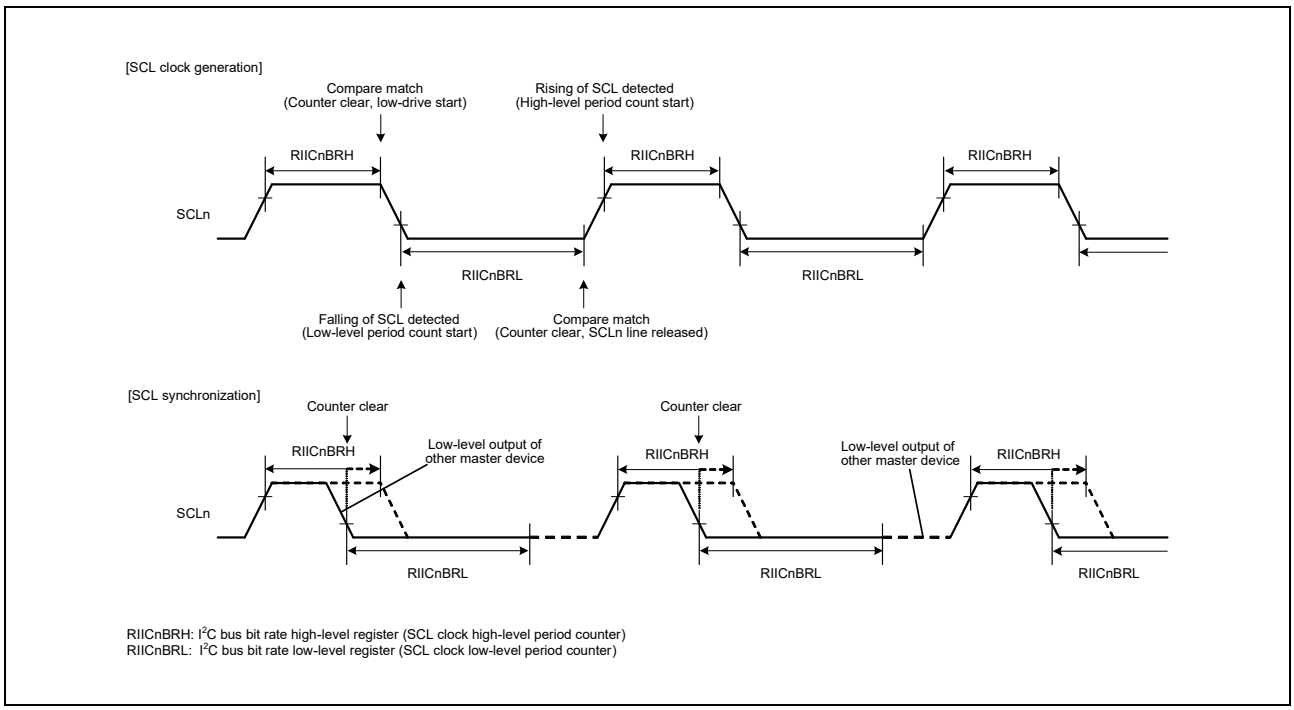


Figure 23.21 Generation and Synchronization of the SCL Signal from the RIIC

23.7 Facility for Delaying SDA Output

The RIIC module incorporates a facility for delaying output on the SDA line. The delay can be applied to all output (issuing of the start, restart, and stop conditions, data, and the ACK and NACK signals) on the SDA line.

With the SDA output delay facility, SDA output is delayed from detection of a falling edge of the SCL signal to ensure that the SDA signal is output within the interval over which the SCL (clock) signal is at the low level. Doing this leads to usage with the aim of preventing erroneous operation of communications devices, with the aim of satisfying the 300-ns (min.) data-hold time requirement of the SMBus specification.

The output delay facility is enabled by setting the RIICnMR2.SDDL[2:0] bits to any value other than 000_B, and disabled by setting the same bits to 000_B.

While the SDA output delay facility is enabled (i.e. while the SDDL[2:0] bits in IMCR2 are set to any value other than 000_B), the RIICnMR2.DLCS bit selects the clock source for counting by the SDA output delay counter as the internal base clock (IICφ) for the RIIC module or as a clock signal derived by dividing the frequency of the internal base clock by two (IICφ/2). The counter counts the number of cycles set in the SDDL[2:0] bits in IMCR2. After counting of the set number of cycles of delay is completed, the RIIC module places the required output (start, restart, or stop condition, data, or an ACK or NACK signal) on the SDA line.

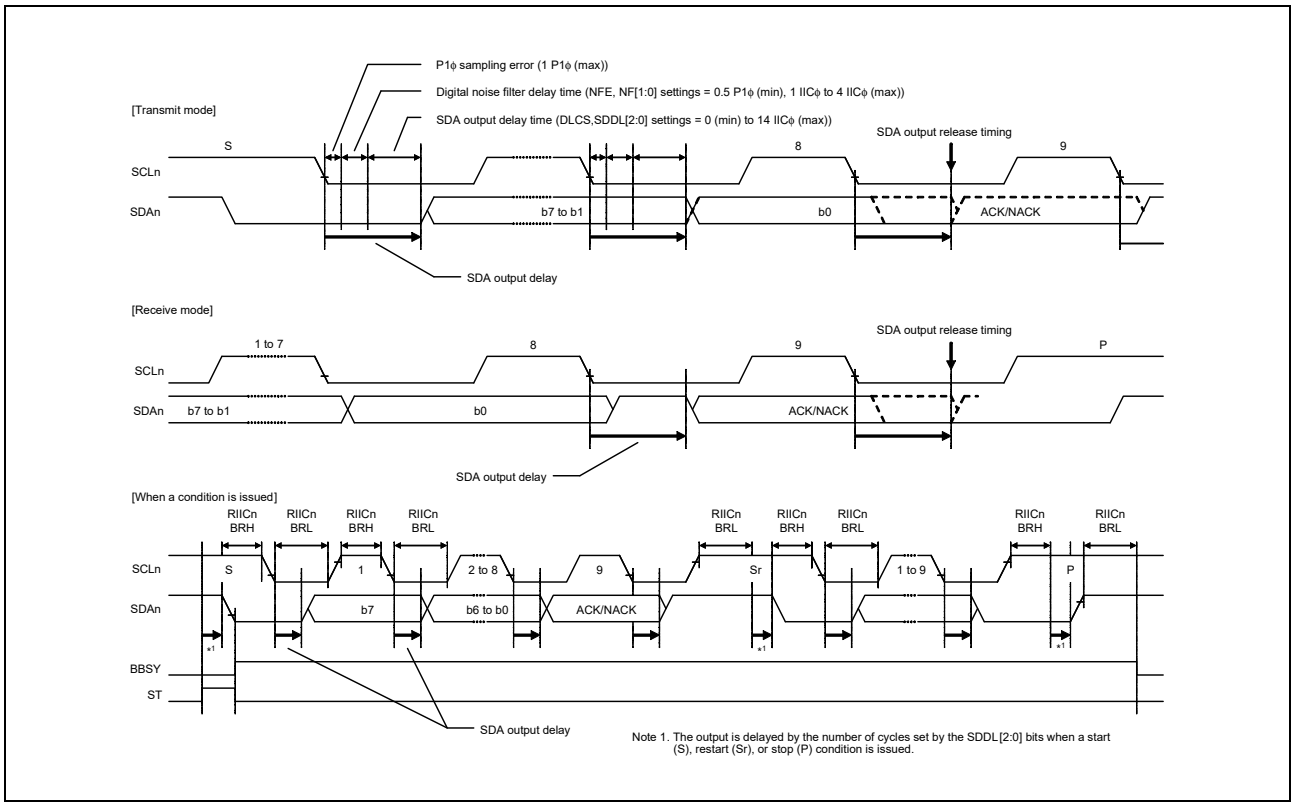


Figure 23.22 SDA Output Delay Facility

23.8 Digital Noise-Filter Circuits

Figure 23.23 is a block diagram of the digital noise-filter circuit. When the NFE bit in the RIICnFER register is set to 1, input to the RIICnSCL and RIICnSDA pins are conveyed to the internal circuitry through digital noise-filter circuits.

The on-chip digital noise-filter circuit of the RIIC consists of four flip-flop circuit stages connected in series and a match-detection circuit.

The number of effective stages in the digital noise filter is selected by the RIICnMR3.NF[1:0] bits. The selected number of effective stages determines the noise-filtering capability as a period from one to four IICφ cycles.

The input signal to the RIICnSCL pin (or RIICnSDA pin) is sampled on falling edges of the IICφ signal. When the input signal level matches the output level of the number of effective flip-flop circuit stages as selected by the RIICnMR3.NF[1:0] bits, the signal level is conveyed to the subsequent stage. If the signal levels do not match, the previous value is retained.

If the ratio between P1φ and IICφ is small when the RIICnMR1.CKS[2:0] bits are set to 000_B, note that the characteristics of the digital noise filter may lead to the elimination of needed signals as noise.

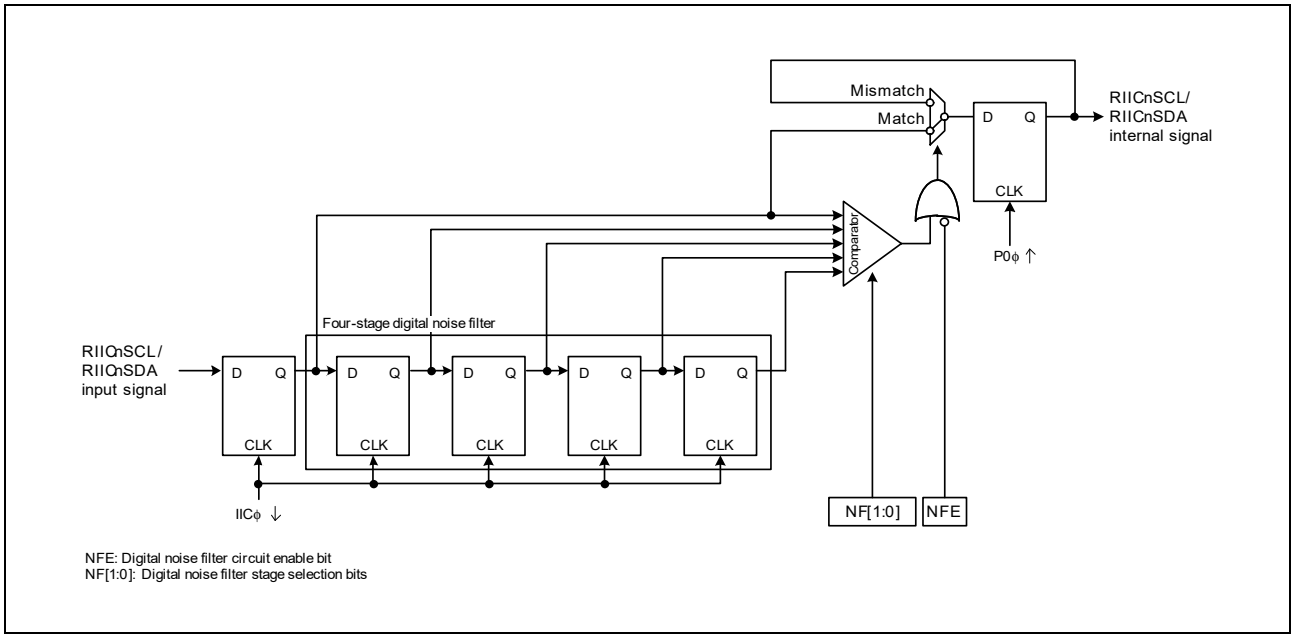


Figure 23.23 Block Diagram of Digital Noise Filter Circuit

23.9 Address Match Detection

The RIIC can set three unique slave addresses in addition to the general call address and host address, and also can set 7-bit or 10-bit slave addresses.

23.9.1 Slave-Address Match Detection

The RIIC can set three unique slave addresses, and has a slave address detection function for each unique slave address. When the RIICnSER.SARy bit (y = 0 to 2) is set to 1, the slave addresses set in RIICnSARy (y = 0 to 2) can be detected. When the RIIC detects a match of the set slave address, the corresponding RIICnSR1.AASy flag (y = 0 to 2) is set to 1 at the rising edge of the ninth SCL clock cycle, and the RIICnSR2.RDRF flag or the RIICnSR2.TDRE flag is set to 1 by the following R/W# bit. This causes a receive data full interrupt (INTRIICRI) or transmit data empty interrupt (INTRIICTI) to be generated. The AASy flag is used to identify which slave address has been specified.
Figure 23.24 to Figure 23.26 show the AASy flag set timing in three cases.

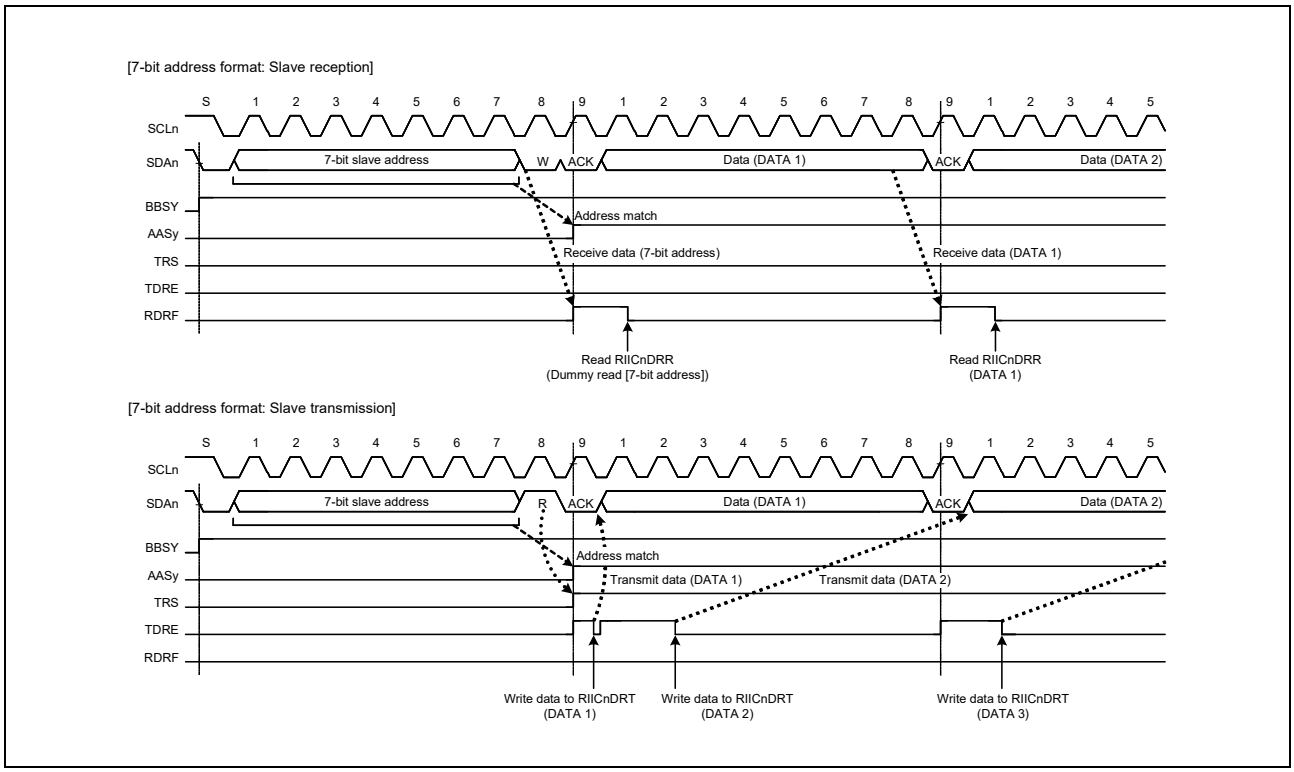


Figure 23.24 AASy Flag Set Timing with 7-Bit Address Format Selected

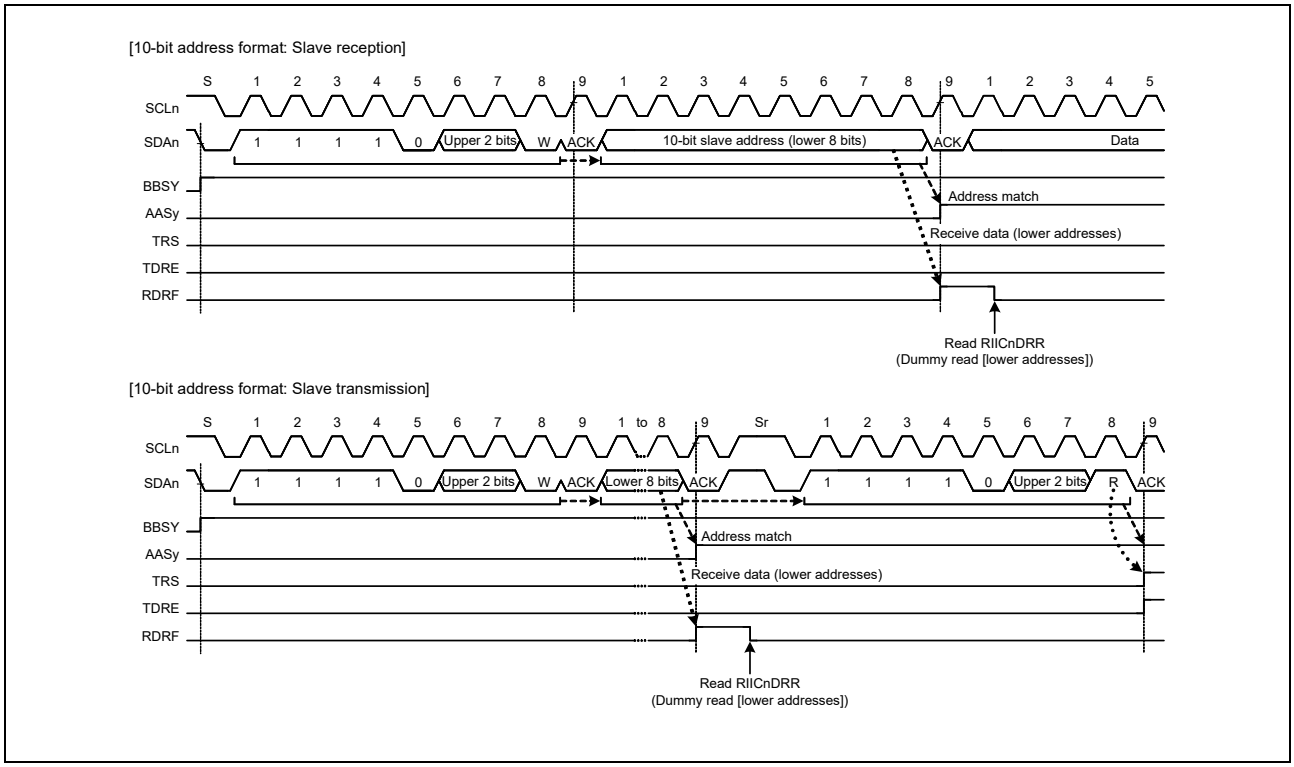


Figure 23.25 AASy Flag Set Timing with 10-Bit Address Format Selected

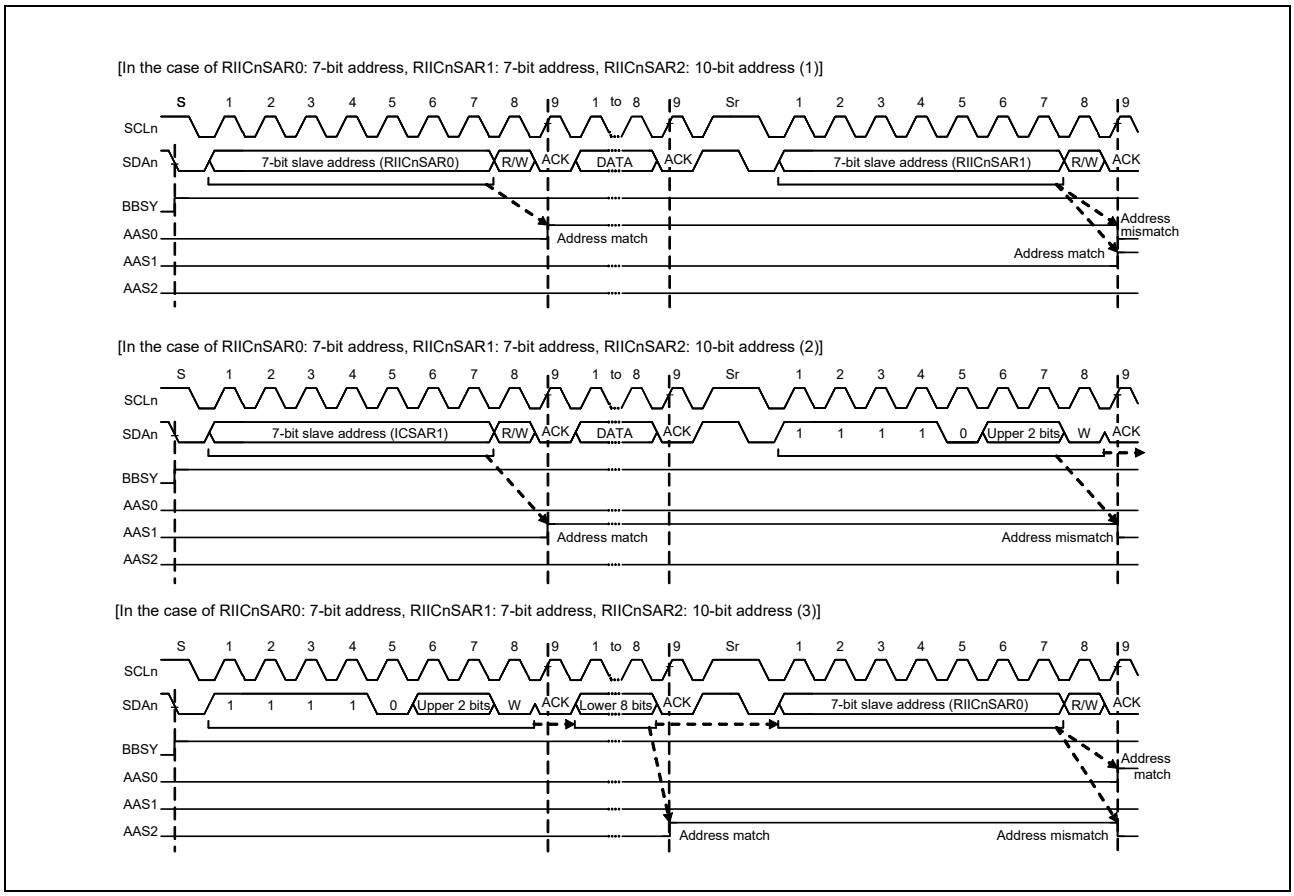


Figure 23.26 AASy Flag Set/Clear Timing with 7-Bit/10-Bit Address Formats Mixed

23.9.2 Detection of the General Call Address

The RIIC has a facility for detecting the general call address (0000 000_B + 0 [W]). This is enabled by setting the RIICnSER.GCE bit to 1.

If the address received after a start or restart condition is issued is 0000 000_B + 1[R] (start byte), the RIIC recognizes this as the address of a slave device with an “all-zero” address but not as the general call address.

When the RIIC detects the general call address, both the RIICnSR1.GCA flag and the RIICnSR2.RDRF flag are set to 1 on the rising edge of the ninth cycle of SCL clock. This leads to the generation of a receive data full interrupt (INTRIICRI). The value of the GCA flag can be confirmed to recognize that the general call address has been transmitted.

Operation after detection of the general call address is the same as normal slave receive operation.

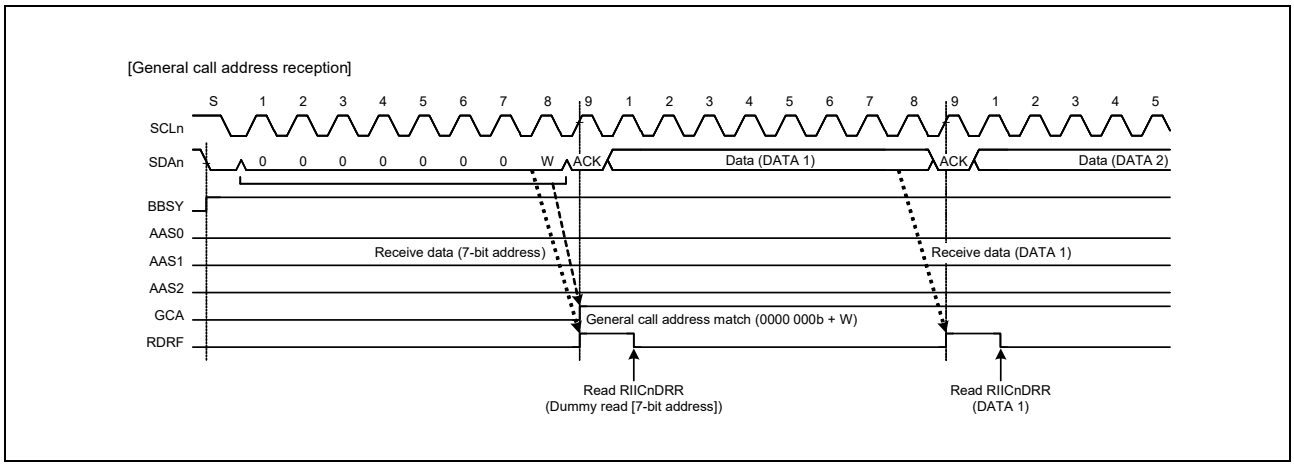


Figure 23.27 Timing of GCA Flag Setting during Reception of General Call Address

23.9.3 Device-ID Address Detection

The RIIC module has a facility for detecting device-ID addresses conformant with the I²C bus specification (Rev. 03). When the RIIC receives 1111 100_B as the first byte after a start condition or restart condition was issued with the RIICnSER.DIDE bit set to 1, the RIIC recognizes the address as a device ID, sets the RIICnSR1.DID flag to 1 on the rising edge of the ninth SCL clock cycle when the following R/W# bit is 0, and then compares the second and subsequent bytes with its own slave address. If the address matches the value in the slave address register, the RIIC sets the corresponding RIICnSR1.AASy flag (y = 0 to 2) to 1.

After that, when the first byte received after a start or restart condition is issued matches the device ID address (1111 100_B) again and the following R/W# bit is 1, the RIIC does not compare the second and subsequent bytes and sets the RIICnSR2.TDRE flag to 1.

In the device-ID address detection function, the RIIC clears the DID flag to 0 if a match with the RIIC's own slave address is not obtained or a match with the device ID address is not obtained after a match with the RIIC's own slave address and the detection of a restart condition. If the first byte after detection of a start or restart condition matches the device ID address (1111 100_B) and the R/W# bit is 0, the RIIC sets the DID flag to 1 and compares the second and subsequent bytes with the RIIC's slave address. If the R/W# bit is 1, the DID flag holds the previous value and the RIIC does not compare the second and subsequent bytes. Therefore, the reception of a device-ID address can be checked by reading the DID flag after confirming that TDRE = 1.

Furthermore, prepare the device-ID fields (three bytes: 12 bits indicating the manufacturer + 9 bits identifying the part + 3 bits indicating the revision) that must be sent to the host after reception of a continuous device-ID field as normal data for transmission. For details, see I²C Bus Standard from NXP Semiconductors.

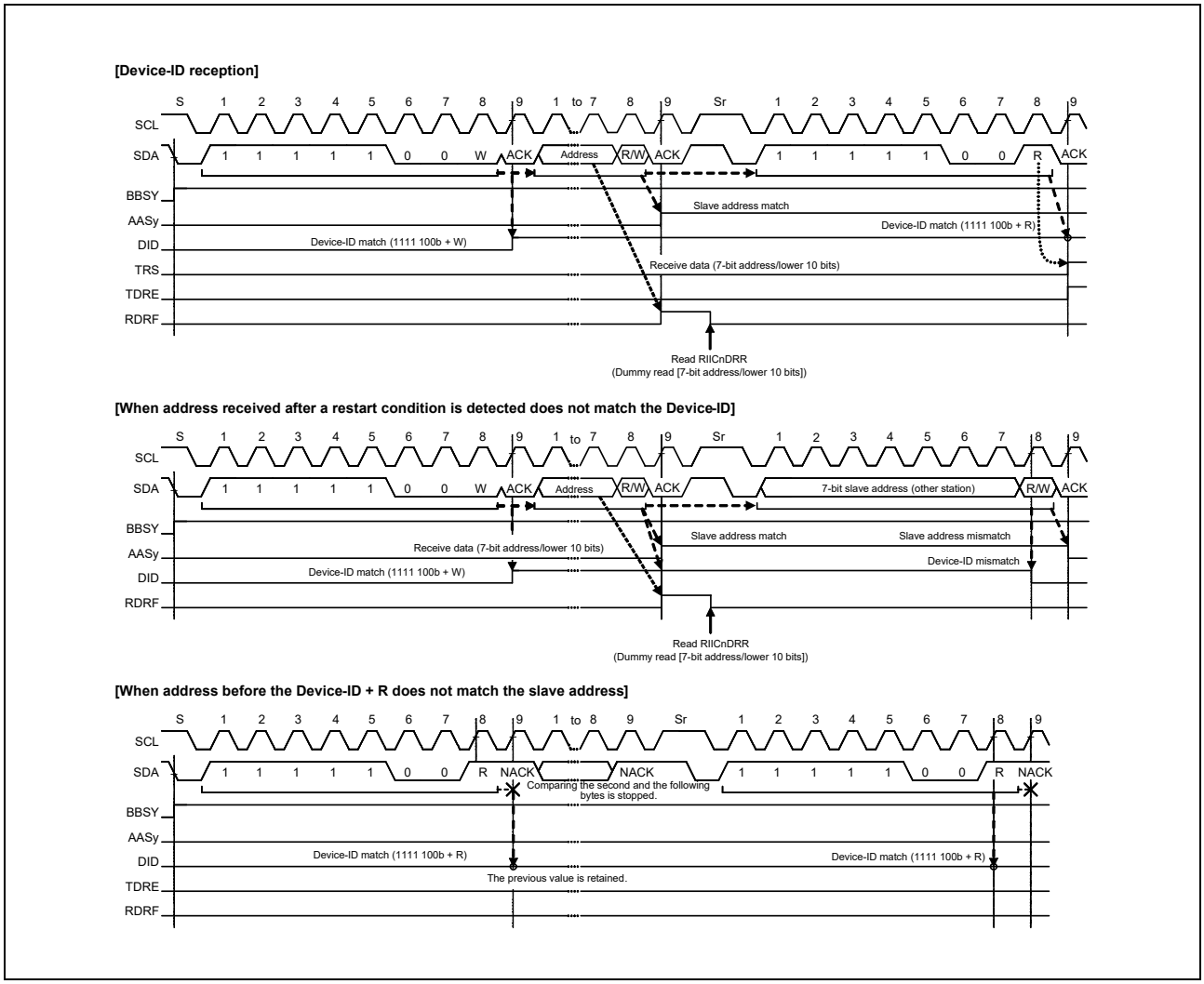


Figure 23.28 AASy/DID Flag Set/Clear Timing during Reception of Device-ID

23.9.4 Host Address Detection

The RIIC has a function to detect the host address while the SMBus is operating. When the RIICnSER.HOAE bit is set to 1 while the RIICnMR3.SMBS bit is 1, the RIIC can detect the host address ($0001\ 000_B$) in slave receive mode (RIICnCR2.MST and TRS bits = 00_B).

When the RIIC detects the host address, the RIICnSR1.HOA flag is set to 1 at the rising edge of the ninth SCL clock cycle, and at the same time, the RIICnSR2.RDRF flag is set to 1 when the R/W# bit is 0 (Wr bit). This causes a receive data full interrupt (INTRIICRI) to be generated. The HOA flag is used to recognize that the host address was sent from the smart battery or other devices.

If the bit following the host address ($0001\ 000_B$) is an Rd bit (R/W# bit = 1), the RIIC can also detect the host address. After the host address is detected, the RIIC operates in the same manner as normal slave operation.

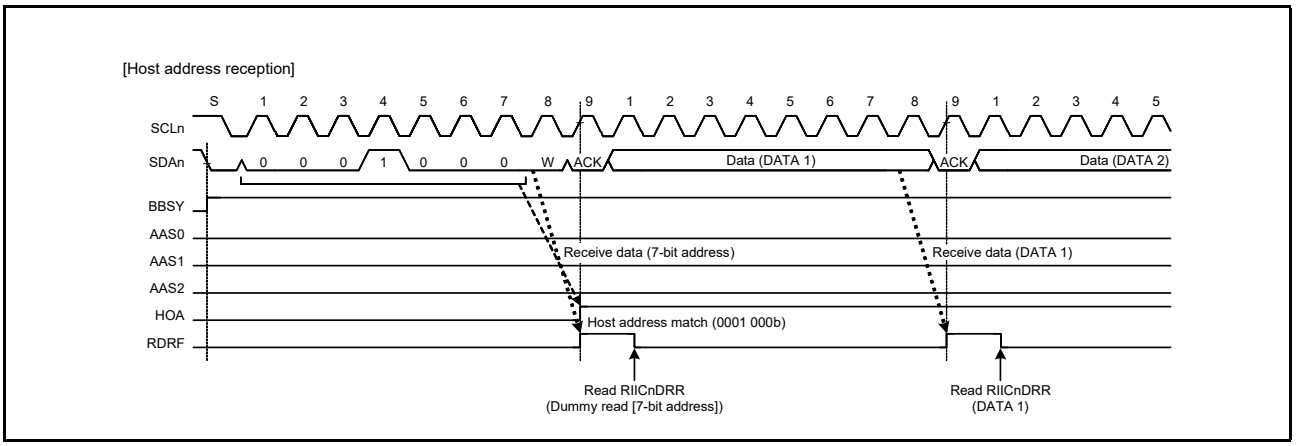


Figure 23.29 HOA Flag Set Timing during Reception of Host Address

23.10 Automatically Low-Hold Function for SCL

23.10.1 Function to Prevent Wrong Transmission of Transmit Data

If the shift register (RIICnDRS) is empty when data have not been written to the transmit data register (RIICnDRT) with the RIIC in transmission mode (RIICnCR2.TRS bit = 1), the SCL signal is automatically held at the low level over the intervals shown below. This low-hold period is extended until data for transmission have been written, which prevents the unintended transmission of erroneous data.

<Master transmit mode>

- Low-level interval after a start condition or restart condition is issued
- Low-level interval of one clock cycle between the ninth clock cycle of one transfer and the first clock cycle of the next

<Slave transmit mode>

- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

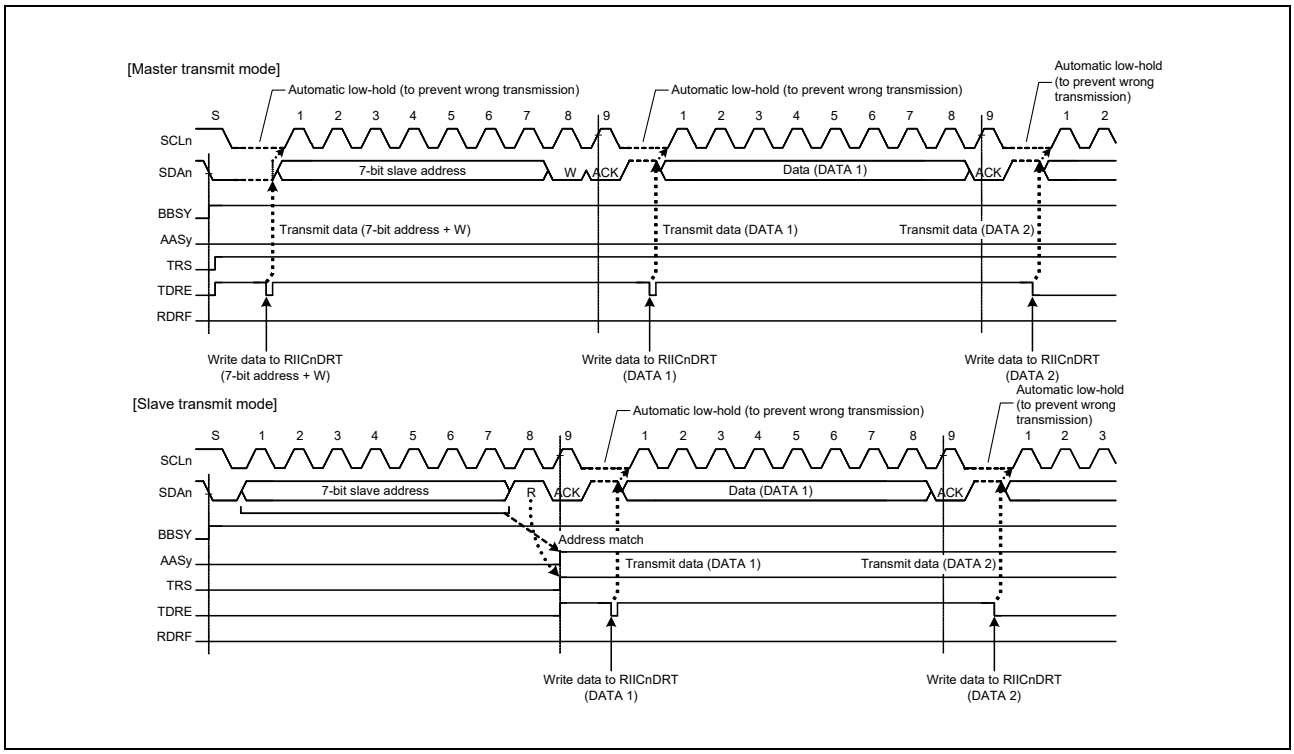


Figure 23.30 Automatic Low-Hold Operation in Transmit Mode

23.10.2 NACK Reception Transfer Suspension Function

The RIIC has a function to suspend transfer operation when NACK is received in transmit mode (RIICnCR2.TRS bit = 1). This function is enabled when the RIICnFER.NACKE bit is set to 1 (transfer suspension enabled). If the next transmit data has already been written (RIICnSR2.TDRE flag = 0) when NACK is received, next data transmission at the falling edge of the ninth SCL clock cycle is automatically suspended. This prevents the SDA line output level from being held low when the MSB of the next transmit data is 0.

If the transfer operation is suspended by this function (RIICnSR2.NACKF flag = 1), transmit operation and receive operation are discontinued. To restore transmit/receive operation, be sure to clear the NACKF flag to 0. In master transmit mode, clear the NACKF flag to 0 after issuing a restart condition or clear the NACKF and STOP flags to 0 after confirming that a stop condition has been issued, and then issue a start condition.

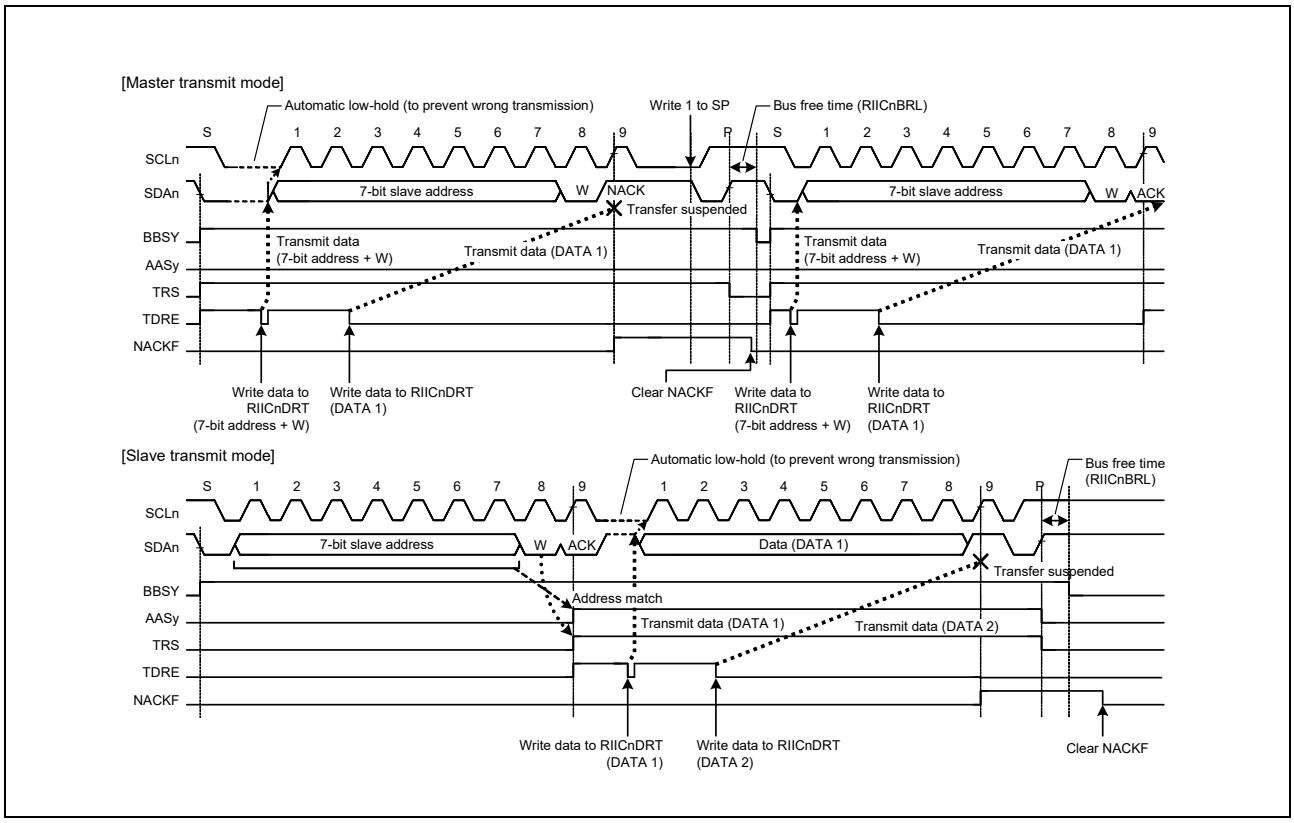


Figure 23.31 Suspension of Data Transfer when NACK is Received (NACKE = 1)

23.10.3 Function to Prevent Failure to Receive Data

If response processing is delayed when receive data (RIICnDRR) read is delayed for a period of one transfer frame or more with receive data full (RIICnSR2.RDRF flag = 1) in receive mode (RIICnCR2.TRS = 0), the RIIC holds the SCL line low automatically immediately before the next data is received to prevent failure to receive data.
This function to prevent failure to receive data using the automatic low-hold function is also enabled even if the read processing of the final receive data is delayed and, in the meantime, the RIIC's own slave address is designated after a stop condition is issued. This function does not disturb other communication because the RIIC does not hold the SCL line low when a mismatch with its own slave address occurs after a stop condition is issued.
Sections in which the SCL line is held low can be selected with a combination of the RIICnMR3.WAIT and RDRFS bits.

(1) One-Byte Receive Operation and Automatic Low-Hold Function Using the WAIT Bit

When the RIICnMR3.WAIT bit is set to 1, the RIIC performs one-byte receive operation using the WAIT bit function. Furthermore, when the RIICnMR3.RDRFS bit is 0, the RIIC automatically sends the RIICnMR3.ACKBT bit value for the acknowledge bit in the period from the falling edge of the eighth SCL clock cycle to the falling edge of the ninth SCL clock cycle, and automatically holds the SCL line low at the falling edge of the ninth SCL clock cycle using the WAIT bit function. This low-hold is released by reading data from RIICnDRR, which enables bytewise receive operation.
The WAIT bit function is enabled for receive frames after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

(2) One-Byte Receive Operation (ACK/NACK Transmission Control) and Automatic Low-Hold Function Using the RDRFS Bit

When the RIICnMR3.RDRFS bit is set to 1, the RIIC performs one-byte receive operation using the RDRFS bit function. When the RIICnSR2.RDRFS bit is set to 1, the RDRF flag (receive data full) in RIICnSR2 is set to 1 at the rising edge of the eighth SCL clock cycle, and the SCL line is automatically held low at the falling edge of the eighth SCL clock cycle. This lowhold is released by writing a value to the RIICnMR3.ACKBT bit, but cannot be released by reading data from RIICnDRR, which enables receive operation by the ACK/NACK transmission control according to the data received in byte units.
The RDRFS bit function is enabled for receive frames after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

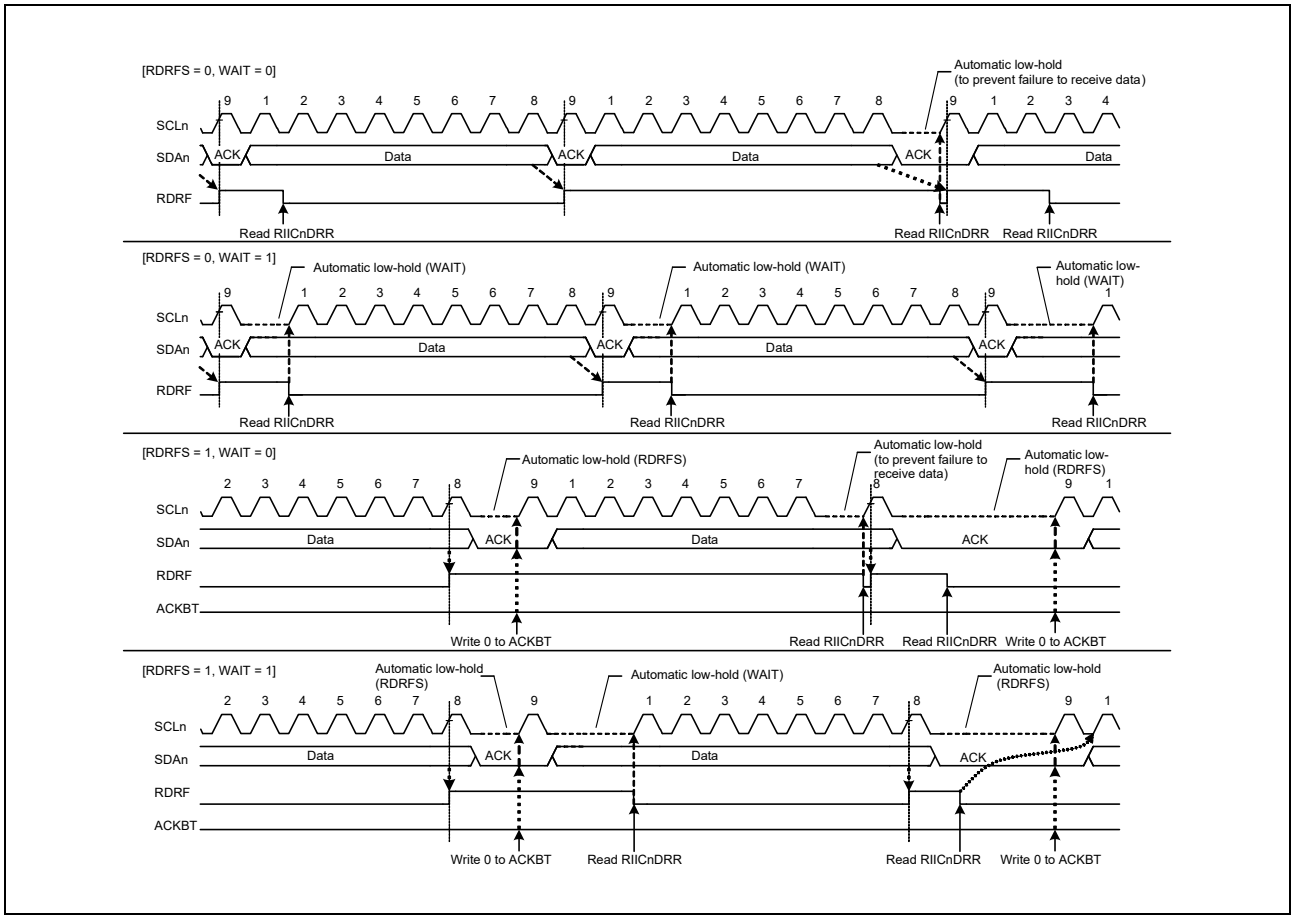


Figure 23.32 Automatic Low-Hold Operation in Receive Mode (Using RDRFS and WAIT Bits)

23.11 Arbitration-Lost Detection Functions

In addition to the normal arbitration-lost detection function defined by the I²C bus standard, the RIIC has functions to prevent double-issue of a start condition, to detect arbitration-lost during transmission of NACK, and to detect arbitration-lost in slave transmit mode.

23.11.1 Master Arbitration-Lost Detection (MALE Bit)

The RIIC drives the SDA line low to issue a start condition. However, if the SDA line has already been driven low by another master device issuing a start condition, the RIIC regards its own issuing of a start condition as an error and considers this a loss in arbitration, so priority is given to transfer by the other master device. Similarly, if a request to issue a start condition is made by setting the RIICnCR2.ST bit to 1 while the bus is busy (RIICnCR2.BBSY flag = 1), the RIIC regards this as a double-issuing-of-start-condition error and considers itself to have lost in arbitration, thus preventing a failure of transfer due to issuing of a start condition while transfer is in progress.
When a start condition is issued successfully, if the data for transmission including the address bits (i.e. the internal SDA output level) and the level on the SDA line do not match (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state) and the low level is detected on the SDA line, the RIIC loses in arbitration.
After a loss in arbitration of mastership, the RIIC immediately enters slave receive mode. If a slave address (including the general call address) matches its own address at this time, the RIIC continues in slave operation.
A loss in arbitration of mastership is detected when the following conditions are met while the RIICnFER.MALE bit is 1 (master arbitration-lost detection enabled).
[Master arbitration-lost conditions]

- Non-matching of the internal level for output on SDA and the level on the SDA line after a start condition was issued by setting the RIICnCR2.ST bit to 1 while the RIICnCR2.BBSY flag was cleared to 0 (erroneous issuing of a start condition)
- Setting of the RIICnCR2.ST bit to 1 (start condition double-issue error) while the RIICnCR2.BBSY flag is set to 1
- When the transmit data excluding acknowledge (internal SDA output level) does not match the level on the SDA line in master transmit mode (RIICnCR2.MST and TRS bits = 11_B)

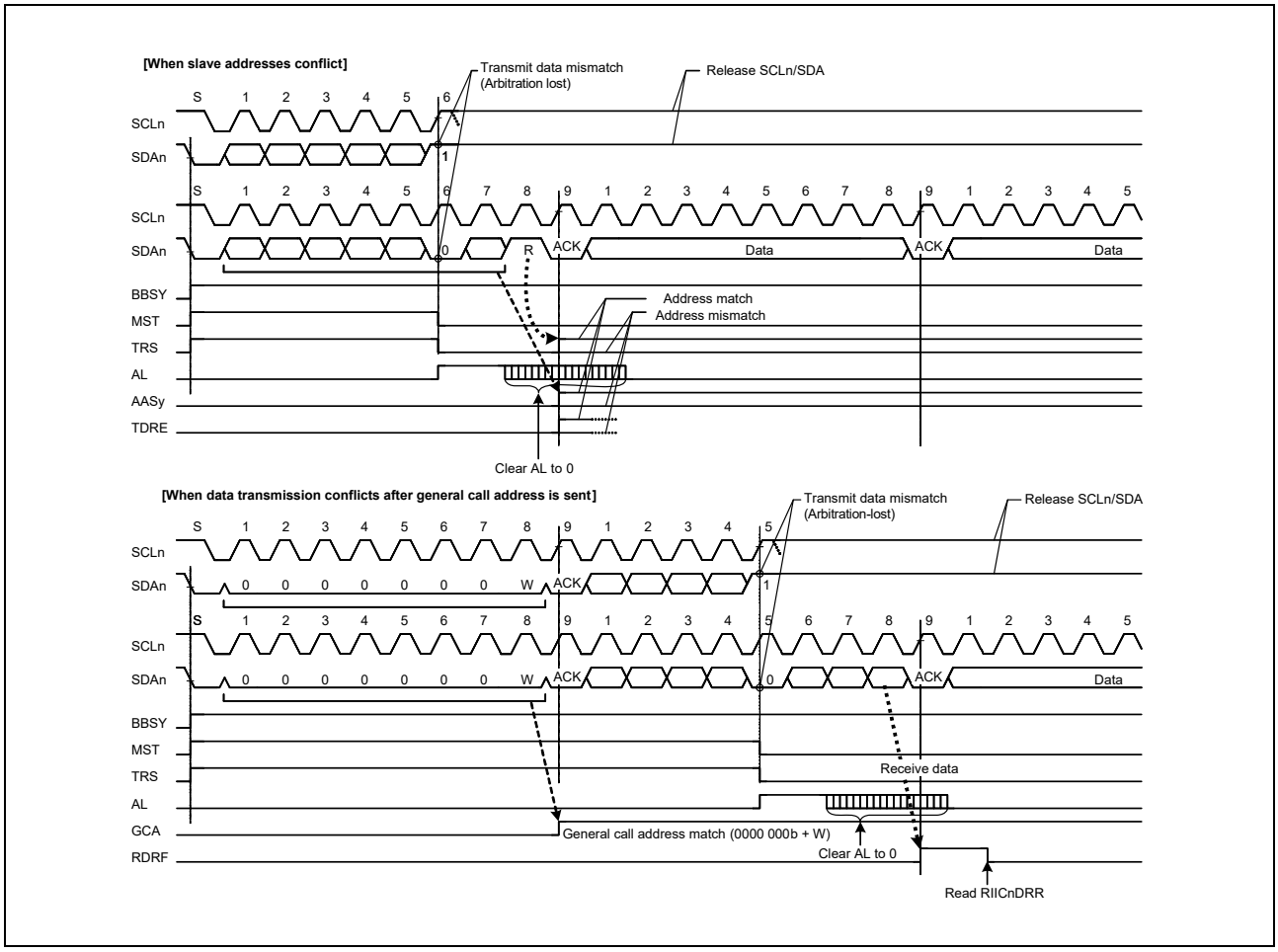


Figure 23.33 Examples of Master Arbitration-Lost Detection (MALE = 1)

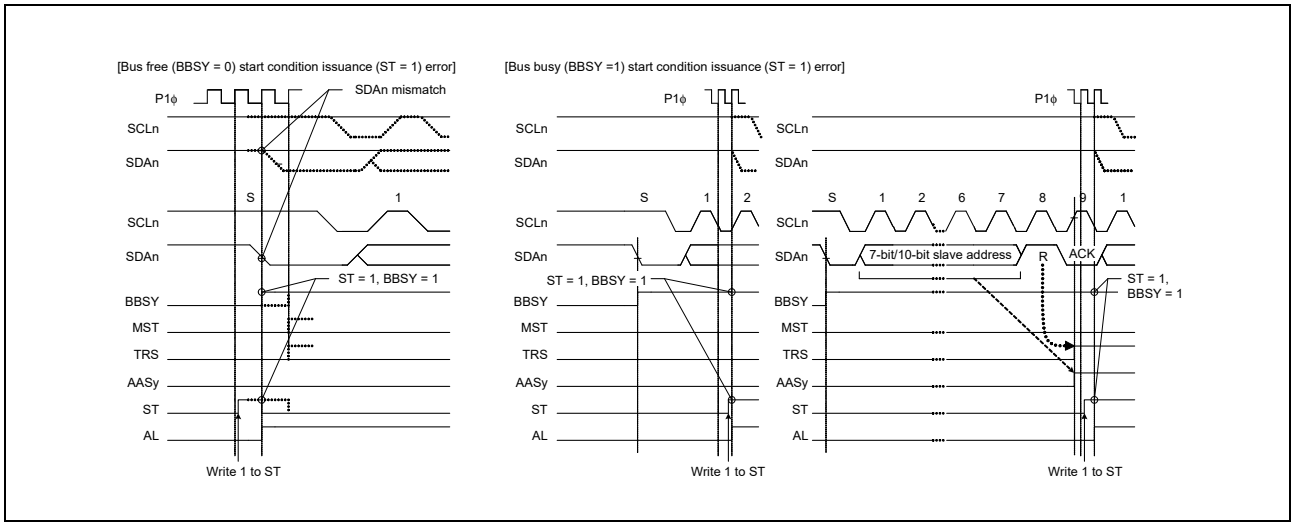


Figure 23.34 Arbitration-Lost when a Start Condition is Issued (MALE = 1)

23.11.2 Function to Detect Loss of Arbitration during NACK Transmission (NALE Bit)

The RIIC has a function to cause arbitration to be lost if the internal SDA output level does not match the level on the SDA line (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state) and the low level is detected on the SDA line during transmission of NACK in receive mode. Arbitration is lost due to a conflict of NACK transmission and ACK transmission when two or more master devices receive data from the same slave device simultaneously in a multi-master system. Such conflict occurs when multiple master devices send/receive the same information through a single slave device. Figure 23.35 shows an example of arbitration-lost detection during transmission of NACK.

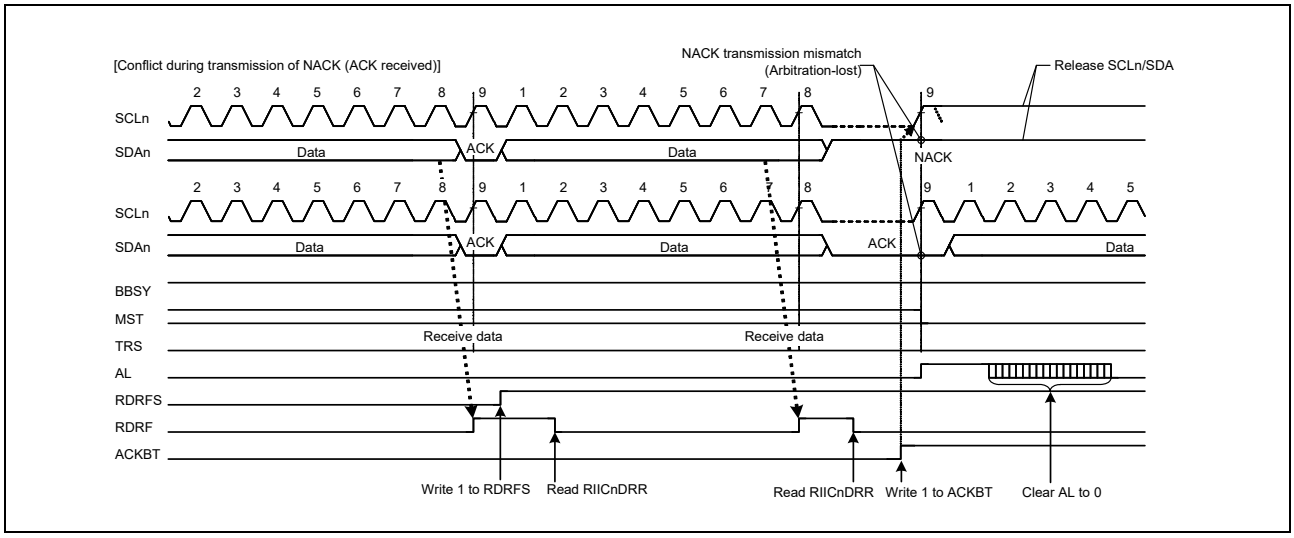


Figure 23.35 Example of Arbitration-Lost Detection during Transmission of NACK (NALE = 1)

The following explains arbitration-lost detection using an example where two master devices (master A and master B) and a single slave device are connected through the bus. In this example, master A receives two bytes of data from the slave device, and master B receives four bytes of data from the slave device.

If master A and master B access the slave device simultaneously, because the slave address is identical, arbitration is not lost in both master A and master B during access to the slave device. Therefore, both master A and master B recognize that they have obtained the bus mastership and operate as such. Here, master A sends NACK when it has received two final bytes of data from the slave device. Meanwhile, master B sends ACK because it has not received necessary four bytes of data. At this time, the NACK transmission from master A and the ACK transmission from master B conflict. In general, if a conflict like this occurs, master A cannot detect ACK transmitted by master B and issues a stop condition. Therefore, the issuance of the stop condition conflicts with the SCL clock output of master B, which disturbs communication.

When the RIIC receives ACK during transmission of NACK, it detects a defeat in conflict with other master devices and causes arbitration to be lost.

If arbitration is lost during transmission of NACK, the RIIC enters slave receive mode. This prevents a stop condition from being issued, preventing a communication failure on the bus.

Similarly, in the ARP command processing of SMBus, the function to detect loss of arbitration during transmission of NACK is also available for eliminating the extra clock cycle processing (such as FF_H transmission processing) necessary if the UDID (Unique Device Identifier) of assign address does not match in the Get UDID (general) processing after the Assign address command.

The RIIC detects arbitration-lost during transmission of NACK when the following condition is met with the RIICnFER.NALE bit set to 1 (arbitration-lost detection during NACK transmission enabled).

[Condition for arbitration-lost during NACK transmission]

When the internal SDA output level does not match the SDA line (ACK is received) during transmission of NACK (RIICnMR3.ACKBT bit = 1).

23.11.3 Slave Arbitration-Lost Detection (SALE Bit)

The RIIC has a function to cause arbitration to be lost if the data for transmission (i.e. the internal SDA output level) and the level on the SDA line do not match (the high output as the internal SDA output; i.e. the SDA pin is in the highimpedance state) and the low level is detected on the SDA line in slave transmit mode. This arbitration-lost detection function is mainly used when transmitting a UDID (Unique Device Identifier) over an SMBus.

When it loses slave arbitration, the RIIC enters slave receive mode. This function can detect conflicts of data during transmission of UDIDs over an SMBus and eliminates subsequent redundant processing (processing for the transmission of FF_H).

The RIIC detects slave arbitration-lost when the following condition is met with the RIICnFER.SALE bit set to 1 (slave arbitration-lost detection enabled).

[Condition for slave arbitration-lost]

When transmit data excluding acknowledge (internal SDA output level) does not match the SDA line in slave transmit mode (RIICnCR2.MST and TRS bits = 01_B)

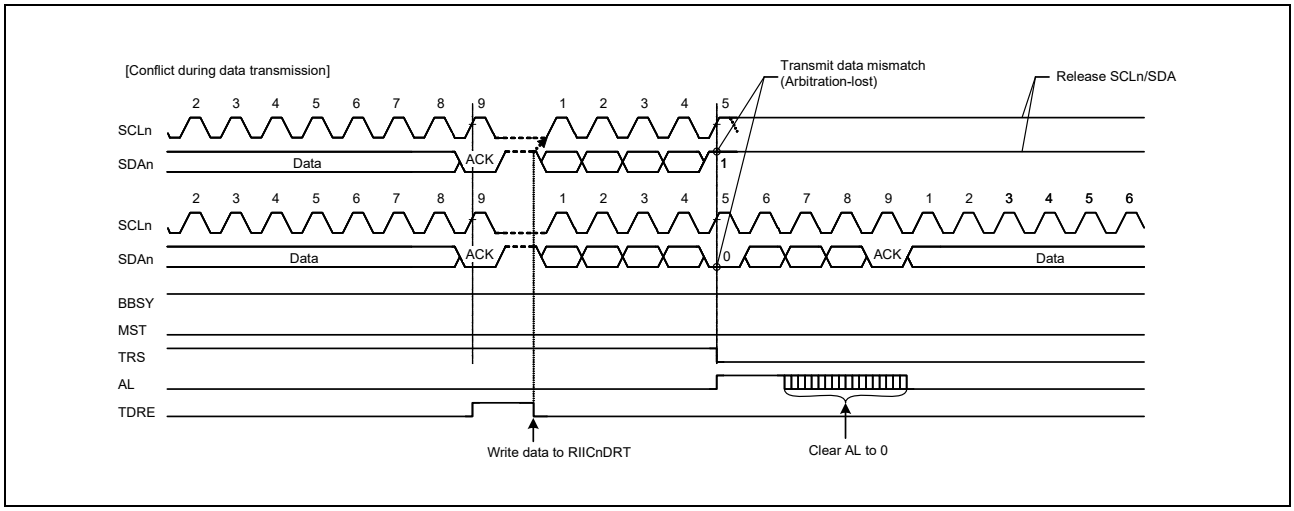


Figure 23.36 Example of Slave Arbitration-Lost Detection (SALE = 1)

23.12 Start Condition/Restart Condition/Stop Condition Issuing Function

23.12.1 Issuing a Start Condition

The RIIC issues a start condition when the RIICnCR2.ST bit is set to 1.
When the ST bit is set to 1, a start condition issuance request is made and the RIIC issues a start condition when the RIICnCR2.BBSY flag is 0 (bus free). When a start condition is issued normally, the RIIC automatically shifts to the master transmit mode.
A start condition is issued in the following sequence.
[Start condition issuance]

- Drive the SDA line low (high level to low level).
- Ensure the time set in RIICnBRH and the start condition hold time.
- Drive the SCL line low (high level to low level).
- Detect low level of the SCL line and ensure the low-level period of SCL line set in RIICnBRL.

23.12.2 Issuing a Restart Condition

The RIIC issues a restart condition when the RIICnCR2.RS bit is set to 1.
When the RS bit is set to 1, a restart condition issuance request is made and the RIIC issues a restart condition when the RIICnCR2.BBSY flag is 1 (bus busy) and the RIICnCR2.MST bit is 1 (master mode).
A restart condition is issued in the following sequence.
[Restart condition issuance]

- Release the SDA line.
- Ensure the low-level period of SCL line set in RIICnBRL.
- Release the SCL line (low level to high level).
- Detect a high level of the SCL line and ensure the time set in RIICnBRL and the restart condition setup time.
- Drive the SDA line low (high level to low level).
- Ensure the time set in RIICnBRH and the restart condition hold time.
- Drive the SCL line low (high level to low level).
- Detect a low level of the SCL line and ensure the low-level period of SCL line set in RIICnBRL.

Caution: To issue a restart condition request, set RIICnSR2.RS to 1 after setting RIICnSR2.START to 0. After checking that RIICnSR2.START is 1, set RIICnSR2.START to 0, and then write the slave address to RIICnDRT.

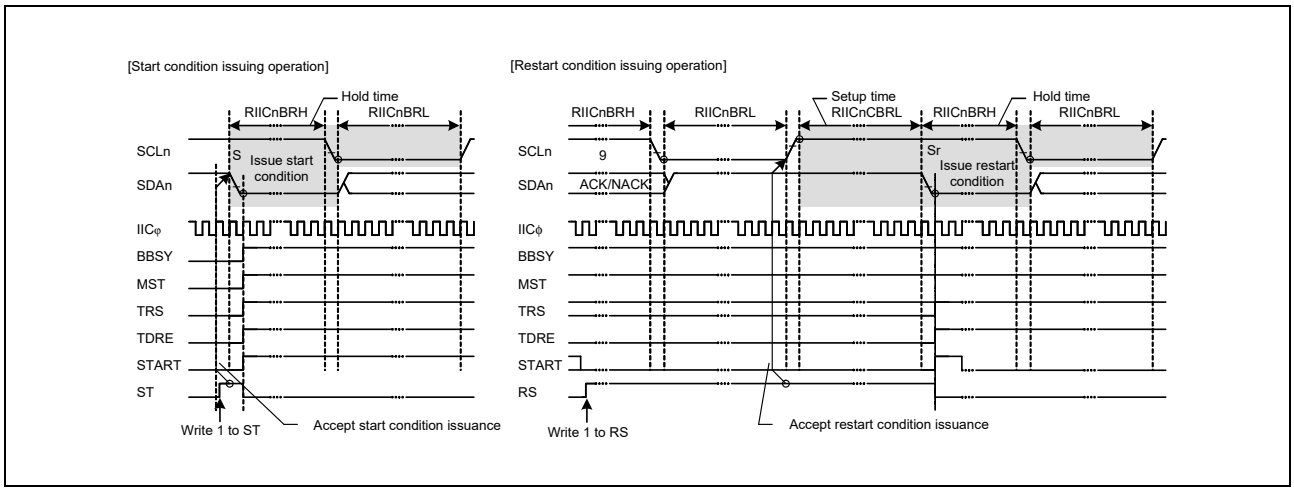


Figure 23.37 Start Condition/Restart Condition Issue Timing (ST and RS Bits)

23.12.3 Issuing a Stop Condition

The RIIC issues a stop condition when the RIICnCR2.SP bit is set to 1.
When the SP bit is set to 1, a stop condition issuance request is made and the RIIC issues a stop condition when the RIICnCR2.BBSY flag is 1 (bus busy) and the RIICnCR2.MST bit is 1 (master mode).
A stop condition is issued in the following sequence.
[Stop condition issuance]

- Drive the SDA line low (high level to low level).
- Ensure the low-level period of SCL line set in RIICnBRL.
- Release the SCL line (low level to high level).
- Detect a high level of the SCL line and ensure the time set in RIICnBRH and the stop condition setup time.
- Release the SDA line (low level to high level).
- Ensure the time set in RIICnBRL and the bus free time.
- Clear the BBSY flag to 0 (to release the bus mastership).

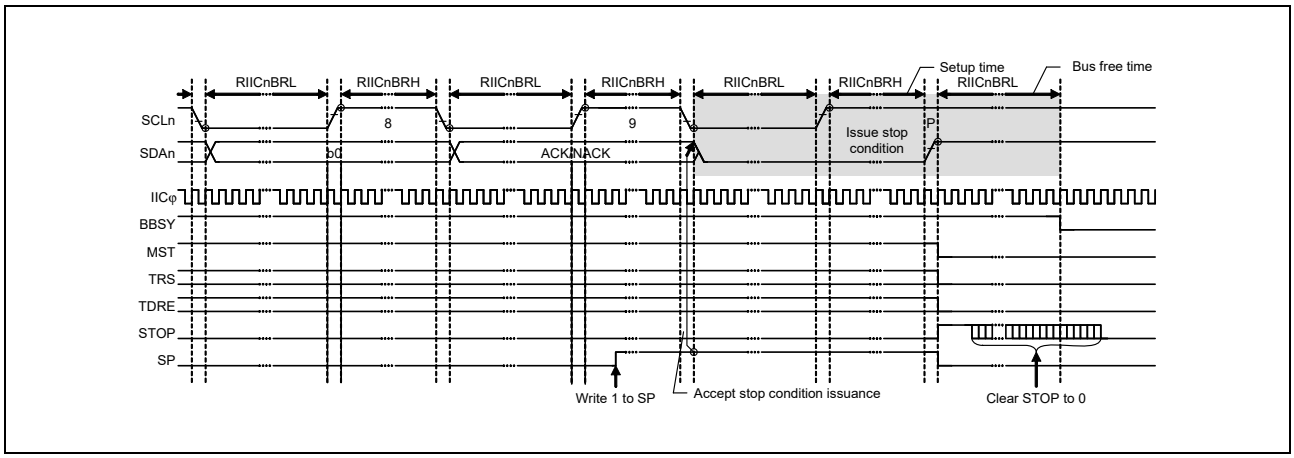


Figure 23.38 Stop Condition Issue Timing (SP Bit)

23.13 Bus Hanging

If the clock signals from the master and slave devices go out of synchronization due to noise or other factors, the I^2C bus might hang with a fixed level on the SCL line and/or SDA line.
As measures against the bus hanging, the RIIC has a timeout function to detect hanging by monitoring the SCL line, a function for the output of an extra SCL clock cycle to release the bus from a hung state due to clock signals being out of synchronization, and the RIIC/internal reset function.
By checking the RIICnCR1.SCLO, SDAO, SCLI, and SDAI bits, it is possible to see whether the RIIC or its partner in communications is placing the low level on the SCL or SDA lines.

23.13.1 Timeout Function

The RIIC has the timeout function to detect an abnormality that the SCL line is held for a certain period of time. The RIIC can detect an abnormal bus state by monitoring that the SCL line is held low or high for a predetermined time.
The timeout function monitors the SCL line state and counts the low-level period or high-level period using the internal counter. The timeout function resets the internal counter each time the SCL line changes (rising or falling), but continues to count unless the SCL line changes. If the internal counter overflows due to no SCL line change, the RIIC can detect the timeout and report the bus abnormality.
This timeout function is enabled when the RIICnFER.TMOE bit is 1. It detects an abnormal bus state in which the SCL line is held low or high in the following cases.

1. When the bus is busy (RIICnCR2.BBSY = 1) in master mode (RIICnCR2.MST = 1)
2. When the bus is busy (RIICnCR2.BBSY = 1) and the RIIC's own slave address matches (RIICnSR1 is not 00_H) in slave mode (RIICnCR2.MST = 0)
3. While the bus is free (RIICnCR2.BBSY = 0) and issuing of a start condition is being requested (RIICnCR2.ST = 1).

The internal counter of the timeout function works using the internal reference clock (IICφ) set by the RIICnMR1.CKS[2:0] bits as a count source. It functions as a 16-bit counter when long mode is selected (RIICnMR2.TMOS bit = 0) or a 14-bit counter when short mode is selected (TMOS bit = 1).
The SCL line level (low/high or both levels) during which this counter is activated can be selected by the setting of the RIICnMR2.TMOH and TMOL bits. If both TMOL and TMOH bits are cleared to 0, the internal counter does not work.

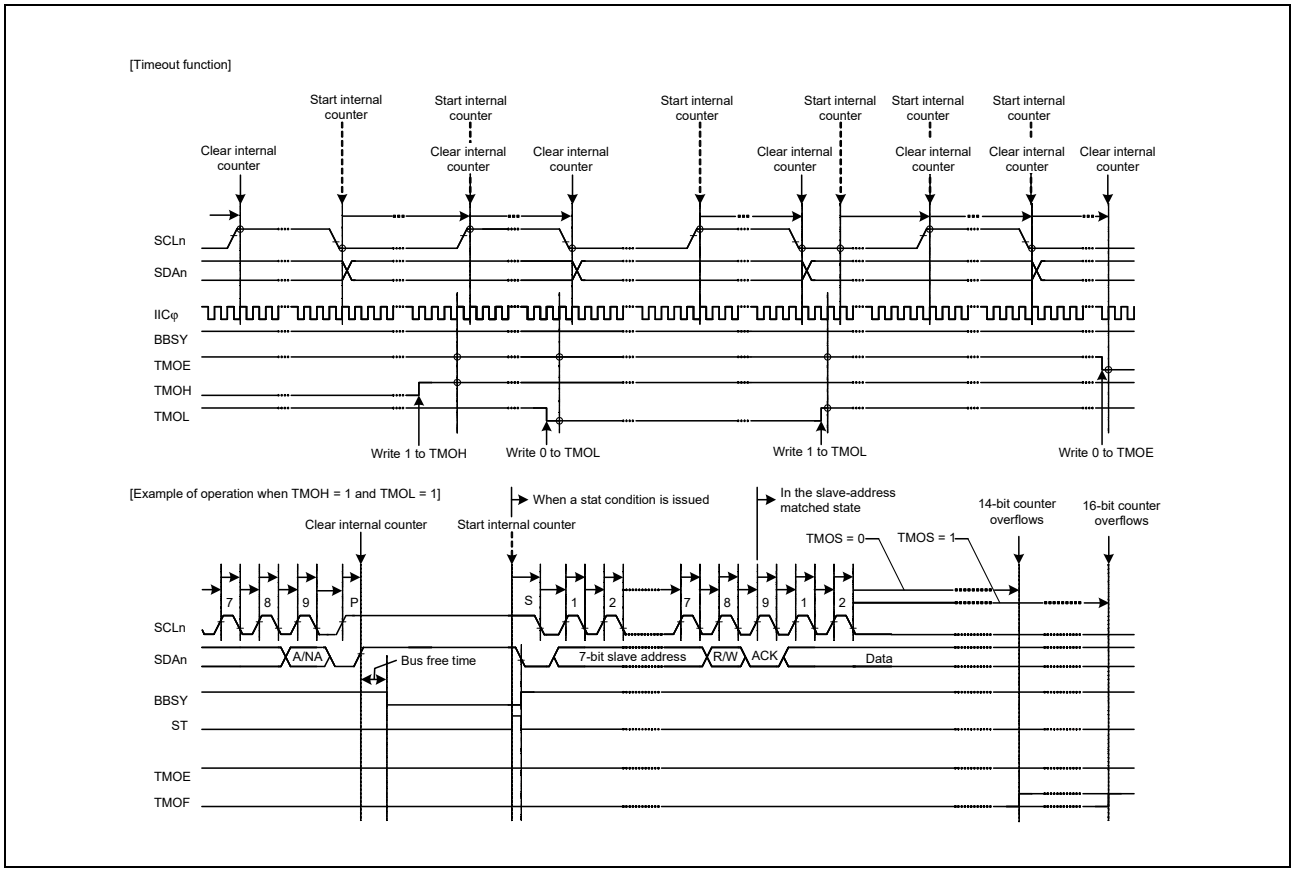


Figure 23.39 Timeout Function (TMOE, TMOS, TMOH, and TMOL Bits)

23.13.2 Extra SCL Clock Cycle Output Function

In master mode, the RIIC module has a facility for the output of extra SCL (clock) cycles to release the SDA line of the slave device from being held at the low level due to the master being out of synchronization with the slave device.
This function is mainly used in master mode to release the SDA line of the slave device from the state of being fixed to the low level by including extra cycles of SCL output from the RIIC with single cycles of the SCL (clock) signal as the unit in the case of a bus error where the RIIC cannot issue a stop condition because the slave device is holding the SDA line at the low level. Do not use this facility in normal situations. Using it when communications are proceeding correctly will lead to malfunctions.
When the RIICnCR1.CLO bit is set to 1 in master mode, a single cycle of the SCL clock at the frequency corresponding to the transfer rate settings (settings of the RIICnMR1.CKS[2:0] bits, and of the RIICnBRH and RIICnBRL registers) is output as an extra clock cycle. After output of this single cycle of the SCL clock, the CLO bit is automatically cleared to 0. At this point, the SCL pin output is held at a low level if the BBSY flag is 1, or the SCL pin outputs at high level if the BBSY flag is 0. Further clock cycles can consecutively be output by writing 1 to the CLO bit after having read CLO is 0 by software.
When the RIIC module is in master mode and the slave device is holding the SDA line at the low level because synchronization with the slave device has been lost due to the effects of noise, etc., the output of a stop condition is not possible. The facility for output of an extra cycle of the SCL (clock) signal can be used to output extra cycles of SCL one by one to make the slave device release the SDA line from being held at the low level, thus recovering the bus from an unusable state. Release of the SDA line by the slave device can be monitored by reading the RIICnCR1.SDAI bit. After confirming release of the SDA line by the slave device, complete communications by reissuing the stop condition.
[Output conditions for using the RIICnCR1.CLO bit]

- When the bus is free (RIICnCR2.BBSY flag = 0) or in master mode (RIICnCR2.MST bit = 1 and BBSY flag = 1)
- When the communication device does not hold the SCL line low

Figure 23.40 shows the operation timing of the extra SCL clock cycle output function (CLO bit).

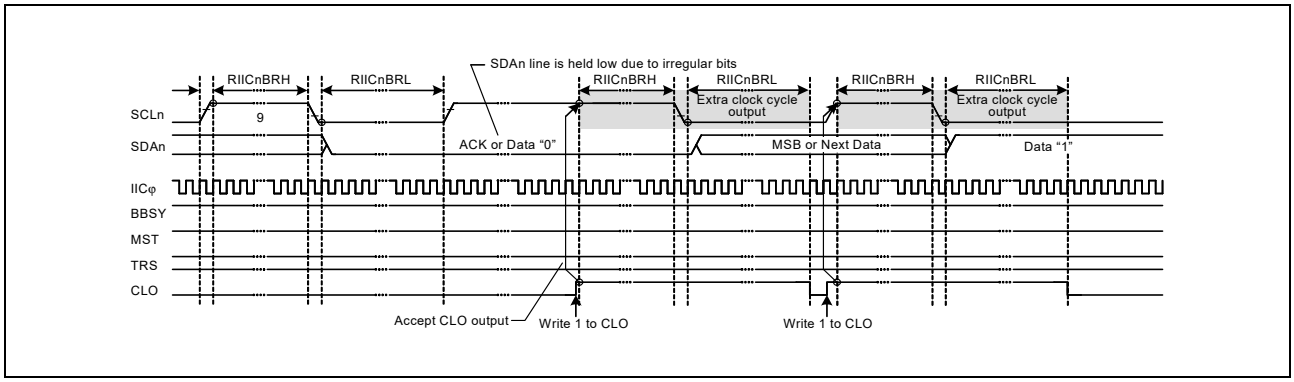


Figure 23.40 Extra SCL Clock Cycle Output Function (CLO Bit)

23.13.3 RIIC Reset and Internal Reset

The RIIC module incorporates a function for resetting itself. There are two types of reset. One is referred to as an RIIC reset; this initializes all registers including the RIICnCR2.BBSY flag. The other is referred to as an internal reset; this releases the RIIC from the slave-address matched state and initializes the internal counter while retaining other settings. After issuing a reset, be sure to clear the RIICnCR1.IICRST bit to 0.

Both types of reset are effective for release from bus-hung states since both restore the output state of the SCL and SDA pins to the high impedance state.

Issuing a reset during slave operation may lead to a loss of synchronization between the master device clock and the slave device clock, so avoided this where possible. Note that monitoring of the bus state, such as for the presence of a start condition, is not possible during an RIIC reset (RIICnCR1.ICE and IICRST bits = 01_B).

For a detailed description of the RIIC and internal resets, see section 23.15, Reset Function of RIIC.

23.14 SMBus Operation

The RIIC is available for data communication conforming to the SMBus (Version 2.0). To perform SMBus communication, set the RIICnMR3.SMBS bit to 1. To use the transfer rate within a range of 10 kbps to 100 kbps of the SMBus standard, set the RIICnMR1.CKS[2:0] bits, RIICnCBRH, and RIICnBRL. In addition, determine the values of the RIICnMR2.DLCS bit and the RIICnMR2.SDDL[2:0] bits to meet the data hold time specification of 300 ns or more. If the RIIC is used only as a slave device, the transfer rate setting is not necessary. When the RIIC is used only as a slave device, the transfer rate setting is not necessary, whereas the RIICnBRL needs to be set to a value at least the same as the data setup time (250 ns).
For the SMBus device default address ($1100\ 001_B$), use one of the slave address registers L0 to L2 (RIICnSAR0, RIICnSAR1, and RIICnSAR2), and set the corresponding RIICnSARy.FSy bit (7-bit/10-bit address format select) (y = 0 to 2) to 0 (7-bit address format).
When transmitting the UDID (Unique Device Identifier), set the RIICnFER.SALE bit to 1 to enable the slave arbitration lost detection function.

23.14.1 SMBus Timeout Measurement

(1) Measuring timeout of slave device

The following period (timeout interval: $T_{LOW:\ SEXT}$) must be measured for slave devices in SMBus communication.

- From start condition to stop condition
  To measure timeout for slave devices, measure the period from start condition detection to stop condition detection with the internal timer using a start condition detection interrupt (INTRIICSTI) and stop condition detection interrupt (INTRIICSPI) of the RIIC. The measured timeout period must be within the total clock low-level period [slave device] $T_{LOW:\ SEXT}$: 25 ms (max.) of the SMBus standard.
  If the time measured with the internal timer exceeds the clock low-level detection timeout $T_{TIMEOUT}$: 25 ms (min.) of the SMBus standard, the slave device must release the bus by writing 1 to the RIICnCR1.IICRST bit to issue an internal reset of the RIIC. When an internal reset is issued, the RIIC stops driving the bus for the SCL pin and SDA pin and make the SCL/SDA pin outputs high impedance, which releases the bus.

(2) Measuring timeout of master device

The following periods (timeout interval: $T_{LOW:\ MEXT}$) must be measured for master devices in SMBus communication.

- From start condition to acknowledge bit
- Between acknowledge bits
- From acknowledge bit to stop condition

To measure timeout for master devices, measure these periods with the internal timer using a start condition detection interrupt (INTRIICSTI), stop condition detection interrupt (INTRIICSPI), and transmit end interrupt (INTRIICTEI) or receive data full interrupt (INTRIICRI) of the RIIC. The measured timeout period must be within the total clock low-level extended period [master device] $T_{LOW:\ MEXT}$: 10 ms (max.) of the SMBus standard, and the total of all $T_{LOW:MEXT}$ from start condition to stop condition must be within $T_{LOW:\ SEXT}$: 25 ms (max.).
For the ACK receive timing (rising edge of the ninth SMBCLK clock cycle), monitor the RIICnSR2.TEND flag in master transmit mode (master transmitter) and the RIICnSR2.RDRF flag in master receive mode (master receiver). For this reason, perform bytewise transmit operation in master transmit mode, and hold the RIICnMR3.RDRFS bit 0 until the byte just before reception of the final byte in master receive mode. While the RDRFS bit is 0, the RDRF flag is set to 1 at the rising edge of the ninth SMBCLK clock cycle.
If the period measured with the internal timer exceeds the total clock low-level extended period [master device] $T_{LOW:MEXT}$: 10 ms (max.) of the SMBus standard or the total of measured periods exceeds the clock low-level detection timeout $T_{TIMEOUT}$: 25 ms (min.) of the SMBus standard, the master device must stop the transaction by issuing a stop condition. In master transmit mode, immediately stop the transmit operation (writing data to RIICnDRT).

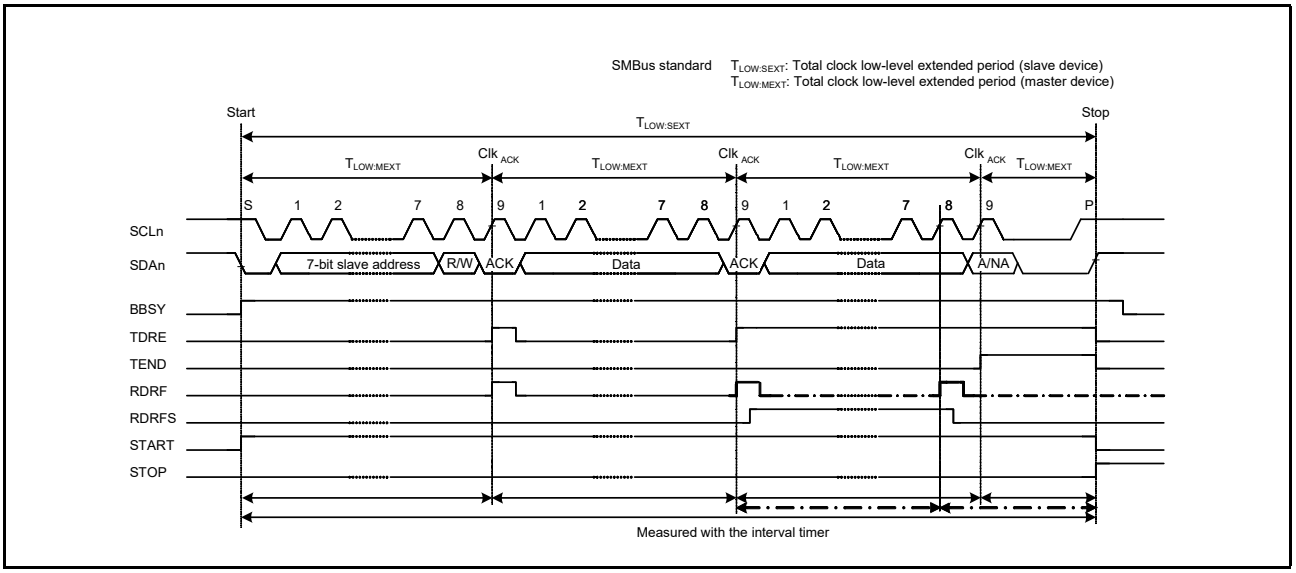


Figure 23.41 SMBus Timeout Measurement

23.14.2 SMBus Host Notification Protocol/Notify ARP Master

In communications over an SMBus, a slave device can temporarily act as a master device to notify the SMBus host (or ARP master) of (or request the SMBus host for) its own slave address or to request its own slave address from the SMBus host.

For this LSI to operate as an SMBus host (or ARP master), the host address ($0001\ 000_B$) sent from the slave device must be detected as a slave address, so the RIIC has a function for detecting the host address. To detect the host address as a slave address, set the RIICnMR3.SMBS bit and the RIICnSER.HOAE bit to 1. Operation after the host address has been detected is the same as normal slave operation.

23.15 Reset Function of RIIC

The RIIC has chip reset, RIIC reset, and internal reset functions. Table 23.24 lists the scope of each reset and reset conditions.

Table 23.24 RIIC Reset Functions

Register		RIIC Reset (ICE = 0, IICRST = 1)	Internal Reset (ICE = 1, IICRST = 1)	Start Condition/Restart Condition Detection	Stop Condition Detection
RIICnCR1	ICE	0	1	Retained	Retained
	IICRST	1	1	Retained	Retained
	CLO	Initialized	Retained	Retained	Retained
	SOWP	Initialized	Retained	Retained	Retained
	SCLO	Initialized	Initialized	Retained	Retained
	SDAO	Initialized	Initialized	Retained	Retained
	SCLI	Initialized	Retained	Retained	Retained
	SDAI	Initialized	Retained	Retained	Retained
RIICnCR2	BBSY	Initialized	Initialized*1	Operation	Retained
	MST	Initialized	Initialized	Operation (retained)	Initialized
	TRS	Initialized	Initialized	Operation (retained)	Initialized
	SP	Initialized	Initialized	Initialized	Initialized
	RS	Initialized	Initialized	Initialized	Initialized
	ST	Initialized	Initialized	Initialized	Retained
RIICnMR1	CKS[2:0]	Initialized	Retained	Retained	Retained
	BCWP	Initialized	Retained	Retained	Retained
	BC[2:0]	Initialized	Initialized	Initialized	Retained
RIICnMR2		Initialized	Retained	Retained	Retained
RIICnMR3	WAIT	Initialized	Retained	Retained	Retained
	RDRFS	Initialized	Retained	Retained	Retained
	ACKWP	Initialized	Retained	Retained	Retained
	ACKBT	Initialized	Retained	Retained	Initialized
	ACKBR	Initialized	Retained	Retained	Retained
	NF[1:0]	Initialized	Retained	Retained	Retained
RIICnFER		Initialized	Retained	Retained	Retained
RIICnSER		Initialized	Retained	Retained	Retained
RIICnIER		Initialized	Retained	Retained	Retained
RIICnSR1	DID	Initialized	Initialized	Retained	Initialized
	GCA	Initialized	Initialized	Retained	Initialized
	AAS2	Initialized	Initialized	Retained	Initialized
	AAS1	Initialized	Initialized	Retained	Initialized
	AAS0	Initialized	Initialized	Retained	Initialized
RIICnSR2	TDRE	Initialized	Initialized	Retained	Initialized
	TEND	Initialized	Initialized	Retained	Initialized
	RDRF	Initialized	Initialized	Retained	Retained
	NACKF	Initialized	Initialized	Retained	Retained
	STOP	Initialized	Initialized	Retained	Operation
	START	Initialized	Initialized	Operation	Initialized
	AL	Initialized	Initialized	Retained	Retained
	TMOF	Initialized	Initialized	Retained	Retained

Table 23.24 RIIC Reset Functions

Register	RIIC Reset (ICE = 0, IICRST = 1)	Internal Reset (ICE = 1, IICRST = 1)	Start Condition/Restart Condition Detection	Stop Condition Detection
RIICnSAR0, RIICnSAR1, RIICnSAR2	Initialized	Retained	Retained	Retained
RIICnBRH, RIICnBRL	Initialized	Retained	Retained	Retained
RIICnDRT	Initialized	Retained	Retained	Retained
RIICnDRR	Initialized	Retained	Retained	Retained
RIICnDRS	Initialized	Initialized	Retained	Retained

Note 1. When an internal reset is applied while the bus is free after detection of a stop condition, the setting of the BBSY flag is 0 while the bus is free following de-assertion of the internal reset signal.
When an internal reset is applied while the bus is not free, the BBSY flag is not cleared.

24. Serial Sound Interface (SSIF-2)

24.1 Overview

The serial sound interface (SSIF-2) transmits and receives audio data to and from various devices that are compatible with I²S format, monaural format and TDM format.

24.1.1 Features

Table 24.1 Features of SSIF-2

Item		Description
Number of interfaces		• 4
Communication mode		• Master/slave • Full-duplex communication is available in interfaces 0, 1, and 3. Half-duplex communication is only available in interface 2.
Communication format		• I²S format • Monaural format • TDM format
Serial data		• MSB first • Data can be left-justified or right-justified. • Data delay (one clock cycle) or no delay selectable for the period from SSILRCK/SSIFS to SSITxD/SSIRxD • System word length: 8, 16, 24, 32, 48, 64, 128, or 256 bits • Data word length: 8, 16, 18, 20, 22, 24, or 32 bits • Padding polarity: Low or high
Bit clock (SSIBCK)	In master mode	• Two clock sources available (AUDIO_CLK/AUDIO_X) • Clock source division ratio: 1/1, 1/2, 1/4, 1/6, 1/8, 1/12, 1/16, 1/24, 1/32, 1/48, 1/64, 1/96, and 1/128. • Supply/stop of SSIBCK is selectable while communication is halted.
	In slave mode	• SSIBCK input method: Direct input or via the noise canceller
	In master/slave mode	• Polarity: Rising/falling edge canceler
Left and right clock/frame synchronization (SSILRCK/SSIFS)	In master mode	• Polarity: Low/high level selectable • Supply/stop of LRCK is selectable while communication is halted.
	In slave mode	• Two input methods: Direct input or via the noise canceller
Transmit data (SSITxD) and receive data (SSIRxD)	Transmission	• Mute: Transmission of transmit FIFO data or fixed value 0 selectable
	Reception (in slave mode)	• Two input methods: Direct input or via the noise canceller
FIFO	Capacity	• Transmit FIFO/receive FIFO: 4 bytes × 32 stages, • Receive FIFO: 4 bytes × 32 stages
	Data alignment	• Method of data transfer between FIFO and shift register (Left-justification and right-justification) selectable.
Interrupt	Interrupt output	• Communication error/idle mode error (level) • Receive data full interrupt (edge) • Transmit data empty interrupt (edge) • Receive data full/transmit data empty interrupt (edge)
	Interrupt capture spilling solution function	• Edge Interrupt supports inhibition function of capture spilling.
Low power consumption function		• Supply/stop of audio clock is selectable in master-mode communication.

The definition of words and phrases used by the format of the communication of SSIF-2 is as follows.

Table 24.2 Word Definition List

Word	Definition
Start trigger	The first edge that reaches value that set to LRCKP in SSILRCK/ SSIFS pin after SSIF-2 permit communication.
Frame boundary	When SSIF-2 begins to transmit the first data of one frame. When SSIF-2 forwarding ends final data of one frame.
Frame word number	Sound channel number in one frame
System word length	Bit length in one frame.
Data word length	Effective bit length in one frame.
Control bit of communication format	• SSICR register: FRM,DWL,SWL,LRCKP,SPDP,SDTA,PDTA,DEL • SSIFCR register: BSW • SSIOFR register: OMOD • SSISCR register: TDES, RDFS

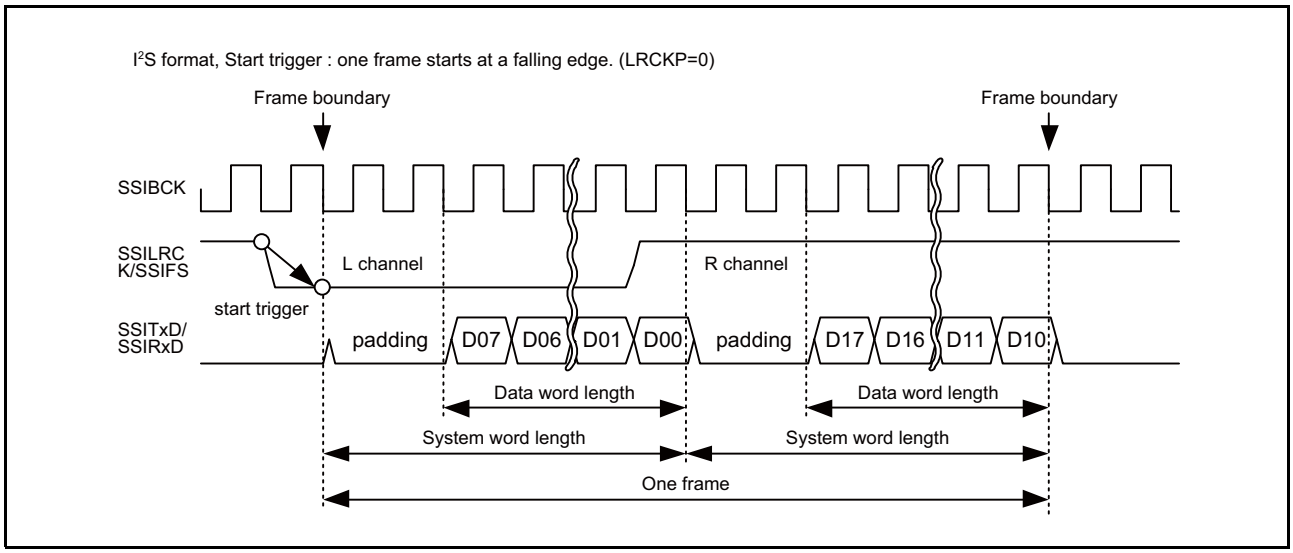
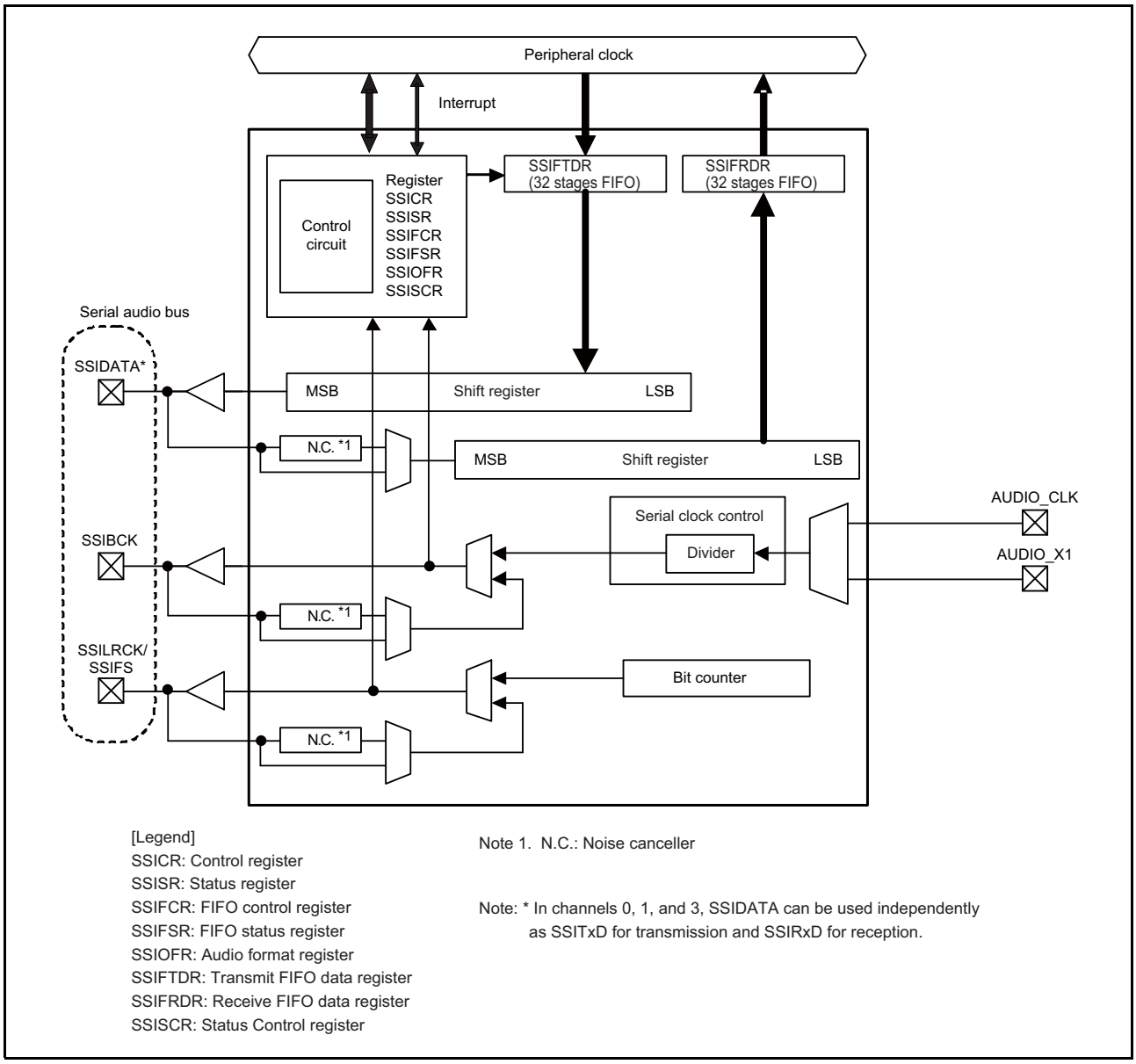


Figure 24.1 Communication Format Definition

24.1.2 Block Diagram

Figure 24.2 shows the block diagram of SSIF-2.



[Legend]
SSICR: Control register
SSISR: Status register
SSIFCR: FIFO control register
SSIFSR: FIFO status register
SSIOFR: Audio format register
SSIFTDR: Transmit FIFO data register
SSIFRDR: Receive FIFO data register
SSISCR: Status Control register

Note 1. N.C.: Noise canceller

Note: * In channels 0, 1, and 3, SSIDATA can be used independently as SSITxD for transmission and SSIRxD for reception.

Figure 24.2 SSIF-2 Block Diagram

Figure 24.3 shows the clock configuration of SSIF-2.

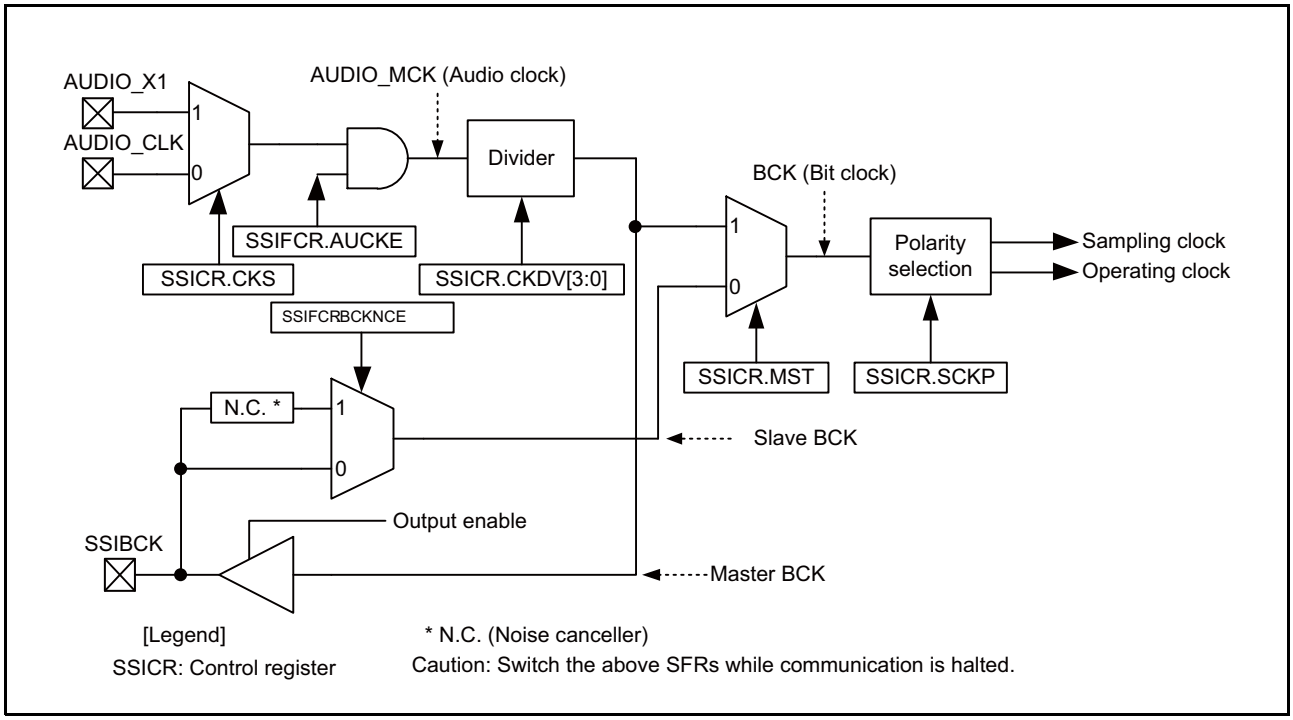


Figure 24.3 SSIF-2 Clock Configuration

24.2 Input/Output Pins

Table 24.3 shows the pin configuration

Table 24.3 Pin Configuration

Interface	name	I/O	Function
0, 1, 3	SSIBCK0, SSIBCK1, SSIBCK3	I/O	Bit clock
	SSILRCK0, SSILRCK1, SSILRCK3	I/O	LR clock/frame synchronization
	SSITxD0, SSITxD1, SSITxD3	Output	Serial data output
	SSIRxD0, SSIRxD1, SSIRxD3	Input	Serial data input
2	SSIBCK2	I/O	Bit clock
	SSILRCK2	I/O	LR clock/frame synchronization
	SSIDATA2	I/O	Serial data I/O
Common	AUDIO_CLK	Input	External clock for audio (The oversampling clock signal is input.)
	AUDIO_X1	Input	Audio clock (crystal oscillator or externally input clock) (The oversampling clock signal is input.)
	AUDIO_X2	Output	

24.3 List of Registers

Table 24.4 List of Control Registers

Channel	Register Name	Symbol	R/W	Initial Value	Address	Access Size
0	Control register	SSICR	RW	0000_0000H	E804_8000H	32 bits
	Status register Note: Bits 31, 30, and 25 to 0 are read only.	SSISR	RW	0200_0000H	E804_8004H	32 bits
	FIFO control register	SSIFCR	RW	0000_0000H	E804_8010H	32 bits
	FIFO status register Note: Bits 31 to 17 and 15 to 1 are read only.	SSIFSR	RW	0001_0000H	E804_8014H	32 bits
	Transmit FIFO data register	SSIFTDR	W	0000_0000H	E804_8018H	32 bits, 16 bits, 8 bits
	Receive FIFO data register	SSIFRDR	R	0000_0000H	E804_801CH	32 bits, 16 bits, 8 bits
	Audio format register	SSIOFR	RW	0000_0000H	E804_8020H	32 bits
	Status control register	SSISCR	RW	0000_0000H	E804_8024H	32 bits
1	Control register	SSICR	RW	0000_0000H	E804_8800H	32 bits
	Status register Note: Bits 31, 30, and 25 to 0 are read only.	SSISR	RW	0200_0000H	E804_8804H	32 bits
	FIFO control register	SSIFCR	RW	0000_0000H	E804_8810H	32 bits
	FIFO status register Note: Bits 31 to 17 and 15 to 1 are read only.	SSIFSR	RW	0001_0000H	E804_8814H	32 bits
	Transmit FIFO data register	SSIFTDR	W	0000_0000H	E804_8818H	32 bits, 16 bits, 8 bits
	Receive FIFO data register	SSIFRDR	R	0000_0000H	E804_881CH	32 bits, 16 bits, 8 bits
	Audio format register	SSIOFR	RW	0000_0000H	E804_8820H	32 bits
	Status control register	SSISCR	RW	0000_0000H	E804_8824H	32 bits
2	Control register	SSICR	RW	0000_0000H	E804_9000H	32 bits
	Status register Note: Bits 31, 30, and 25 to 0 are read only.	SSISR	RW	0200_0000H	E804_9004H	32 bits
	FIFO control register	SSIFCR	RW	0000_0000H	E804_9010H	32 bits
	FIFO status register Note: Bits 31 to 17 and 15 to 1 are read only.	SSIFSR	RW	0001_0000H	E804_9014H	32 bits
	Transmit FIFO data register	SSIFTDR	W	0000_0000H	E804_9018H	32 bits, 16 bits, 8 bits
	Receive FIFO data register	SSIFRDR	R	0000_0000H	E804_901CH	32 bits, 16 bits, 8 bits
	Audio format register	SSIOFR	RW	0000_0000H	E804_9020H	32 bits
	Status control register	SSISCR	RW	0000_0000H	E804_9024H	32 bits

Table 24.4 List of Control Registers

Channel	Register Name	Symbol	R/W	Initial Value	Address	Access Size
3	Control register	SSICR	RW	0000_0000H	E804_9800H	32 bits
	Status register Note: Bits 31, 30, and 25 to 0 are read only.	SSISR	RW	0200_0000H	E804_9804H	32 bits
	FIFO control register	SSIFCR	RW	0000_0000H	E804_9810H	32 bits
	FIFO status register Note: Bits 31 to 17 and 15 to 1 are read only.	SSIFSR	RW	0001_0000H	E804_9814H	32 bits
	Transmit FIFO data register	SSIFTDR	W	0000_0000H	E804_9818H	32 bits, 16 bits, 8 bits
	Receive FIFO data register	SSIFRDR	R	0000_0000H	E804_981CH	32 bits, 16 bits, 8 bits
	Audio format register	SSIOFR	RW	0000_0000H	E804_9820H	32 bits
	Status control register	SSISCR	RW	0000_0000H	E804_9824H	32 bits

Note: Access to the addresses other than those listed above is prohibited.

24.4 Function Details

24.4.1 Register Descriptions

24.4.1.1 Control Register (SSICR)

This is a 32-bit readable/writable register. With this register, select an audio clock, control IRQ, select data formats, and set an operation mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bits	Bit Name	R/W	Initial Value	Description
31	—	R	0	Reserved. Write 0. The read value is 0.
30	CKS	RW	0	Selects an audio clock for master-mode communication*1 0: Selects the AUDIO_CLK input 1: Selects the AUDIO_X input
29	TUIEN	RW	0	Transmit underflow interrupt output enable 0: Disables transmit underflow interrupt output 1: Enables transmit underflow interrupt output
28	TOIEN	RW	0	Transmit overflow interrupt output enable 0: Disables transmit overflow interrupt output 1: Enables transmit overflow interrupt output
27	RUIEN	RW	0	Receive underflow interrupt output enable 0: Disables receive underflow interrupt output 1: Enables receive underflow interrupt output
26	ROIEN	RW	0	Receive overflow interrupt output enable 0: Disables receive overflow interrupt output 1: Enables receive overflow interrupt output
25	IIEN	RW	0	Idle mode interrupt output enable 0: Disables idle mode interrupt output 1: Enables idle mode interrupt output
24	—	R	0	Reserved. Write 0. The read value is 0.
23, 22	FRM[1:0]	RW	00B	Select frame word number*1

	Communication format (SSIOFR.OMOD[1:0])		
FRM[1:0]	I²S (00B)	Monaural (10B)	TDM (01B)
00B	2	1	Set prohibition
01B	Set prohibition	Set prohibition	4
10B			6
11B			8

Bits	Bit Name	R/W	Initial Value	Description
21 to 19	DWL[2:0]	R/W	000B	Selects data word length*1 000: 8 bits 001: 16 bits 010: 18 bits 011: 20 bits 100: 22 bits 101: 24 bits 110: 32 bits 111: Setting prohibited
18 to 16	SWL[2:0]	RW	000B	Selects system word length*1 000: 8 bits 001: 16 bits 010: 24 bits 011: 32 bits 100: 48 bits 101: 64 bits 110: 128 bits 111: 256 bits
15	—	R	0	Reserved. Write 0. The read value is 0.
14	MST	RW	0	Master enable*1 0: Slave-mode communication 1: Master-mode communication
13	BCKP	RW	0	Selects bit clock polarity*1 0: SSILRCK/SSIFS and SSIFTxD/SSIFRxD change at a falling edge (sampling SSILRCK/SSIFS and SSIFRxD at a rising edge of SSIBCK). 1: SSILRCK/SSIFS and SSIFTxD/SSIFRxD change at a rising edge (sampling SSILRCK/SSIFS and SSIFRxD at a falling edge of SSIBCK).
12	LRCKP	RW	0	Selects the initial value and polarity of LRCK/FS*1 0: The initial value is at a high level For SSILRCK/SSIFS, one frame start trigger is falling edge of SSILRCK/SSIFS. 1: The initial value is at a low level For SSILRCK/SSIFS, one frame start trigger is rising edge of SSILRCK/SSIFS.
11	SPDP	RW	0	Selects serial padding polarity*1 0: Padding data is at a low level 1: Padding data is at a high level
10	SDTA	RW	0	Selects serial data alignment*1 0: Transmits and receives serial data first and then padding bits. 1: Transmit and receives padding bits first and then serial data.
9	PDTA	RW	0	Selects placement data alignment*1 0: Left-justifies placement data (SSIFTDR, SSIFRDR) 1: Right-justifies placement data (SSIFTDR, SSIFRDR)
8	DEL	RW	0	Selects serial data delay*1 0: Delay of one cycle of SSIBCK between SSILRCK/SSIFS and SSIFTxD/SSIFRxD. 1: No delay between SSILRCK/SSIFS and SSIFTxD/SSIFRxD. This bit control SSILRCK/SSIFS in monaural format (Refer to section 24.4.2.2, Monaural Format for details).

Bits	Bit Name	R/W	Initial Value	Description
7 to 4	CKDV[3:0]	RW	0H	Selects bit clock division ratio*1 CKDV[3:0] 0000: AUDIO_MCK 0001: AUDIO_MCK/2 0010: AUDIO_MCK/4 0011: AUDIO_MCK/8 0100: AUDIO_MCK/16 0101: AUDIO_MCK/32 0110: AUDIO_MCK/64 0111: AUDIO_MCK/128 1000: AUDIO_MCK/6 1001: AUDIO_MCK/12 1010: AUDIO_MCK/24 1011: AUDIO_MCK/48 1100: AUDIO_MCK/96 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited
3	MUEN	RW	0	Mute (silent) enable 0: Disables muting on the next frame boundary. 1: Enables muting on the next frame boundary.
2	—	R	0	Reserved. Write 0. The read value is 0.
1	TEN	RW	0	Transmission and reception enable*2 (TEN, REN): Operating 00: Disables transmission and reception 01: Enables reception (starts reception) 10: Enables transmission (starts transmission) 11: Enables transmission and reception (starts transmission and reception)
0	REN	RW	0	

Note 1. Writing to these bits while SSIF-2 is in a communication state (SSISR.IIRQ = 0) is prohibited. If written, the operation performed immediately after writing is not guaranteed.

Note 2. If the TEN bit or REN bit is rewritten, make sure that the SSISR.IIRQ bit is in the desired status. If written, the operation performed immediately after writing is not guaranteed. For example, after enabling operation, make sure that SSISR.IIRQ = 0 and after disabling operation, make sure that SSISR.IIRQ = 1.

- CKS Bit

This bit sets the audio clock in master-mode communication (MST = 1). In slave-mode communication (MST = 0), setting of this bit is invalid.

To write this bit, do so when AUDIO_MCK is not supplied. Refer to the AUCKE bit details explanation in section 24.4.1.3, FIFO Control Register (SSIFCR) for detailed timing.

- TUIEN Bit

This bit enables/disables output of transmit underflow interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.TUIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.TUIRQ = 1.

- TOIEN Bit

This bit enables/disables output of transmit overflow interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.TOIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.TOIRQ = 1.

- RUIEN Bit

This bit enables/disables output of receive underflow interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.RUIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.RUIRQ = 1.

- ROIEN Bit

This bit enables/disables output of receive overflow interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.ROIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.ROIRQ = 1.

- IIEN Bit

This bit enables/disables output of idle mode interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.IIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.IIRQ = 1.

- FRM[1:0] Bits

These bits set system word number in one frame of the communication formats.
Rewrite these bits when the LR clock supply to the terminal SSILRCK stops. Refer to the LRCONT bit details explanation in section 24.4.1.7, Audio Format Register (SSIOFR) for the output operation of the LR clock.

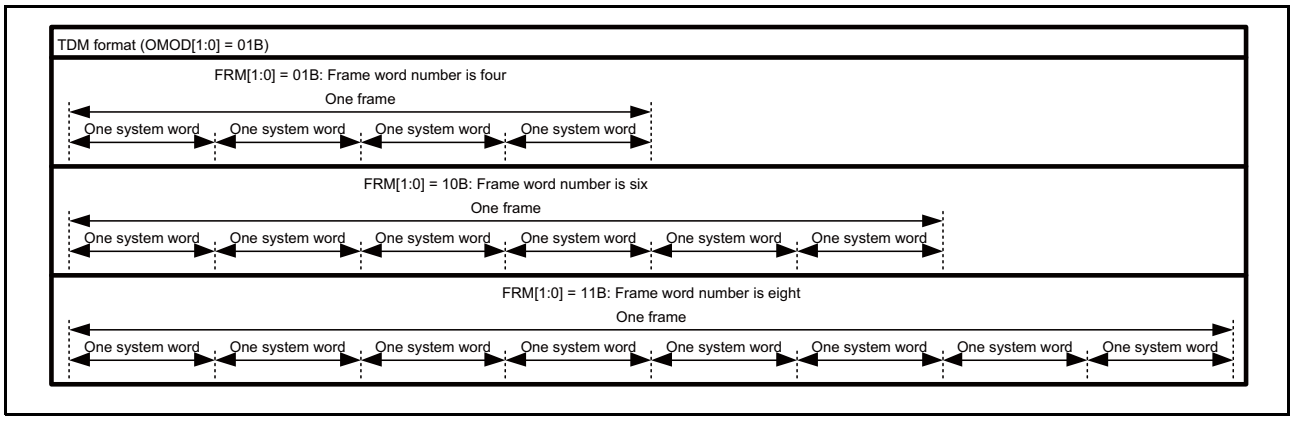


Figure 24.4 Frame word number

- DWL[2:0] Bits

These bits set the number of bits in one data word. The setting whose number of data of system word lengths is shorter than that of the data word length is a prohibition. Refer to Figure 24.13 in section 24.4.2, Communication Formats for details.

- SWL[2:0] Bits

These bits set the number of bits in one system word. Padding bits are sent and received in relation with one data word set with DWL[2:0]. See Table 24.13 in section 24.4.2, Communication Formats for details.
Rewrite these bits when the LR clock supply to the terminal SSILRCK stops. Refer to the LRCONT bit details explanation in section 24.4.1.7, Audio Format Register (SSIOFR) for the output operation of the LR clock.

- MST Bit

This bit sets master-/slave-mode communication.
To write this bit, do so when AUDIO_MCK is not supplied. Refer to the AUCKE bit details explanation in section 24.4.1.3, FIFO Control Register (SSIFCR) for detailed timing.

- BCKP Bit

This bit sets the bit clock polarity (Table 24.5).
To write this bit, do so when AUDIO_MCK is not supplied. Refer to the AUCKE bit details explanation in section 24.4.1.3, FIFO Control Register (SSIFCR) for detailed timing.

Table 24.5 Bit Clock Polarity

Communication	Master/Slave	Timing	BCKP = 0	BCKP = 1
Reception	Slave	At SSILRCK/SSIFS sampling	SSIBCK rising edge	SSIBCK falling edge
	Master/slave	At SSIFRxD sampling	SSIBCK rising edge	SSIBCK falling edge
Transmission	Master	At change of SSILRCK/SSIFS output	SSIBCK falling edge	SSIBCK rising edge
	Master/slave	At change of SSIFTxD output	SSIBCK falling edge	SSIBCK rising edge

- LRCKP Bit

This bit sets the initial value and polarity of SSILRCK/SSIFS. Set this bit according to the communication format to be used in SSIF-2 (Table 24.6). Only the start trigger is used at slave communication (MST=0).
Rewrite this bit when the LR clock supply to the terminal SSILRCK stops. Refer to the LRCONT bit details explanation in section 24.4.1.7, Audio Format Register (SSIOFR) for the output operation of the LR clock.

Table 24.6 Initial Output Value and Polarity of SSILRCK/SSIFS Pin

Communication Format	Expected Initial State	Setting Value of LRCKP
I²S	H	0
Monaural	L	1
TDM	L	1

Note: Do the setting that can be communicated respectively to I²S, monaural, and TDM at the compatible format.

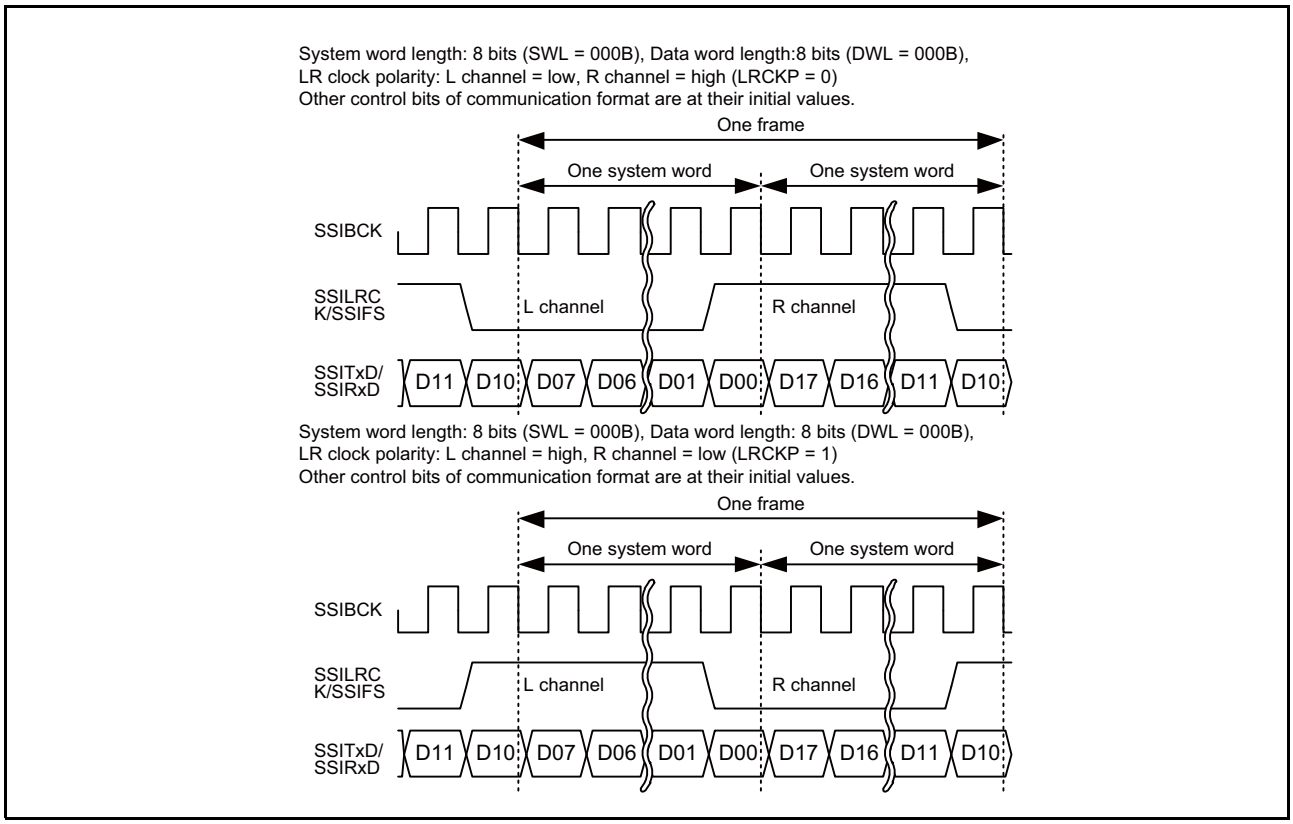


Figure 24.5 LRCK/FS Polarity Setting

- SPDP Bit

This bit sets polarity of padding bits.

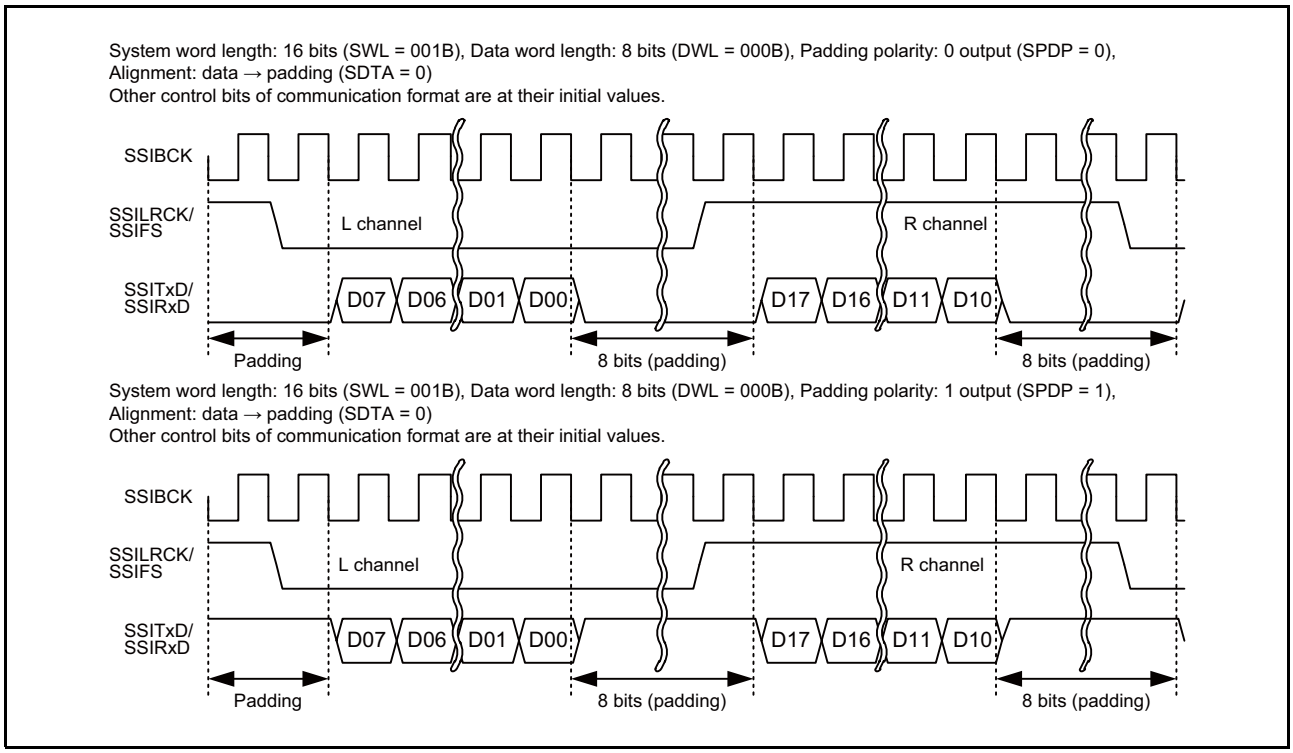


Figure 24.6 Padding Bit Polarity

- SDTA Bit

This bit sets alignment of serial data (data bits and padding bits). For communication without padding bits, this bit is invalid.

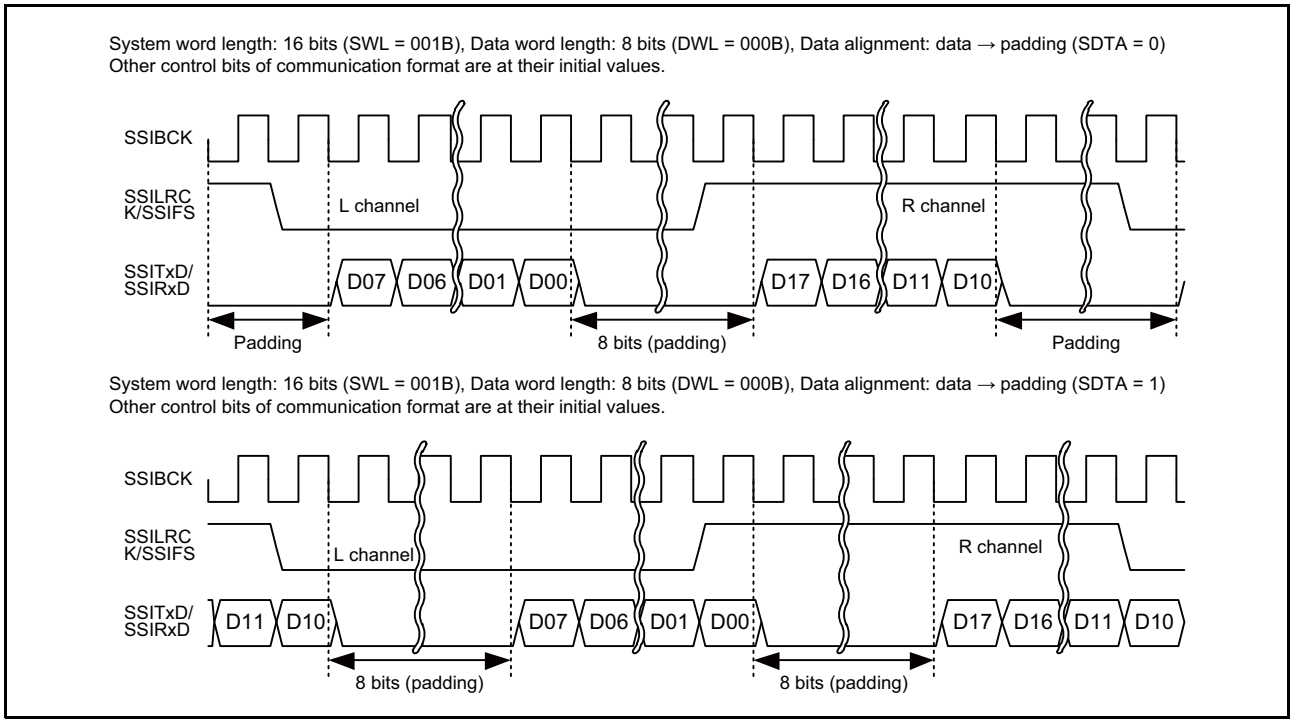


Figure 24.7 Alignment Setting of Serial Data with Padding Bits

- PDTA Bit

This bit sets alignment of placement data. With the setting of data word length as 32 bits (DWL[2:0] = 110B), this bit is invalid.

At transmission:

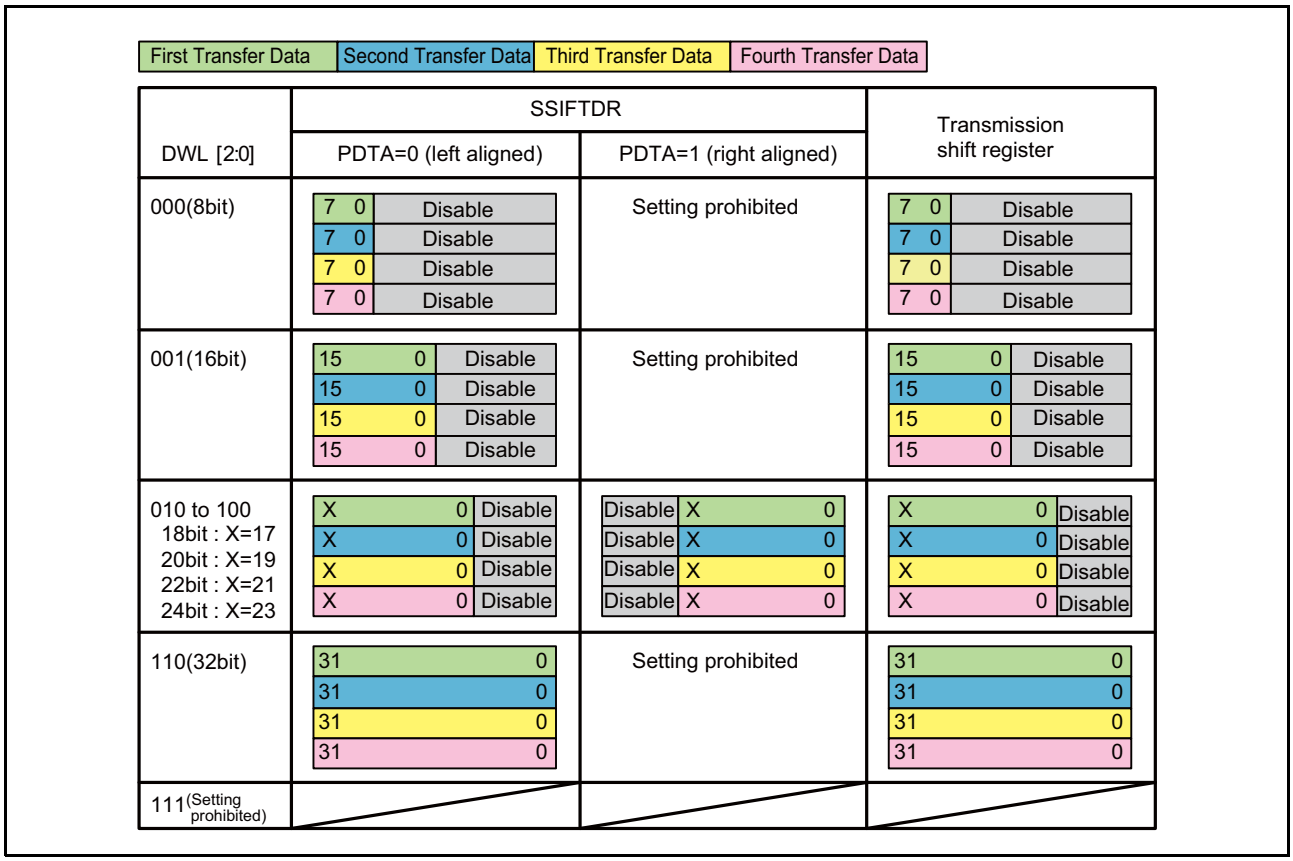


Figure 24.8 Alignment Setting of Placement Data in Transmission

At reception:

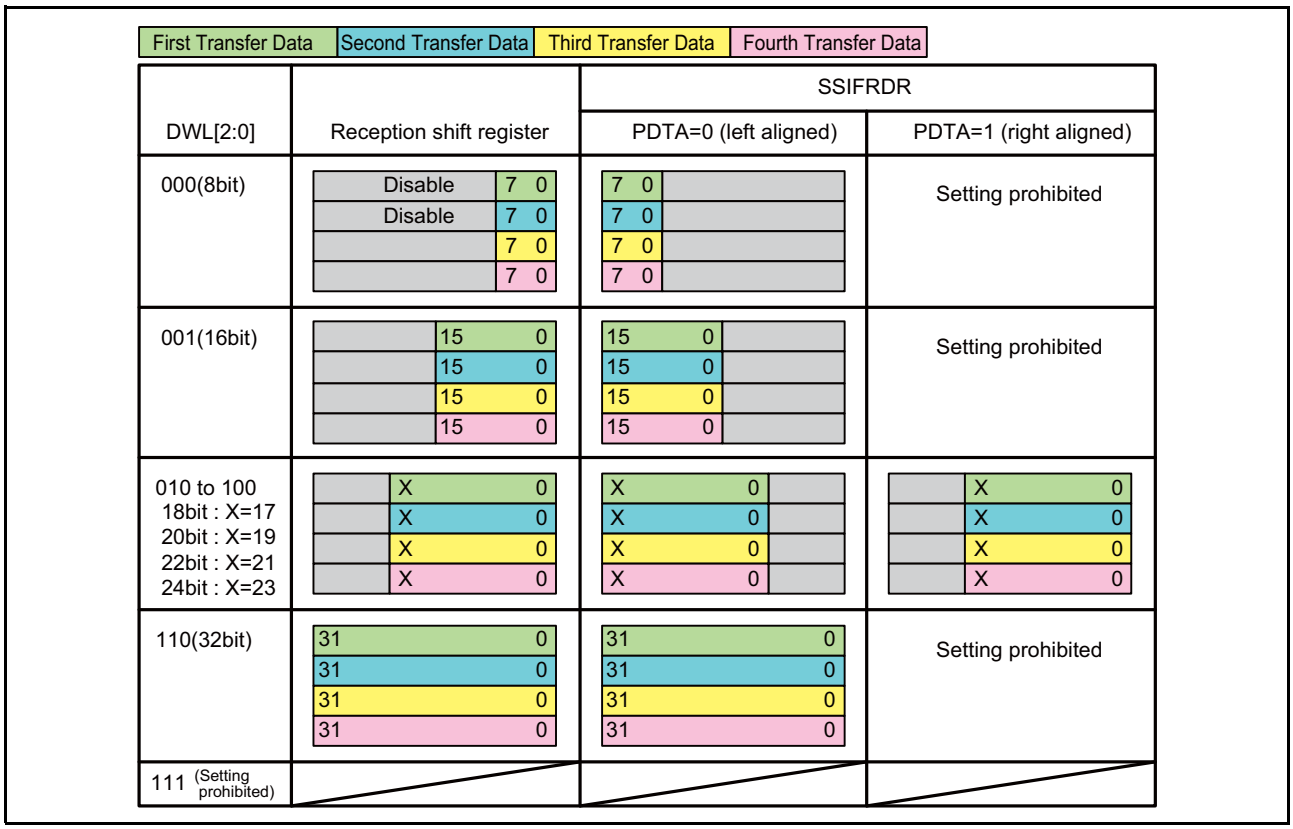


Figure 24.9 Alignment Setting of Placement Data in Reception

- DEL Bit

This bit sets whether or not there will be a delay between SSILRCK/SSIFS and SSITxD/SSIRxD.

With the I²S format and TDM format, set 0 to DEL (Figure 24.10). With the monaural format, this bit changes the high period width. Refer to section 24.4.2.2, Monaural Format for details. Do the setting that can be communicated at the compatible communication format.

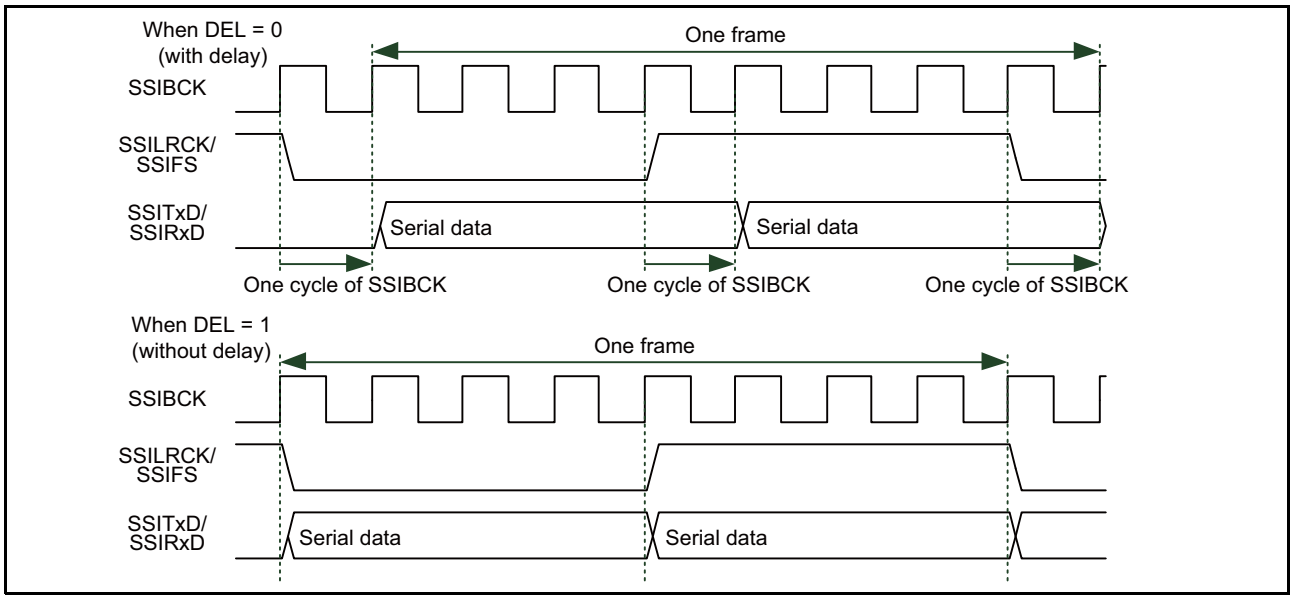


Figure 24.10 Setting of Delay in Serial Data

- CKDV[3:0] Bits

These bits set the division ratio of the bit clock based on AUDIO_MCK in master-mode communication (MST = 1) (Figure 24.11). In slave-mode communication (MST = 0), setting of these bits are invalid.
To write this bit, do so when AUDIO_MCK is not supplied. Refer to the AUCKE bit details explanation in section 24.4.1.3, FIFO Control Register (SSIFCR) for detailed timing.

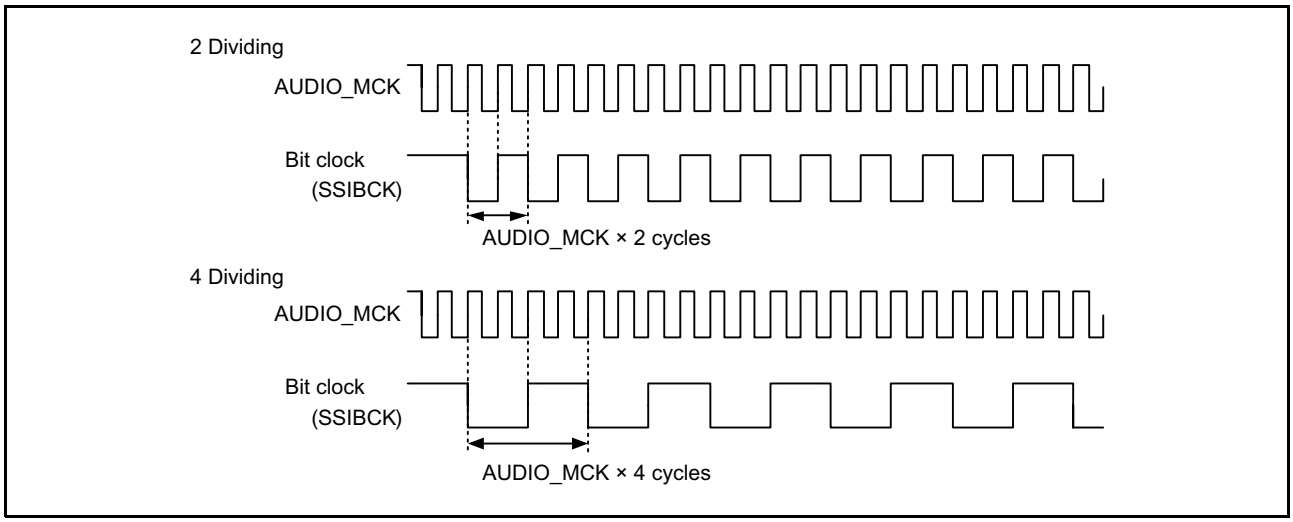


Figure 24.11 Sampling Frequencies in Master-mode Communication

- MUEN Bit

This bit sets/clears the mute function for the data output from the terminal SSITxD. When you set this bit to 1, the output value of SSITxD becomes 0 from the next frame boundary. When you set this bit to 0, the output value of SSITxD can be made the data of the FIFO transmission data register from the following frame boundary.
Note that this bit controls data only. Status flags and interrupt signals are normally generated.
Set this bit after setting completing the communication format.

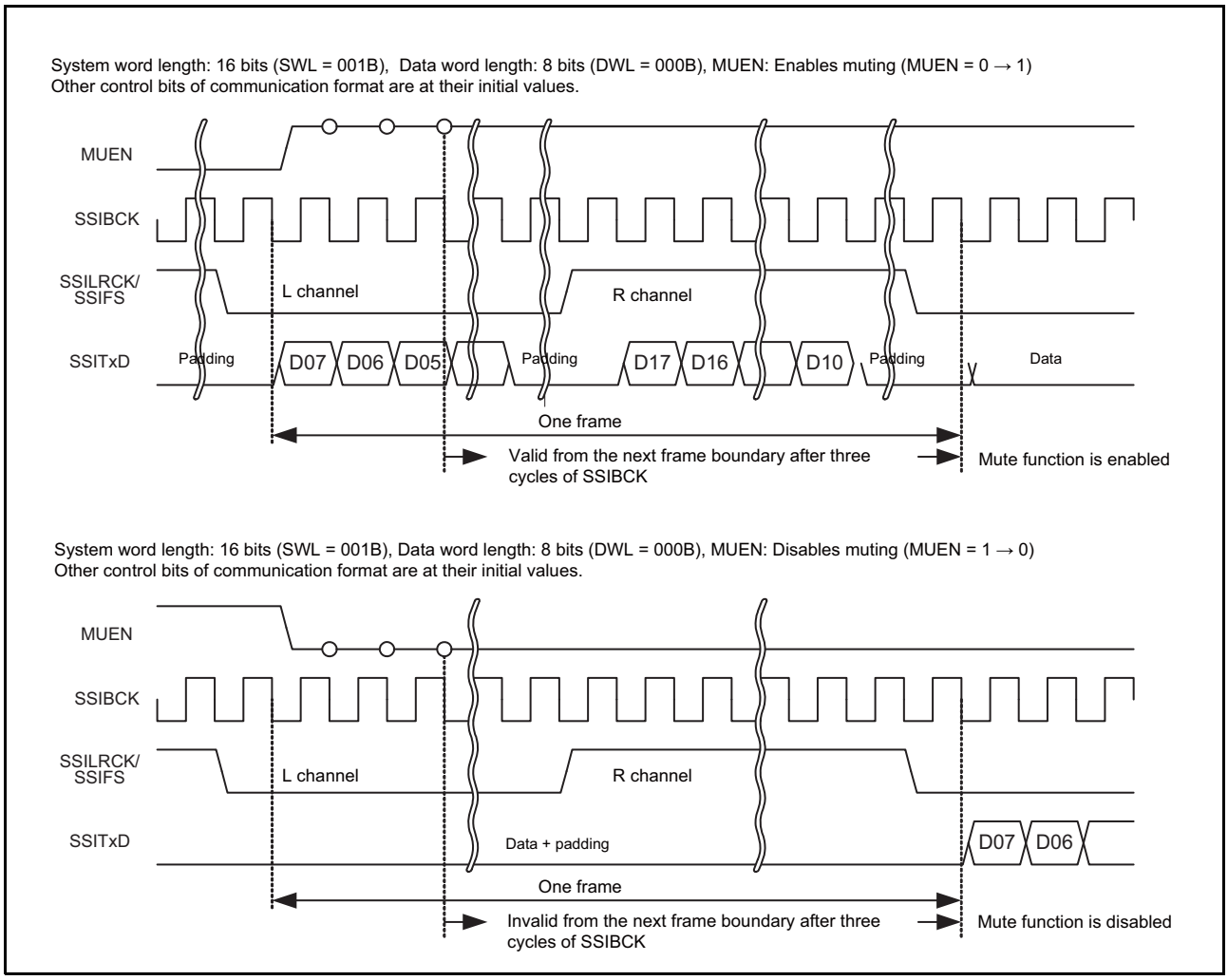


Figure 24.12 Transmit Data with the Mute Function Set

- TEN and REN Bits

These bits enable/disable transmission and reception. Writing 1 to these bits, transmission starts in synchronization with start trigger. Refer to section 24.5.2.2, Transmission, section 24.5.2.3, Reception, and section 24.5.2.4, Transmission and Reception for details. Writing 0 to these bits, transmission stops in synchronization with next frame boundary. Set 1 at the same time when you use SSIF-2 as transmission and reception. To stop communication, make sure to stop transmission and reception (TEN = 0 and REN = 0).
To stop communication without waiting for the frame boundary, follow the software reset procedure.

24.4.1.2 Status Register (SSISR)

This is a readable/writable 32-bit register. It is configured with status flags that indicate SSIF-2 operational state.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W0	R/W0	R/W0	R/W0	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	R/W	Initial Value	Description
31, 30	—	R	00B	Reserved. Write 0. The read value is 0.
29	TUIRQ	RW0	0	Transmit underflow error status flag 0: No transmit underflow error is generated. 1: A transmit underflow error is generated.
28	TOIRQ	RW0	0	Transmit overflow error status flag 0: No transmit overflow error is generated. 1: A transmit overflow error is generated.
27	RUIRQ	RW0	0	Receive underflow error status flag 0: No receive underflow error is generated. 1: A receive underflow error is generated.
26	ROIRQ	RW0	0	Receive overflow error status flag 0: No receive overflow error is generated. 1: A receive overflow error is generated.
25	IIRQ	R	1	Idle mode status flag 0: In the communication state 1: In the idle state
24 to 0	—	R	0000000H	Reserved. Write 0. The read value is 0.

- TUIRQ Bit

This is a status flag that indicates a transmit underflow error. This flag is set by automatic determination but it must be cleared by register access. This flag indicates that writing serial data necessary for one frame to SSIFTDR was slower than one frame of the transmission operation. Even if this flag is cleared after this flag becomes '1', the output value of SSIFTxD is 0. Follow the communication stop procedure (Figure 24.55) and the error procedure (Figure 24.56) to output the data written in transmission FIFO data register (SSIFTDR) to the terminal SSIFTxD. See section 24.5.2.6, Error Handling for the error recovery procedure. This flag is not cleared by transmission FIFO data register reset (SSIFCR.TFRST).

[Priority order for setting and clearing]
Setting is prioritized.*1

[Clearing condition]
When either the following is approved.
(1) Writing 0 to this flag after reading this flag as 1.*2
(2) When you do the communication permission (SSICR.TEN is changed from 0 to 1)

[Clearing timing]
Timing according to the above-mentioned clear condition.
(1) At completion of writing 0 to this flag after reading this flag as 1 (Figure 24.13).
(2) Write 1 in SSICR.TEN, complete, and after one cycle of P1φ (Figure 24.13).

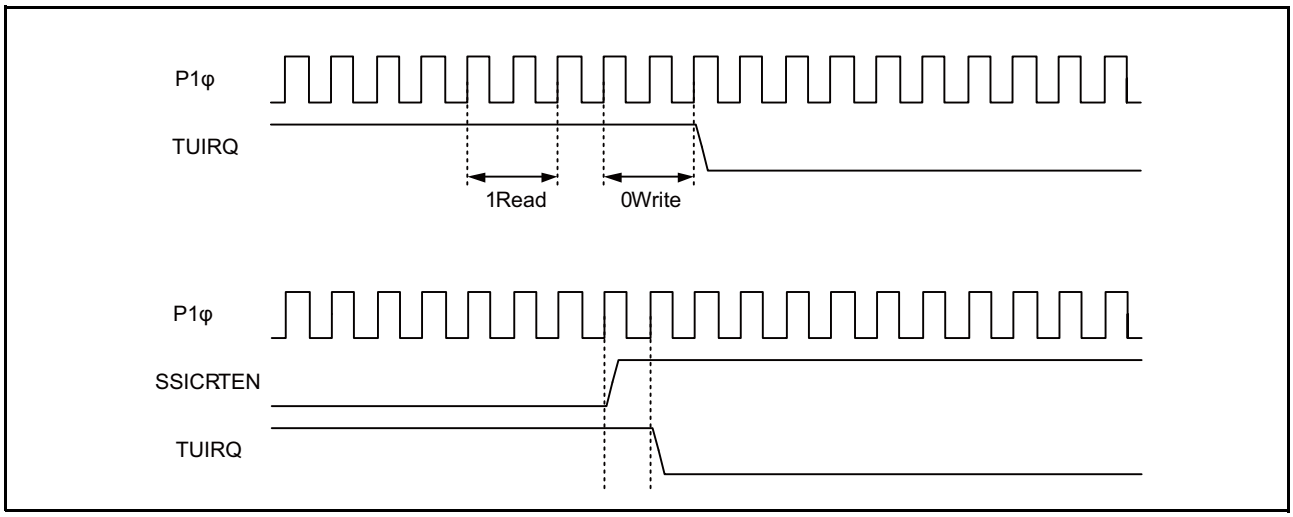


Figure 24.13 TUIRQ Clearing Timing

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1). The software reset has priority over all the clearing conditions described above.
Note 2. The state of reading this bit as one is cleared by any of the following three conditions being met:
• Software reset (SSIFCR.SSIRST=1).
• Completion of writing 0 to this flag after having read it as 1.
• One cycle of P1φ elapsing after the completion of writing 1 to SSICR.TEN.
Note 3. After the communication is permitted (SSICR.TEN is written and '1' is written from '0'), the transmission error flag (TOIRQ and TUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the transmission error flag might not be able to be read.

[Setting condition]
When data to be transmitted in the next frame is not fully written to SSIFTDR at the frame boundary of continuance communication.

[Setting timing]
After three cycles of P1φ from frame boundary (Figure 24.14).

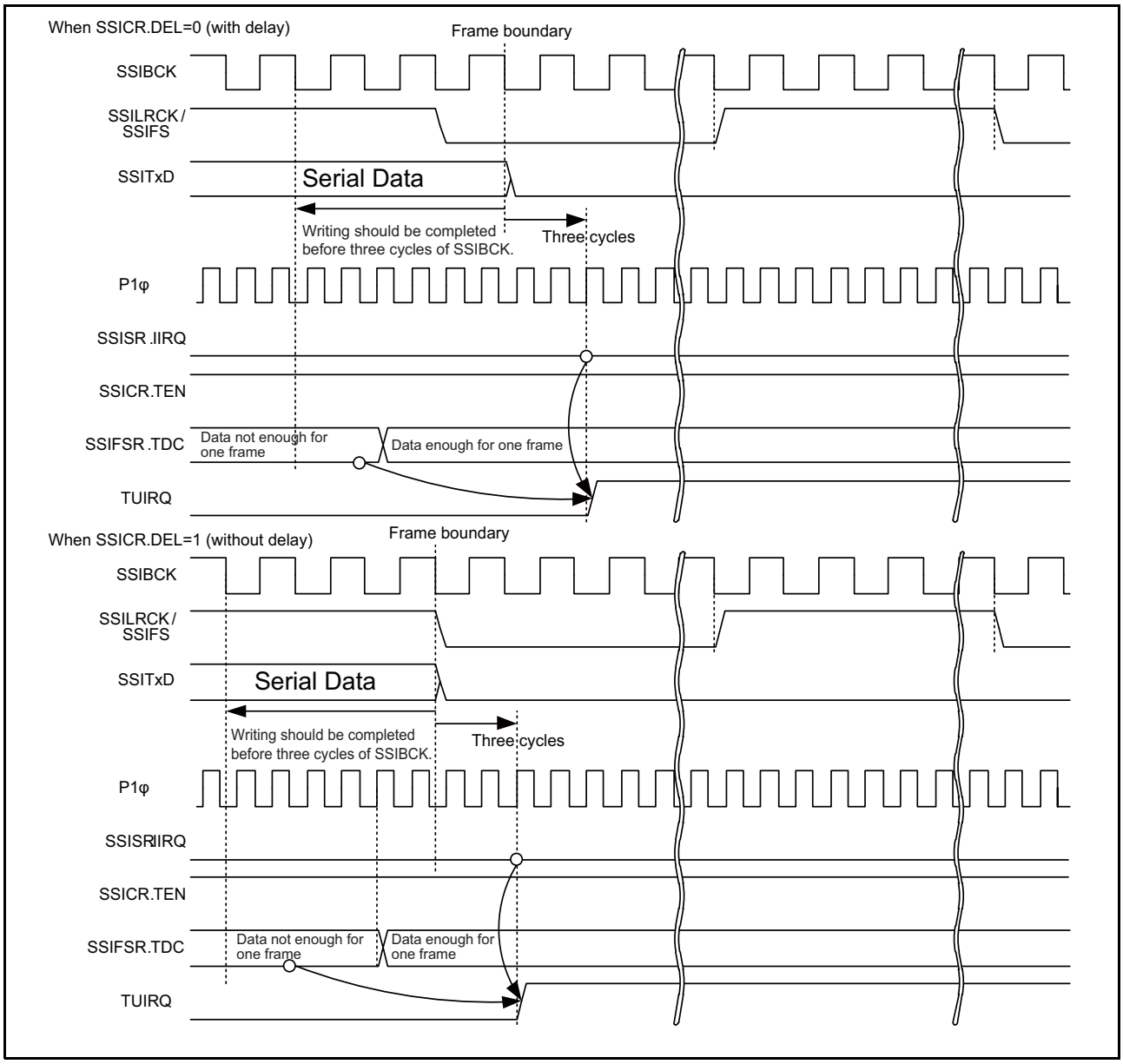


Figure 24.14 TUIRQ Setting Timing (When you continue communicating)

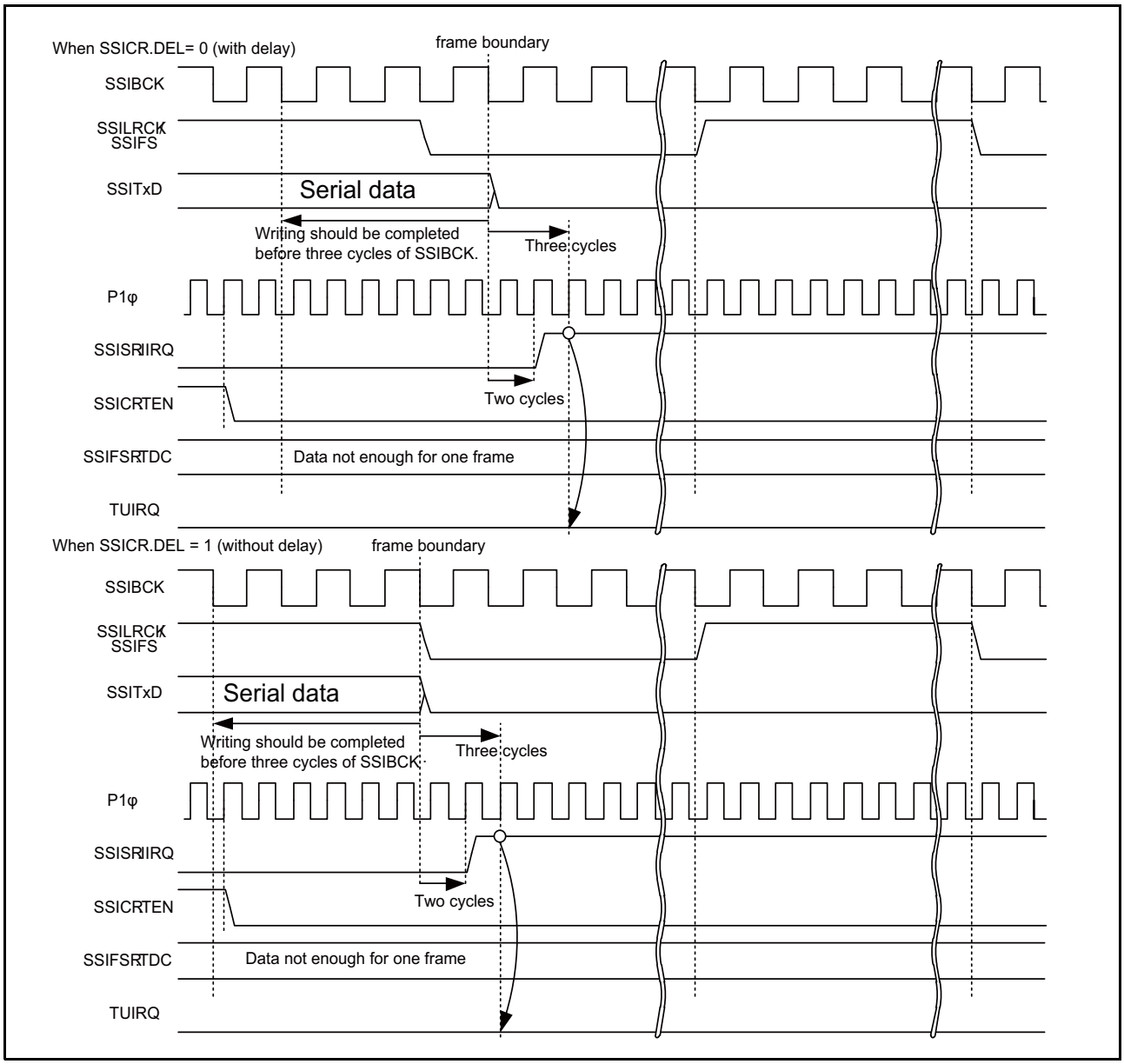


Figure 24.15 TUIRQ Setting Timing (When you stop communicating)

- TOIRQ Bit

This is a status flag that indicates a transmit overflow error. This flag is set by automatic determination but it must be cleared by register access. This flag indicates that serial data is written to SSIFTDR when it is full during transmission (SSICR.TEN=1). Data is not written to SSIFTDR where a transmit overflow error is generated. See section 24.5.2.6, Error Handling for the error recovery procedure. This flag is not cleared by transmission FIFO data register reset (SSIFCR.TFRST).

[Priority order for setting and clearing]

Setting is prioritized.*1

[Clearing condition]

When either the following is approved.

(1) Writing 0 to this flag after reading this flag as 1.*2

(2) When you do the communication permission (SSICR.TEN is changed from 0 to 1)

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this flag after reading this flag as 1 (same as Figure 24.13).

(2) Write 1 in SSICR.TEN, complete, and after one cycle of P1φ (same as Figure 24.13).

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1). The software reset has priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following three conditions being met:
- Software reset (SSIFCR.SSIRST=1).
- Completion of writing 0 to this flag after having read it as 1.
- One cycle of P1φ elapsing after the completion of writing 1 to SSICR.TEN.

Note 3. After the communication is permitted (SSICR.TEN is written and '1' is written from '0'), the transmission error flag (TOIRQ and TUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the transmission error flag might not be able to be read.

[Setting condition]

During transmission operation (SSICR.TEN = 1), data is written to SSIFTDR while it is full.

[Setting timing]

At completion of writing to SSIFTDR (Figure 24.16)

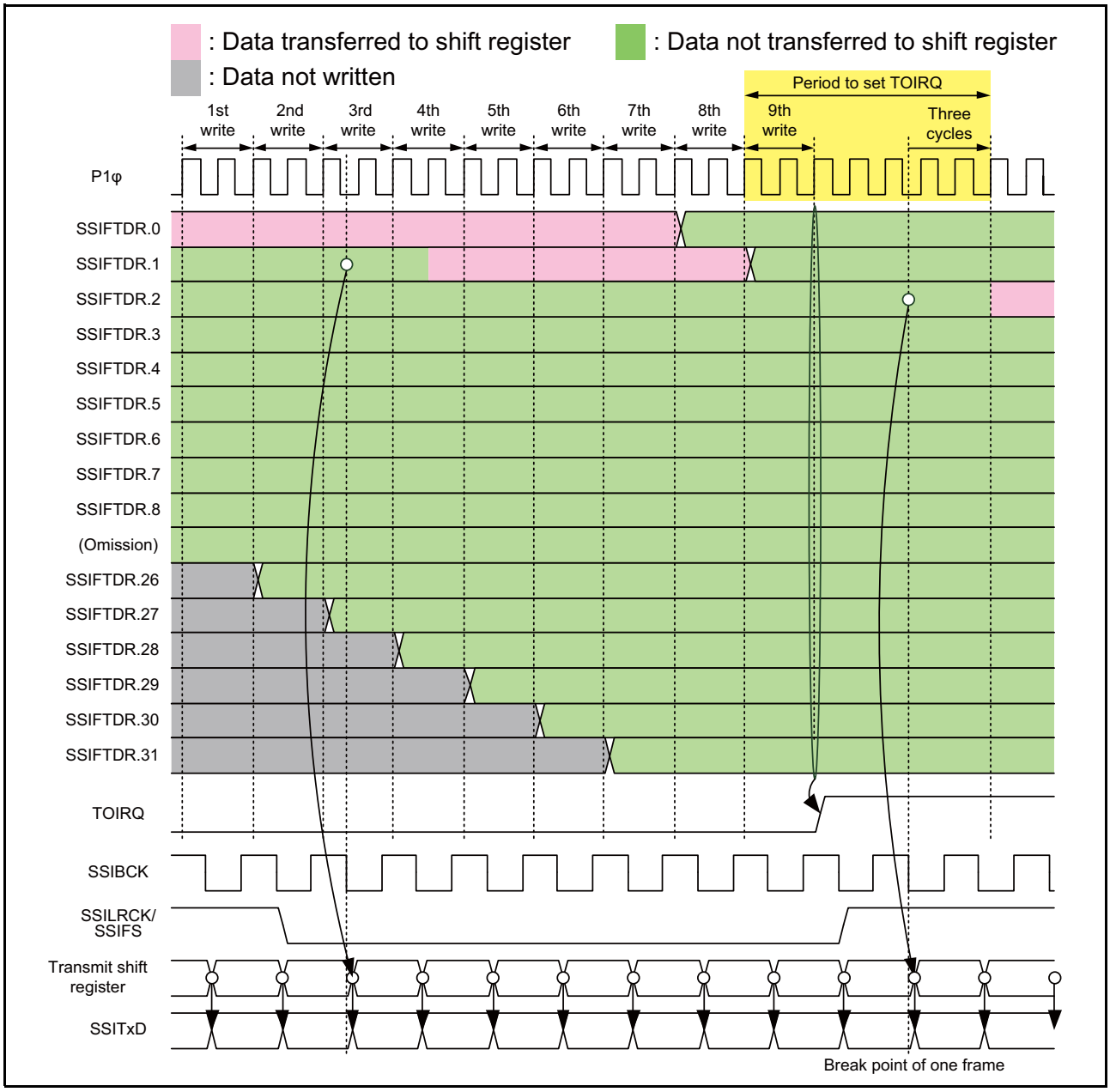


Figure 24.16 TOIRQ Setting Timing

- RUIRQ Bit

This is a status flag that indicates a receive underflow error. This flags is set by automatic determination but it must be cleared by register access. This flag indicates that SSIFRDR is read while it is empty. Data read from SSIFRDR where a receive underflow error is generated is invalid. See section 24.5.2.6, Error Handling for the error recovery procedure. This flag is not cleared by reception FIFO data register reset (SSIFCR.RFRST). Even if SSIFRDR is read to (SSIFCR.RFRST=1) while resetting reception FIFO data register, it doesn't set 1.

[Priority order for setting and clearing]
Setting is prioritized.*1
[Clearing condition]
When either the following is approved.
(1) Writing 0 to this flag after reading this flag as 1.*2
(2) When you do the communication permission (SSICR.REN is changed from 0 to 1)

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this flag after reading this flag as 1 (same as Figure 24.13).

(2) Write 1 in SSICR.REN, complete, and after one cycle of P1φ (same as Figure 24.13).

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1). The software reset has priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following three conditions being met:
- Software reset (SSIFCR.SSIRST=1).
- Completion of writing 0 to this flag after having read it as 1.
- One cycle of P1φ elapsing after the completion of writing 1 to SSICR.TEN.

Note 3. After the communication is permitted (SSICR.REN is written and '1' is written from '0'), the transmission error flag (ROIRQ and RUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the reception error flag might not be able to be read.

[Setting condition]

Reading from SSIFRDR while it is empty.

[Setting timing]

At completion of reading from SSIFRDR (Figure 24.17).

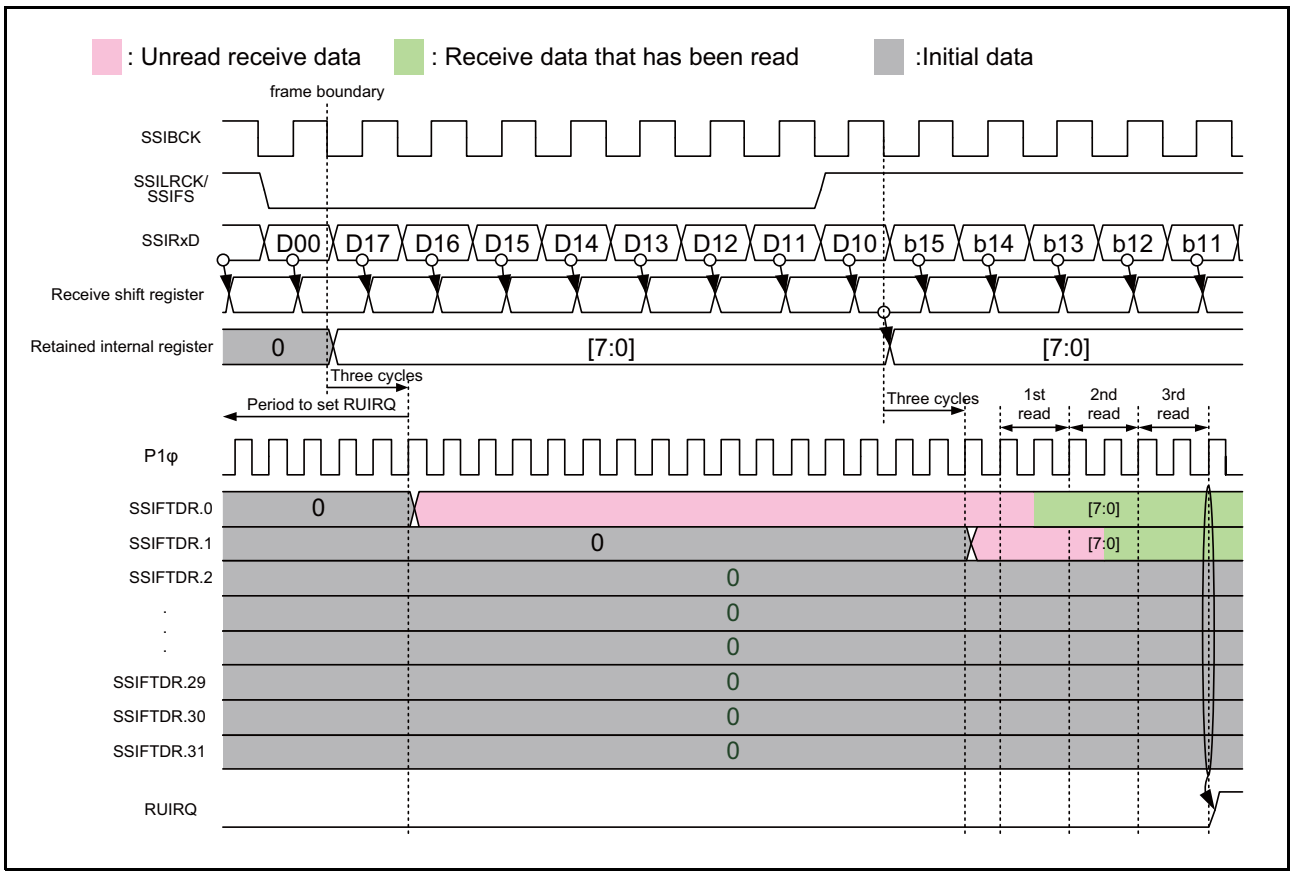


Figure 24.17 RUIRQ Setting Timing

- ROIRQ Bit

This is a status flag that indicates a receive overflow error. This flags is set by automatic determination but it must be cleared by register access. This flag indicates that received data is supplied at a higher rate than requested. Data is not transferred from the receive shift register to SSIFRDR where a receive overflow error is generated. See section 24.5.2.6, Error Handling for the error recovery procedure. This flag is not cleared by reception FIFO data register reset (SSIFCR.RFRST).

[Priority order for setting and clearing]
Setting is prioritized.
[Clearing condition]
Writing 0 to this flag after reading this flag as 1.

[Clearing timing]
At completion of writing 0 after reading 1 from an SFR (same as Figure 24.13).
[Priority order for setting and clearing]
Setting is prioritized.*1
[Clearing condition]
When either the following is approved.
(1) Writing 0 to this flag after reading this flag as 1.*2
(2) When you do the communication permission (SSICR.REN is changed from 0 to 1)

[Clearing timing]
Timing according to the above-mentioned clear condition.
(1) At completion of writing 0 to this flag after reading this flag as 1 (same as Figure 24.13).
(2) Write 1 in SSICR.REN, complete, and after one cycle of P1φ (same as Figure 24.13).

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1). The software reset has priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following three conditions being met:
- Software reset (SSIFCR.SSIRST=1).
- Completion of writing 0 to this flag after having read it as 1.
- One cycle of P1φ elapsing after the completion of writing 1 to SSICR.TEN.

Note 3. After the communication is permitted (SSICR.REN is written and '1' is written from '0'), the transmission error flag (ROIRQ and RUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the reception error flag might not be able to be read.

[Setting condition]
At completion of receiving new data while SSIFRDR is full.
[Setting timing]
Three cycles of P1φ after reception is completed (Figure 24.18).

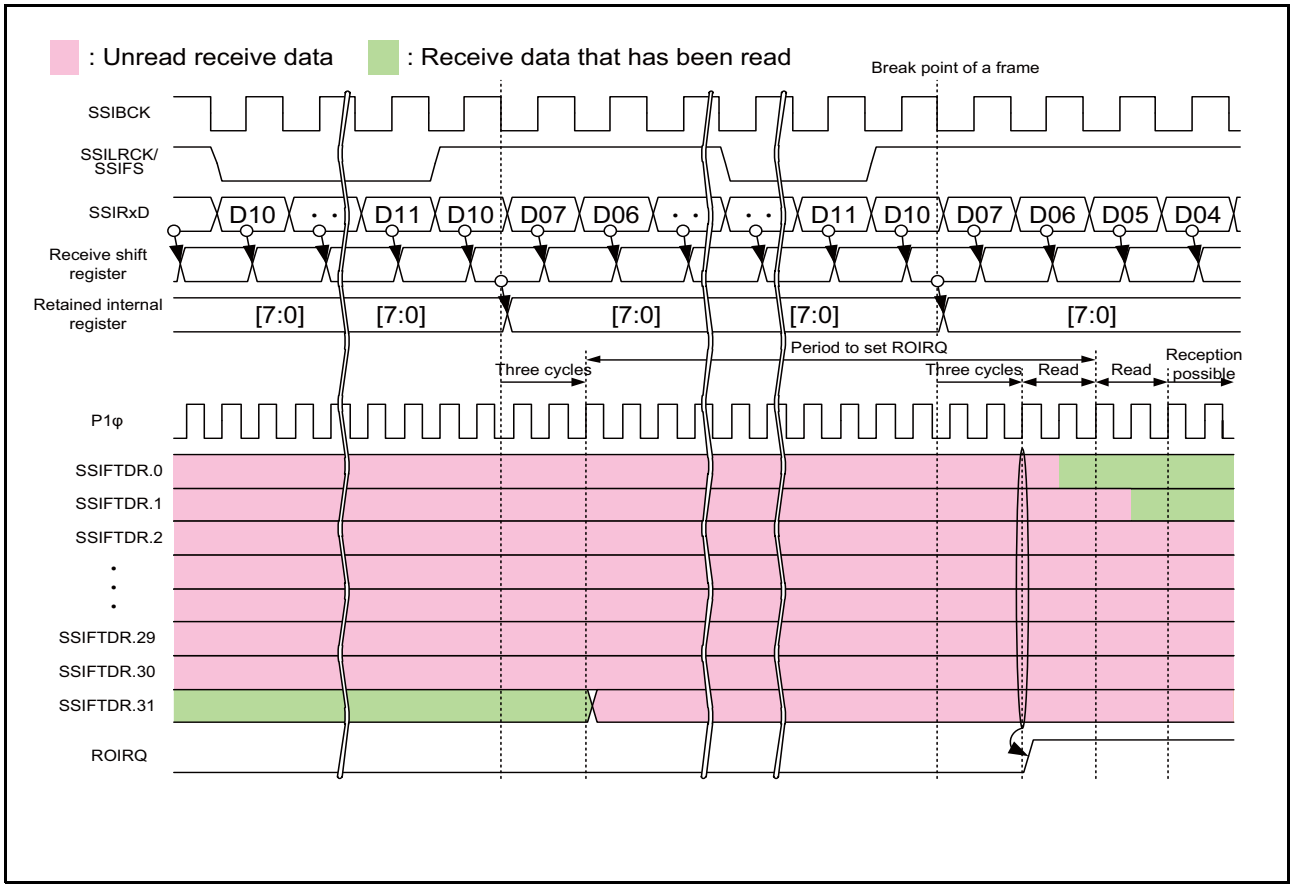


Figure 24.18 ROIRQ Setting Timing

- IIRQ Bit

This is a status flag that indicates the idle state. It indicates whether SSIF-2 is in the idle state or communication state (Figure 24.19 and Figure 24.20).

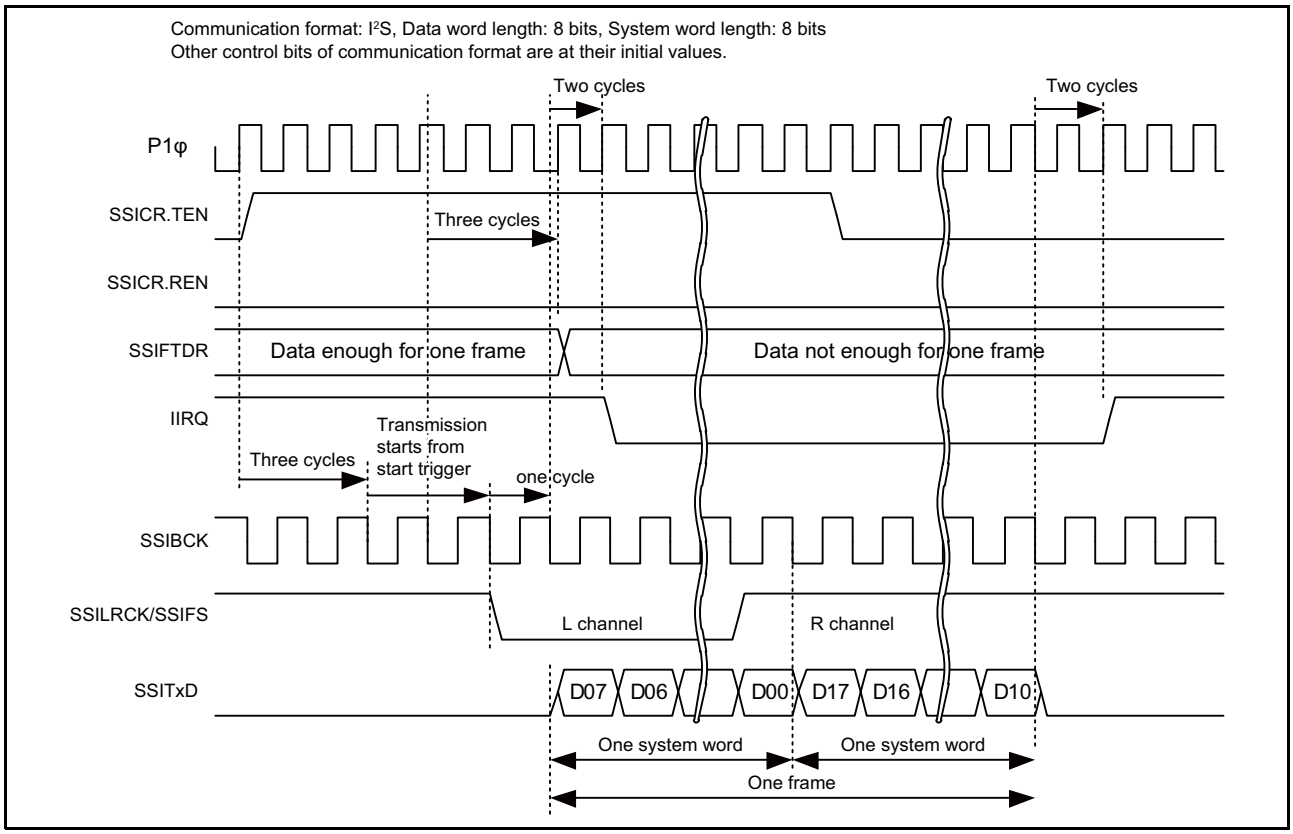


Figure 24.19 IIRQ Setting Timing (Transmission)

- Transmitter (dedicated to transmission)

[Clearing condition]

While transmission is enabled (SSICR.TEN = 1 and SSICR.REN = 0), one frame of data to be transmitted has been written to SSIFTDR, and a start trigger is generated in SSILRCK/SSIFS.

[Clearing timing]

Two cycles of P1φ after one cycle of SSIBCK where a start trigger is generated.

[Setting condition]

One frame of data has been transmitted while transmission and reception are disabled (SSICR.TEN = 0 and SSICR.REN = 0).

[Setting timing]

Two cycles of P1φ after transmission is complete according to the setting condition (frame boundary).

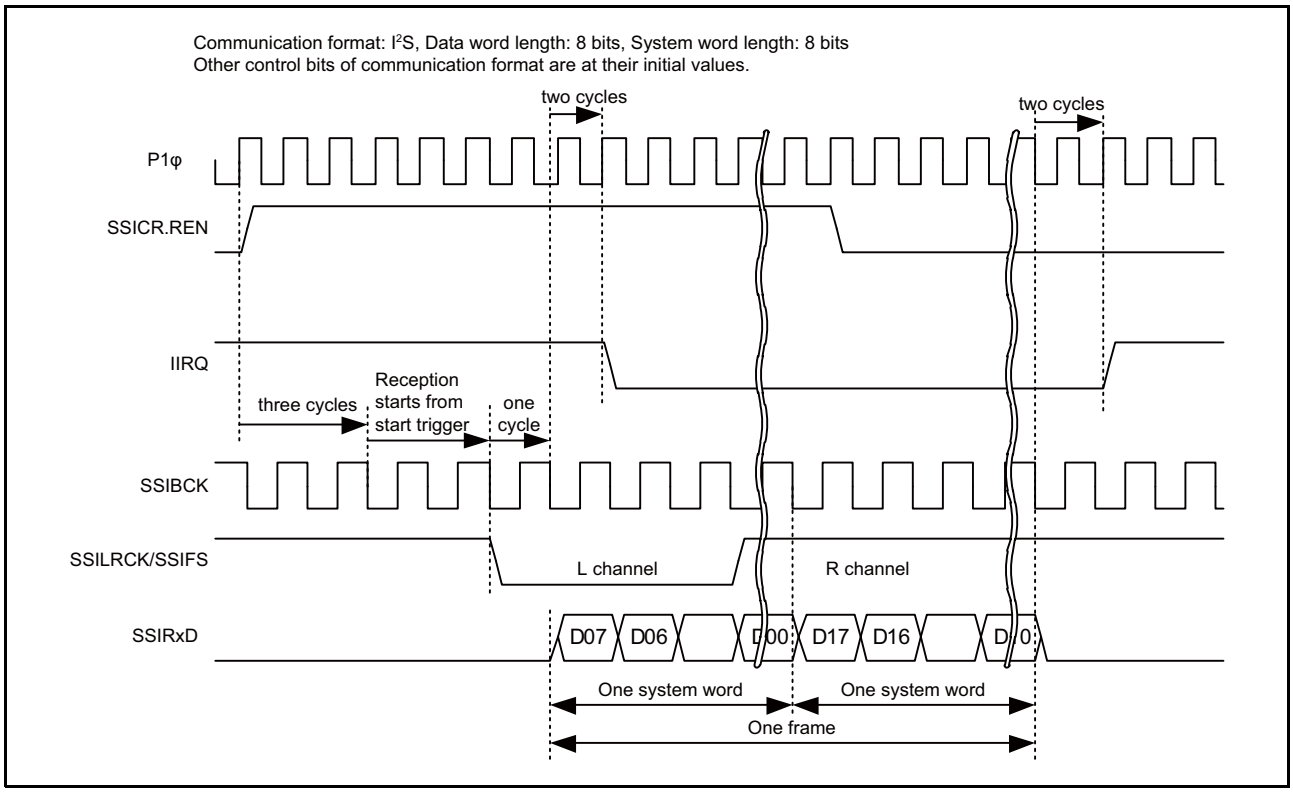


Figure 24.20 IIRQ Setting Timing (Reception)

- Receiver (dedicated to reception)

[Clearing condition]
While reception is enabled (SSICR.TEN = 0 and SSICR.REN = 1), and a start trigger is generated in SSILRCK/SSIFS.
[Clearing timing]
Two cycles of P1φ after one cycle of SSIBCK when a start trigger of the clearing condition is generated.
[Setting condition]
One frame of data has been received while transmission and reception are disabled (SSICR.TEN = 0 and SSICR.REN = 0).
[Setting timing]
Two cycles of P1φ after reception of the setting condition is completed (frame boundary).

- Transceiver (transmission and reception)

[Clearing condition]
While transmission and reception are enabled (SSICR.TEN = 1 and SSICR.REN = 1), one frame of data to be transmitted has been written to SSIFTDR, and a start trigger is generated in SSILRCK/SSIFS.
[Clearing timing]
Two cycles of P1φ after one cycle of SSIBCK when a start trigger of the clearing condition is generated.
[Setting condition]
One frame of data has been transmitted while transmission and reception are disabled (SSICR.TEN = 0 and SSICR.REN = 0).
[Setting timing]
Two cycles of P1φ after transmission is completed according to the setting condition (frame boundary).

24.4.1.3 FIFO Control Register (SSIFCR)

This is a readable/writable 32-bit register. It sets a software reset, byte swap, noise canceller, and enable/disable of interrupt requests.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	BSW	BCK NCE	LRCK NCE	RXD NCE	—	—	—	—	TIE	RIE	TFRST	RFRST
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	R/W	Initial Value	Description
31	AUCKE	RW	0	AUDIO_MCK Enable in Master-mode Communication*1 0: Disables supply of AUDIO_MCK 1: Enables supply of AUDIO_MCK
30 to 17	—	R	0000H	Reserved. Write 0. The read value is 0.
16	SSIRST	RW	0	Software Reset 0: Clears a software reset condition 1: Sets a software reset condition
15 to 12	—	R	00H	Reserved. Write 0. The read value is 0.
11	BSW	RW	0	Byte Swap Enable*1 0: Disables byte swap 1: Enables byte swap
10	BCKNCE	RW	0	Noise Canceller Enable in Slave-mode Communication (SSIBCK)*1*2 0: Disables (bypasses) the noise canceller 1: Enables the noise canceller
9	LRCKNCE	RW	0	Noise Canceller Enable in Slave-mode Communication (SSILRCK/SSIFS)*1*2 0: Disables (bypasses) the noise canceller 1: Enables the noise canceller
8	RXDNCE	RW	0	Received Data Input Noise Canceller Enable in Slave-mode Communication (SSIRxD)*1*2 0: Disables (bypasses) the noise canceller 1: Enables the noise canceller
7 to 4	—	R	00H	Reserved. Write 0. The read value is 0.
3	TIE	RW	0	Transmit Data Empty Interrupt Output Enable 0: Disables transmit data empty interrupts 1: Enables transmit data empty interrupts
2	RIE	RW	0	Receive Data Full Interrupt Output Enable 0: Disables receive data full interrupts 1: Enables receive data full interrupts
1	TFRST	RW	0	Transmit FIFO data register reset*1 0: Clears a transmit data FIFO reset condition 1: Sets a transmit data FIFO reset condition
0	RFRST	RW	0	Receive FIFO data register reset*1 0: Clears a receive data FIFO reset condition 1: Sets a receive data FIFO reset condition

Note 1. Writing to these bits while SSIF-2 is in a communication state (SSISR.IIRQ = 0) is prohibited. If written, the operation performed immediately after writing is not guaranteed.

Note 2. This bit can only be set to 1 in slave-mode communication (SSICR.MST = '0') except in the case of transmission (SSICR.TEN = '1') while SSICR.DEL = '1'. Be sure to clear this bit to 0 in master-mode communication (SSICR.MST = '1') or during transmission in slave mode (SSICR.MST = '0', SSICR.TEN = '1') while SSICR.DEL = '1'. Set the BCKNCE, LRCKNCE, and RXDNCE bits to the same value.

- AUCKE Bit

This bit enables/disables supply to AUDIO_MCK while in master-mode communication (MST = 1). Select AUDIO_MCK (SSICR.CKS bit) before writing 1 to this bit.
Rewrite this bit after completing the setting (CKS, MST, BCKP, and CKDV of the SSICR register) that relates to AUDIO_MCK.

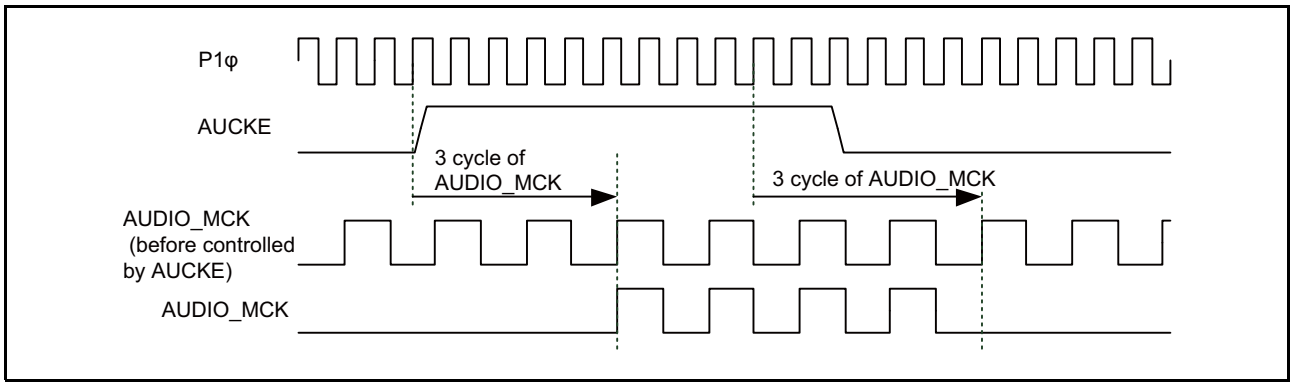


Figure 24.21 Stop/Resume of AUDIO_MCK

Caution: In slave-mode communication (SSICR.MST = 0), SSIF-2 needs supply of SSIBCK. To stop BCK on the master side, make sure that SSIF-2 is in the idle state (SSISR.IIRQ = 1). If BCK is stopped before SSIF-2 becomes idle, take the procedure to start communication (Figure 24.52) or wait for an idle state by taking the procedure to resume communication (Figure 24.57).

Caution: In master-mode communication (SSICR.MST = 1), SSIF-2 operates with the audio clock (AUDIO_MCK). To stop SSIF-2 completely, make sure that SSIF-2 is in the idle state (SSISR.IIRQ = 1) and then write 0 to SSIFCR.ADCKE. If 0 is written to SSIFCR.ADCKE before SSIF-2 becomes idle, take the procedure to start communication (Figure 24.52) or wait for the idle state by taking the procedure to resume communication (Figure 24.57).

Figure 24.22 and Figure 24.23 show timing until outputting it to the terminal SSIBCK after this bit is set to '1'.

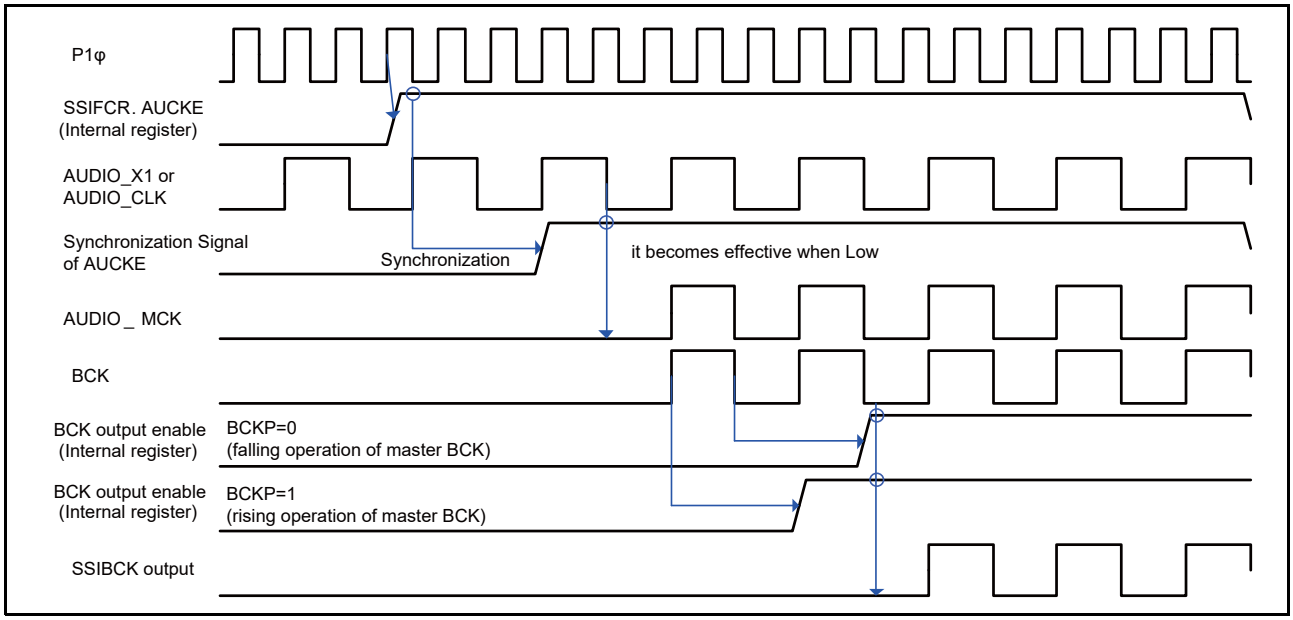


Figure 24.22 Timing chart when it begins to communicate master from system reset

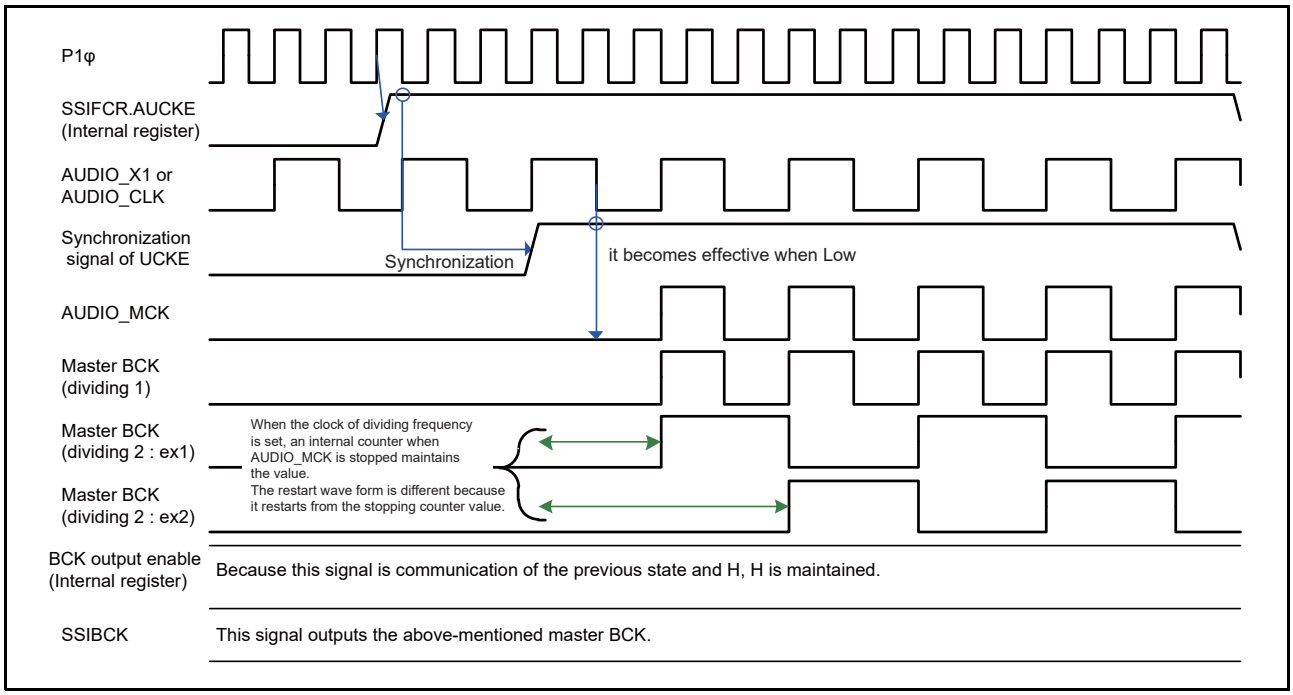


Figure 24.23 Timing chart when it begins to communicate master from communication stop

Note: When the supply of AUDIO_MCK is stopped, the terminal SSIBCK maintains the value. The terminal SSIBCK might stop in the state of "H".

- SSIRST Bit

This bit sets a software reset of SSIF-2. Writing 1 to this bit initializes the internal state of SSIF-2. After a reset by writing 1, write 0 to release the reset because this bit is not automatically cleared to 0. After writing 0 to this bit, make sure that the bit becomes 0 before the next procedure.
To stop communication of SSIF-2 immediately, after turning off the peripheral functions, write 1 to this bit. Initialization by a software reset is performed without any relation with the bit clock.

Table 24.7 Bits Initialized by Software Reset of the SSIFCR.SSIRST Bit

Symbol	Address (Base+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	00h	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	04h	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	10h	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	BCKNCE	LRCKNCE	RxDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	14h	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	18h	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	1ch	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	20h	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	24h	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

- BSW Bit

This bit enables/disables byte swap of register access for the transmit FIFO data register (SSIFTDR) and the receive FIFO data register (SSIFRDR). This bit is valid only with 16-bit access or 32-bit access to SSIFTDR and SSIFRDR (Figure 24.24).

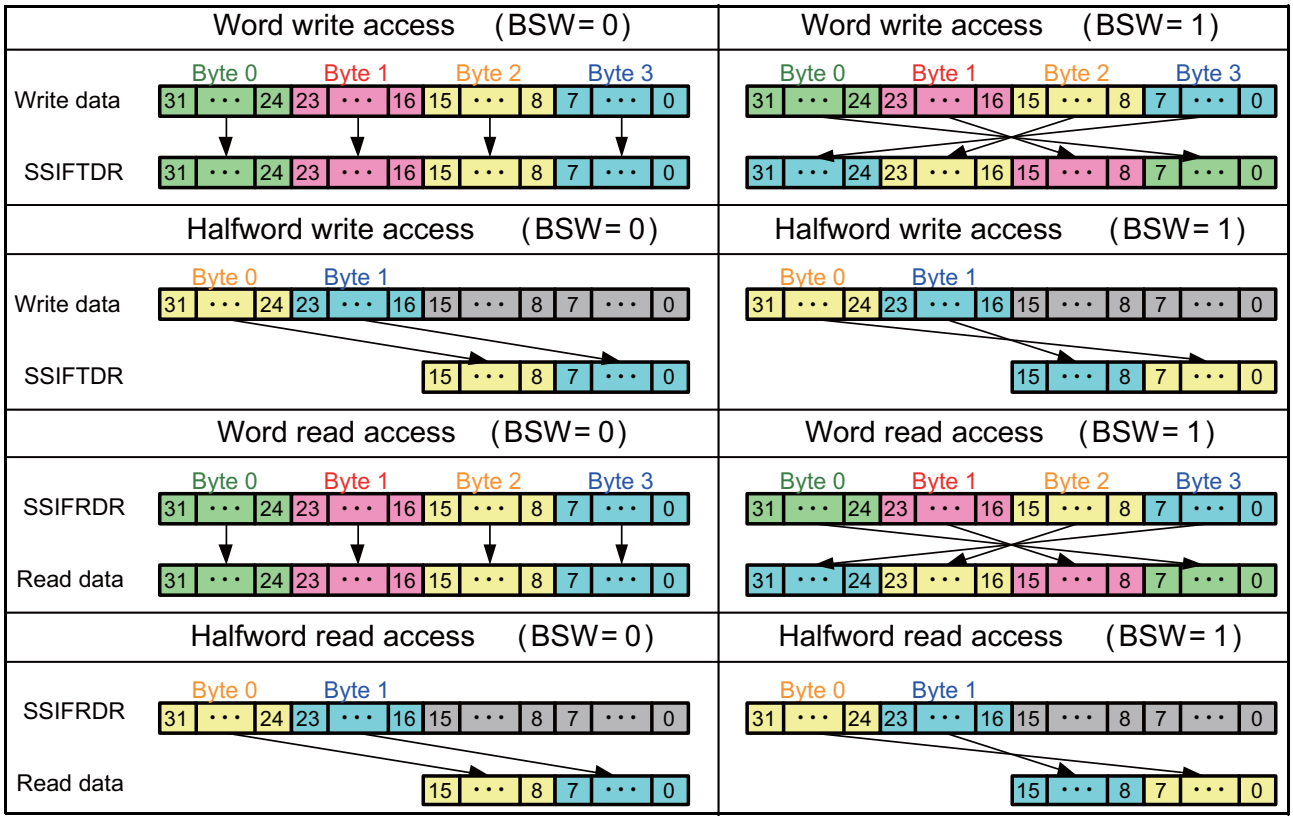


Figure 24.24 Operation Example of Byte Swap

- BCKNCE Bit

This bit enables or disables the noise cancellation function supporting the suppression of transients 15-ns in length (typ.) in input signals on the SSIBCK pin in slave-mode communication (SSICR.MST = '0') except in the case of transmission (SSICR.TEN = '1') while SSICR.DEL = '1'. The setting of this bit in master-mode communication (SSICR.MST = '1') is disabled.

Be sure to clear this bit to 0 in master-mode communication or during transmission in slave mode (SSICR.MST = '0', SSICR.TEN = '1') while SSICR.DEL = '1'. Set the BCKNCE, LRCKNCE, and RXDNCE bits to the same value.

- LRCKNCE Bit

This bit enables or disables the noise cancellation function supporting the suppression of transients 15-ns in length (typ.) in input signals on the SSILRCK and SSIFS pins in slave-mode communication (SSICR.MST = '0') except in the case of transmission (SSICR.TEN = '1') while SSICR.DEL = '1'. The setting of this bit in master-mode communication (SSICR.MST = '1') is disabled.

Be sure to clear this bit to 0 in master-mode communication or during transmission in slave mode (SSICR.MST = '0', SSICR.TEN = '1') while SSICR.DEL = '1'. Set the BCKNCE, LRCKNCE, and RXDNCE bits to the same value.

- RXDNCE Bit

This bit enables or disables the noise cancellation function supporting the suppression of transients 15-ns in length (typ.) in input signals on the SSIRxD pin in slave-mode communication (SSICR.MST = '0') except in the case of transmission (SSICR.TEN = '1') while SSICR.DEL = '1' and while reception is enabled (SSICR.REN = '1').

The setting of this bit in master-mode communication (SSICR.MST = '1') or while reception is disabled (SSICR.REN = '0') is disabled.
Be sure to clear this bit to 0 in master-mode communication or during transmission in slave mode (SSICR.MST = '0', SSICR.TEN = '1') while SSICR.DEL = '1'. Set the BCKNCE, LRCKNCE, and RXDNCE bits to the same value.

- TIE Bit

This bit enables/disables output of transmit data empty interrupts. Transmission data empty interrupt is used as an interrupt factor written in the FIFO transmission data register. Set this bit to 1 after setting set condition (SSISCR.TDES) of transmission data empty interrupt. See Figure 24.25 for the generation timing of a transmit data empty interrupt.

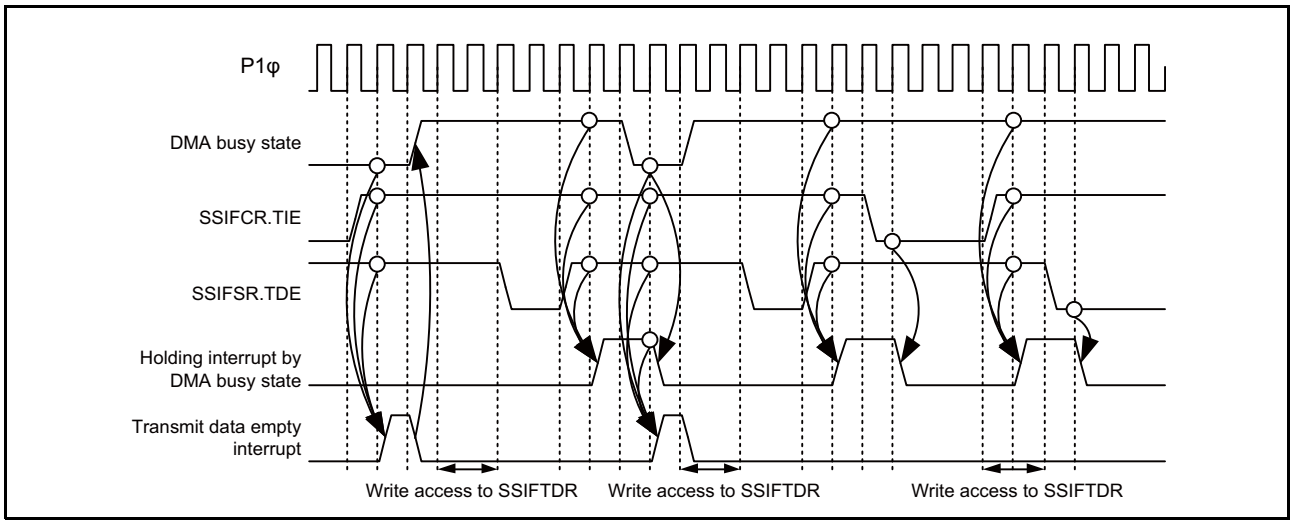


Figure 24.25 Generation timing of transmit data empty interrupt

- RIE Bit

This bit enables/disables output of receive data full interrupts. Receive data empty interrupt is used as an interrupt factor to read the FIFO receive data register. Set this bit to 1 after setting set condition (SSISCR.RDFS) of Receive data empty interrupt. See Figure 24.26 for the generation timing of a receive data full interrupt.

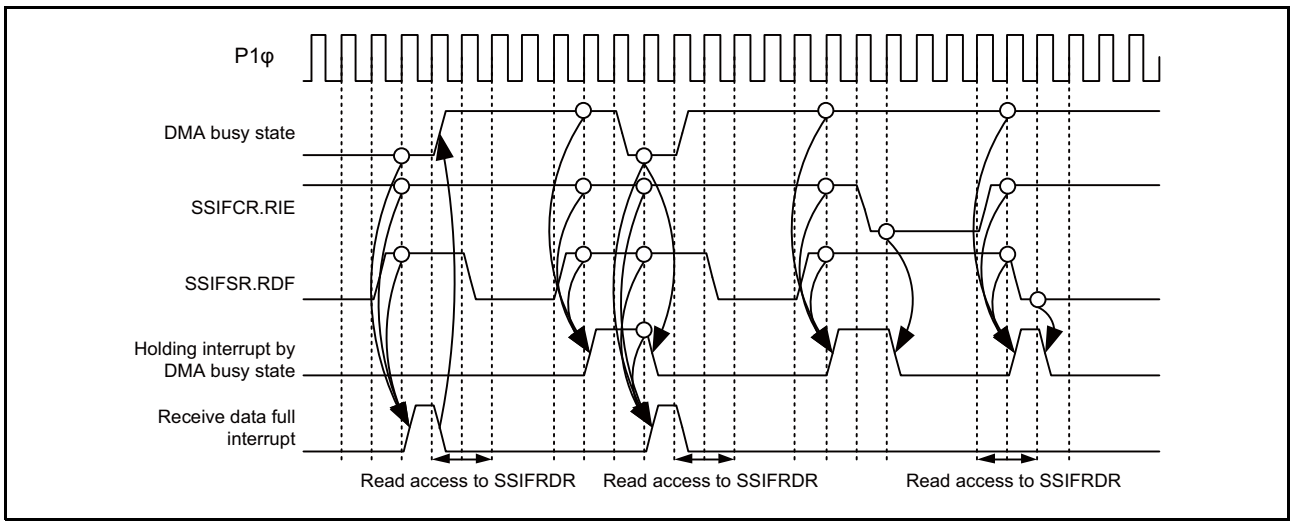


Figure 24.26 Generation timing of receive data empty interrupt

- TFRST Bit

This bit sets a software reset of the transmit FIFO data register (SSIFTDR). Writing 1 to this bit initializes the internal state related to SSIFTDR. See Table 24.8 for the registers to which this software reset is applied. After a reset by writing 1, write 0 to clear the reset condition because this bit is not automatically cleared to 0. After writing 0 to this bit, make sure that the bit becomes 0 before the next procedure.
This bit is a bit of SSIRST for reset. Software reset by SSIRST becomes and writing this bit becomes invalid because of priority.

Table 24.8 Bits Initialized by Software Reset of the SSIFCR.TFRST Bit

			+0								+1							
	Address		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol	(Base+)		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	00h	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	04h	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	10h	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	BCKNCE	LRCKNCE	RxDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	14h	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	18h	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	1ch	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	20h	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	24h	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

- RFRST Bit

This bit sets a software reset of the receive FIFO data register (SSIFRDR). Writing 1 to this bit initializes the internal state related to SSIFRDR. After a reset by writing 1, write 0 to clear the reset condition because this bit is not automatically cleared to 0. After writing 0 to this bit, make sure that the bit becomes 0 before the next procedure.
This bit is a bit of SSIRST for reset. Software reset by SSIRST becomes and writing this bit becomes invalid because of priority.

Table 24.9 Bits Initialized by Software Reset of the SSIFCR.TFRST Bit

			+0								+1							
	Address		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol	(Base+)		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	00h	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	04h	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	10h	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	BCKNCE	LRCKNCE	RxDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	14h	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	18h	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	1ch	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	20h	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	24h	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

24.4.1.4 FIFO Status Register (SSIFSR)

This register is configured with status flags that indicate the status of the transmit FIFO data register and the receive FIFO data register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W0

Bit	Bit Name	R/W	Initial Value	Description
31, 30	—	R	0H	Reserved. Write 0. The read value is 0.
29 to 24	TDC[5:0]	R	00H	Number of Transmit FIFO Data Indication Flag
23 to 17	—	R	00H	Reserved. Write 0. The read value is 0.
16	TDE	RW0	1	Transmit Data Empty Flag 0: The free space of SSIFTDR is not more than the value of SSISCR.TDES 1: The free space of SSIFTDR is not less than the value of SSISCR.TDES plus one.
15, 14	—	R	0H	Reserved. Write 0. The read value is 0.
13 to 8	RDC[4:0]	R	00H	Number of Receive FIFO Data Indication Flag
7 to 1	—	R	00H	Reserved. Write 0. The read value is 0.
0	RDF	RW0	0	Receive Data Full Flag 0: The size of received data in SSIFRDR is not more than the value of SSISCR.RDFS 1: The size of received data in SSIFRDR is not less than the value of SSISCR.RDFS plus one.

- TDC[5:0] Bits

These bits indicate the number of valid data that are stored in the transmit FIFO data register (SSIFTDR). With this flag as 0H, there is no data to be transmitted. With 10H, there is no space to write data.

- TDE Bit

This bit indicates that the transmit FIFO data register (SSIFTDR) has free space not less than the amount set with the SSIFCR.TTRG bit plus one. This flag is set by automatic determination but it must be cleared by register access.
[Priority order for setting and clearing]
Clearing is prioritized.*1
[Clearing condition]
Either of the following two:
(1) Writing 0 to this bit after reading this bit as 1. (CPU operation)*2
(2) Last access when SSIFTDR is written by using DMA in one interrupt routine (DMA operation).

[Clearing timing]
Timing according to the above-mentioned clear condition.
(1) At completion of writing 0 to this bit after reading this bit as 1 (Figure 24.13).
(2) At completion of last access when SSIFTDR is written by using DMA in one interrupt routine.

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1) and transmit FIFO data register reset (SSIFCR.TFRST = 1). The software reset and transmit FIFO data register reset have priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following four conditions being met:
- Software reset (SSIFCR.SSIRST = 1)
- Resetting of the transmit FIFO data register (SSIFCR.TFRST = 1)
- Completion of writing 0 to this flag after having read it as 1.
- Last access is performed to write data to SSIFTDR by an interrupt routine using DMA.

[Setting condition]
SSIFTDR has free space not less than the amount set with the SSIFCR.TTRG bit plus one.
[Setting timing]
While operating on P1φ, SSIFTDR is found to have free space not less than "size set in the SSISCR.TDES bits + 1."

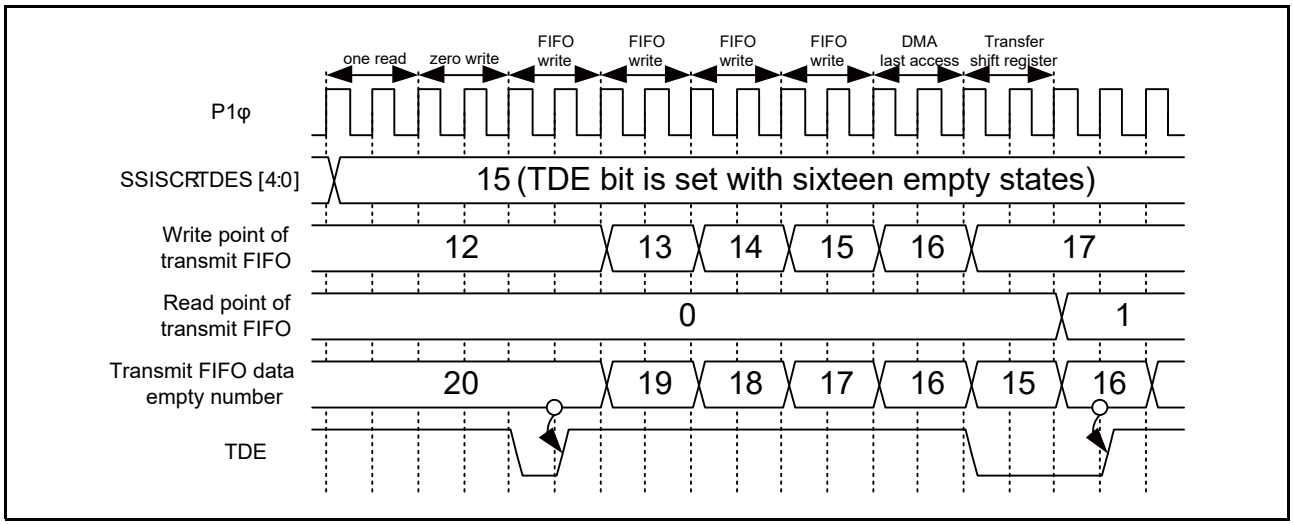


Figure 24.27 Set timing and clear timing of TDE

- RDC[5:0] Bits

These bits indicate the number of valid data that are stored in the receive FIFO data register (SSIFRDR). With this flag as 0H, there is no received data. With 20H, the register is filled with received data and there is no free space.

- RDF Bit

This bit indicates that the receive FIFO data register (SSIFRDR) has unread received data not less than the amount set with the SSISCR.RDFS bit plus one. This flag is set by automatic determination but it must be cleared by register access.
[Priority order for setting and clearing]
Clearing is prioritized.*1
[Clearing condition]
Either of the following two:
(1) Writing 0 to this bit after reading this bit as 1. (CPU operation)*2
(2) Last access when SSIFRDR is read by using DMA in one interrupt routine (DMA operation).

[Clearing timing]
Timing according to the above-mentioned clear condition.
(1) At completion of writing 0 to this bit after reading this bit as 1 (Figure 24.13).
(2) One cycle of P1φ after the last read access to SSIFRDR by using DMA in one interrupt routine.

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1) and receive FIFO data register reset (SSIFCR.RFRST = 1). Clearing conditions available for these bits are the software reset and receive FIFO data register reset as well as the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following four conditions being met:
- Software reset (SSIFCR.SSIRST = 1)
- Receive FIFO data register reset (SSIFCR.RFRST = 1)
- Completion of writing 0 to this flag after having read it as 1.
- Last access is performed to read data from SSIFRDR by an interrupt routine using DMA.

[Setting condition]

SSIFRDR has data not less than the amount set with the SSISCR.RDFS bit plus one.

[Setting timing]

At completion of transfer from the shift register that results in SSIFRDR having data not less than the amount set with the SSISCR.RDFS bit plus one.

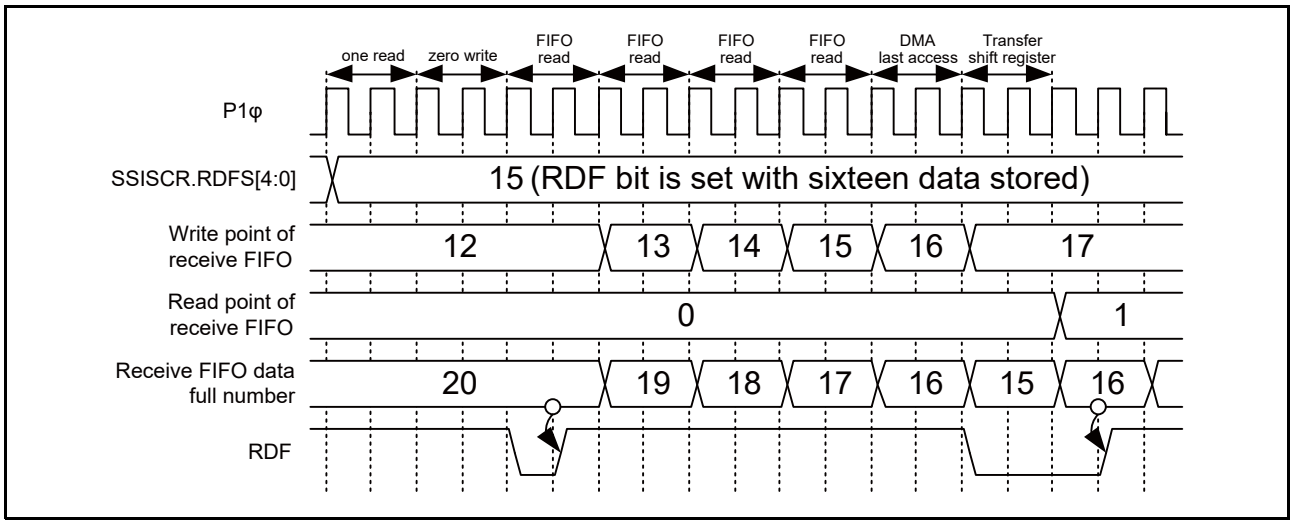
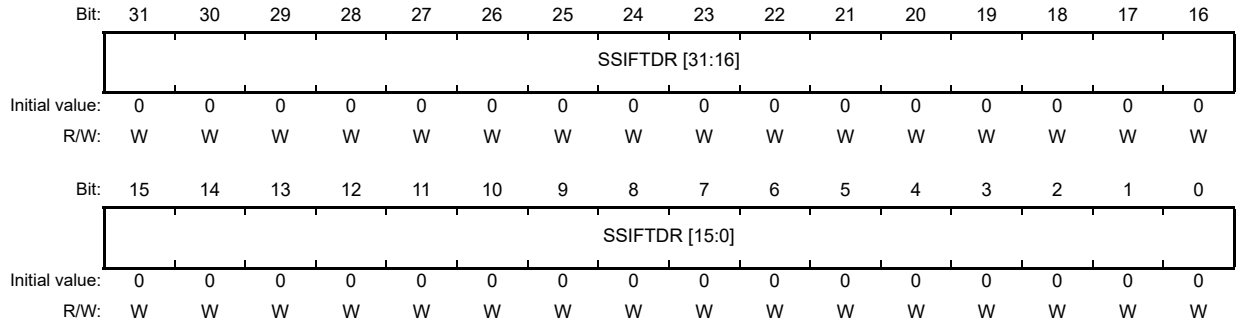


Figure 24.28 Set timing and clear timing of RDF

24.4.1.5 Transmit FIFO Data Register (SSIFTDR)

This is a 32-bit writable register. 0 is returned when this register is read. This register stores data to be serially transmitted.



Bit	Bit Name	R/W	Initial Value	Description
31 to 0	SSIFTDR[31:0]	W	00000000H	Transmit FIFO data

To use this register for transmission, set DMA operation handling of a transmit data empty interrupt as writing to this register. Determine the access size to this register according to the data word length to be communicated (Table 24.10).

Table 24.10 Register Access Restriction to FIFOs

		Access Size		
SSICR.DWL[2:0]	Data Word Length	Byte	Halfword	Word
000B	8	√	—	—
001B	16	—	√	—
010B	18	—	—	√
011B	20	—	—	√
100B	22	—	—	√
101B	24	—	—	√
110B	32	—	—	√
111B	Setting prohibited	—	—	—

Figure 24.29 shows register access to the transmit FIFO data register.

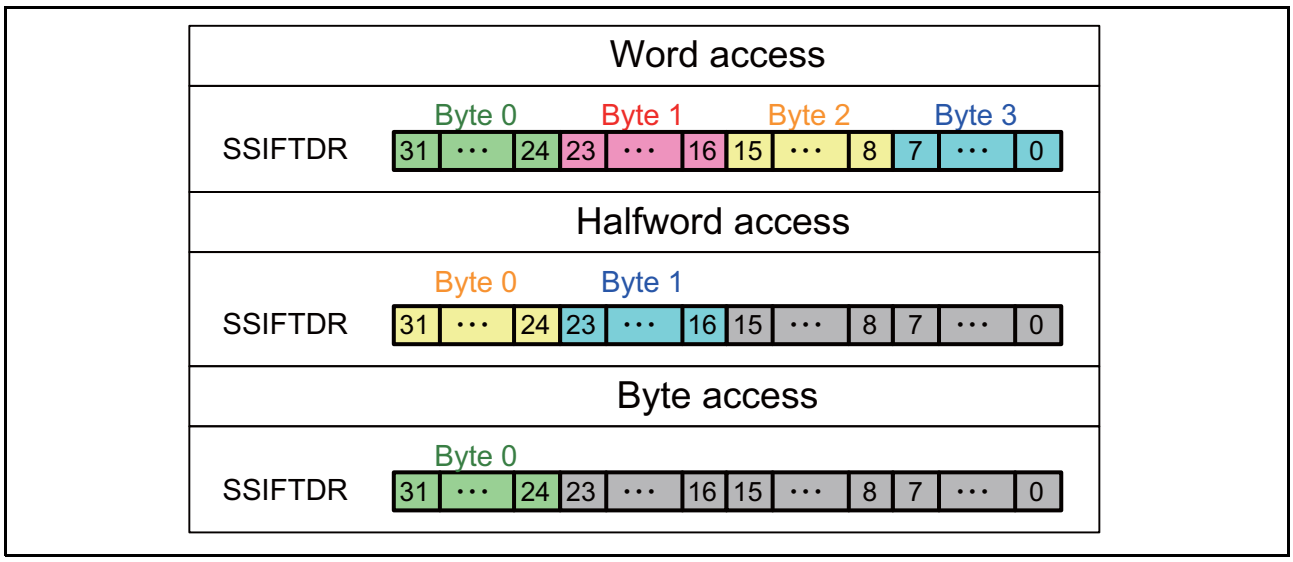


Figure 24.29 Example of Register Access to the Transmit FIFO Data Register

Figure 24.30 shows the configurations and operation examples of the transmit FIFO data register and transmit shift register. The configurations are for storing data to FIFO and not related with communication.

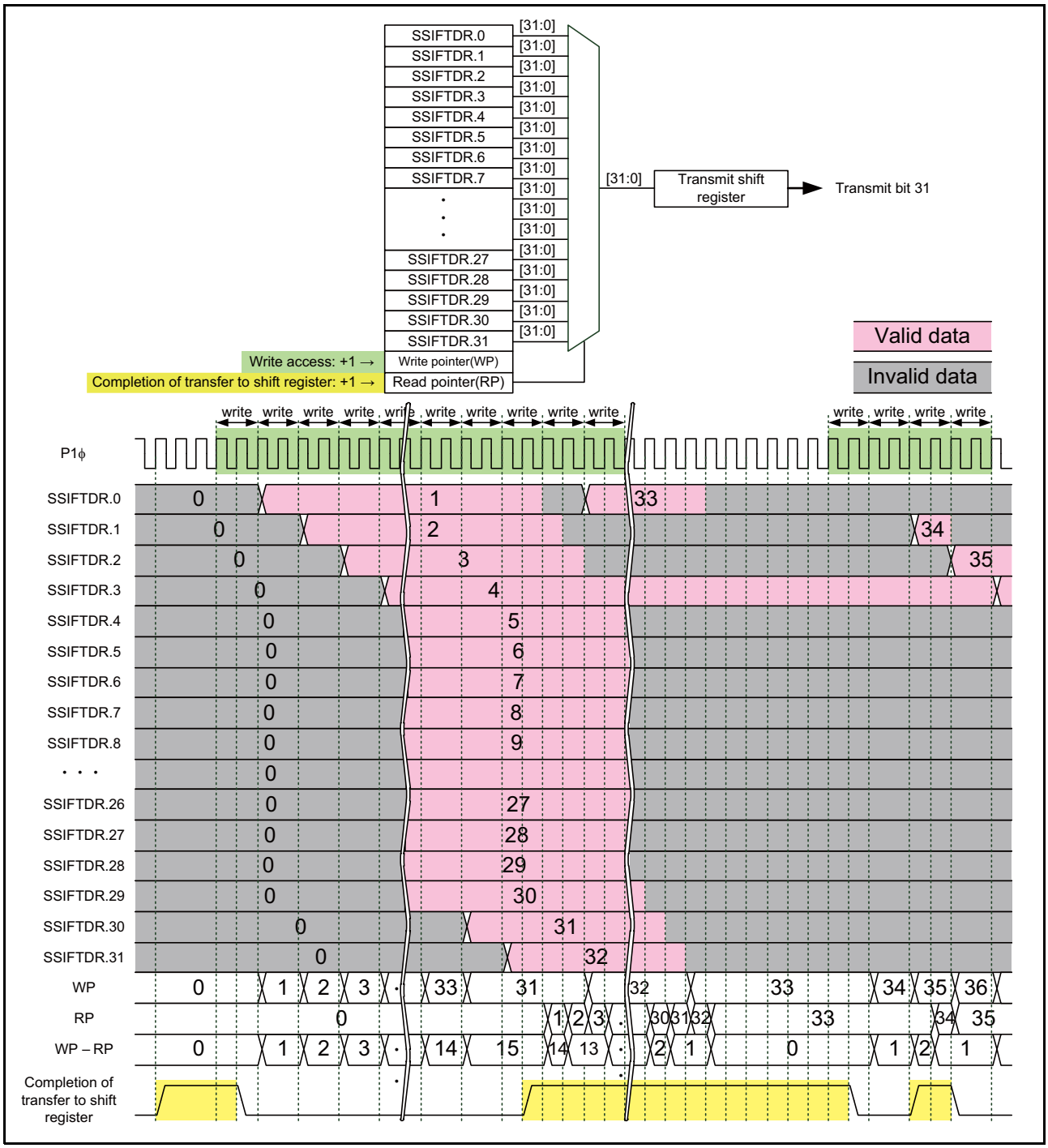
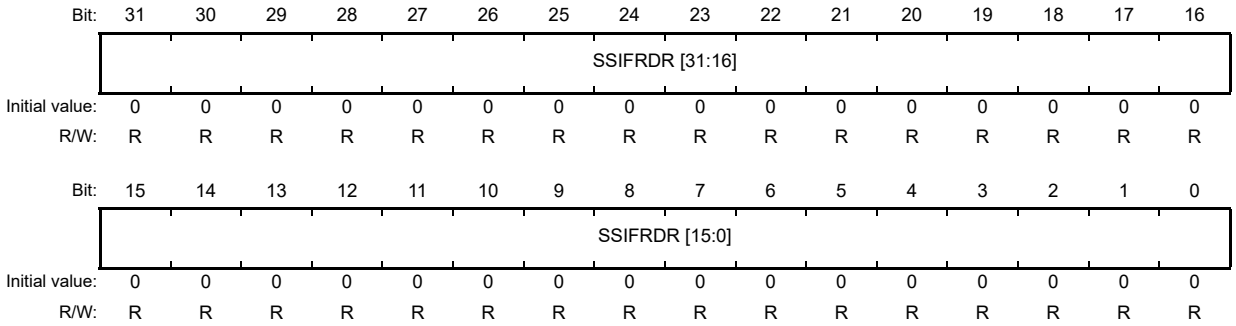


Figure 24.30 Configuration of the Transmit FIFO Data Register and Transmit Shift Register, and FIFO Operation Example

24.4.1.6 Receive FIFO Data Register (SSIFRDR)

This is a readable 32-bit register. This register stores received serial data.



Bit	Bit Name	R/W	Initial Value	Description
31 to 0	SSIFRDR[31:0]	R	00000000H	Receive FIFO data

To use this register for reception, set DMA operation handling of a receive data full interrupt as reading from this register. Determine the access size to this register according to the data word length to be communicated (Table 24.10). Register access to the receive FIFO data register is same as for the transmit FIFO data register (Figure 24.29). Figure 24.31 shows the configurations and operation examples of the receive FIFO data register and receive shift register.

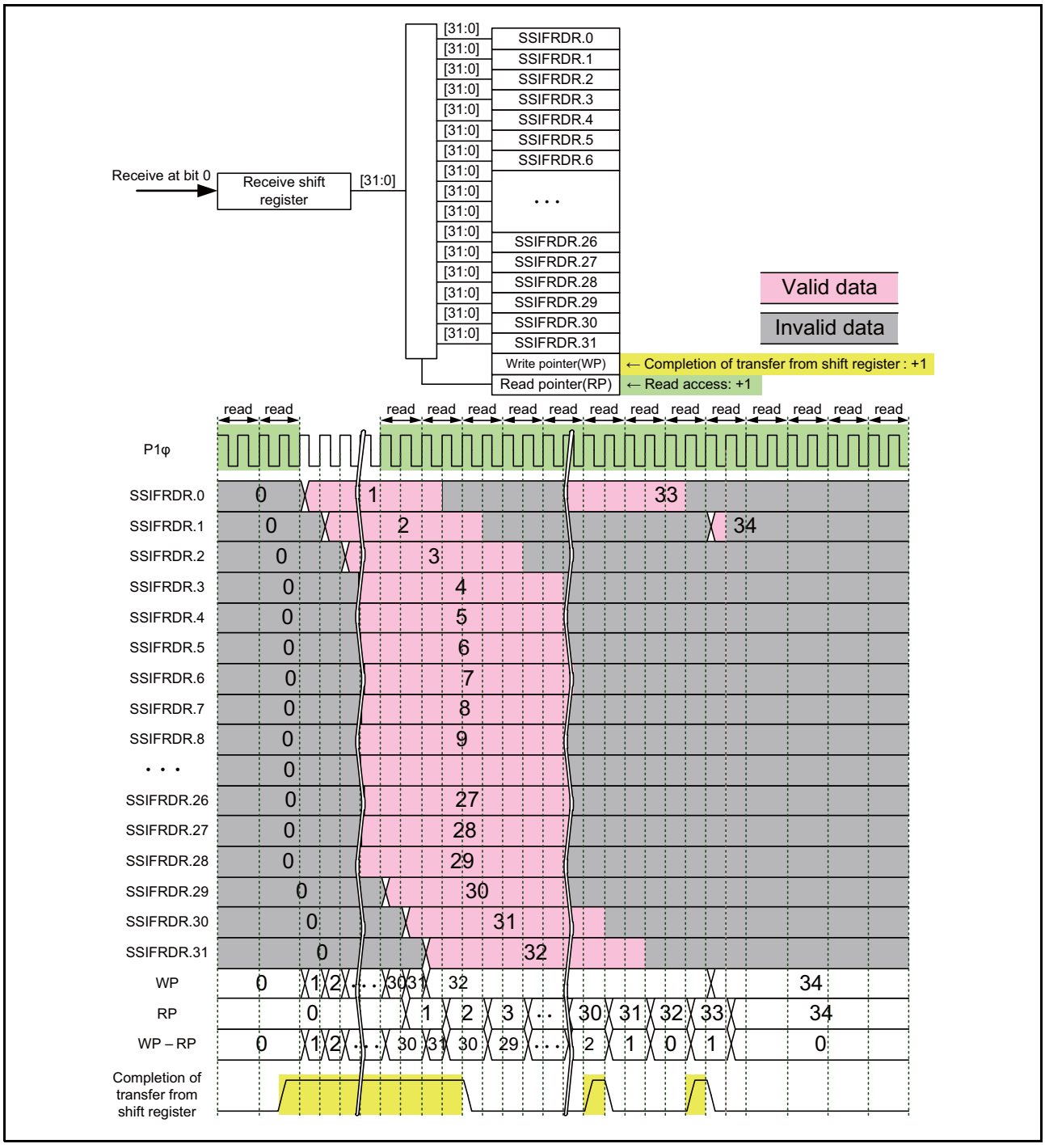


Figure 24.31 Configuration of the Receive FIFO Data Register and Receive Shift Register, and FIFO Operation Example

24.4.1.7 Audio Format Register (SSIOFR)

This is a readable/writable 32-bit register. It sets an audio format including the communication format, LRCK/FS continuation mode, and BCK output stop.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	BCK STP	LR CONT	—	—	—	—	—	—	OMOD[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	R/W	Initial Value	Description
31 to 10	—	R	000000H	Reserved. Write 0. The read value is 0.
9	BCKASTP	RW	0	BCK Output Stop Enable*1*2 0: Enables output of BCK 1: Disables output of BCK
8	LRCONT	RW	0	LRCK/FS Continuation Enable*1*2 0: Disables LRCK continuation 1: Enables LRCK continuation
7 to 2	—	R	00H	Reserved. Write 0. The read value is 0.
1,0	OMOD[1:0]	RW	0	Audio Format Select*3*4 OMOD[1:0]: Format 00: I²S format 01: TDM format 10: Monaural format 11: Setting prohibited

Note 1. This bit is valid only in master-mode communication (SSICR.MST = 1). The setting is invalid in slave-mode communication (SSICR.MST = 0).
Note 2. BCKASTP=1 and LRCONT=1 is prohibitions.
Note 3. Writing in this bit is a prohibition when SSIF-2 is communication state (SSISR.IIRQ=0). Operation is not guaranteed when rewriting it.
Note 4. Use the communication format setting that can be communicated when the format of the communication of the opposing device is compatible format of SSIF-2.

- BCKASTP Bit

This bit turns on or off the function to output BCK to the SSIBCK pin according to the communication shown in Figure 24.32 and Figure 24.33 in master-mode communication (SSICR.MST = 1).
Set this bit after setting completing the communication format.
This bit must defend the following usage. Start communication with BCKASTP = 0, and set 1 to BCKASTP while communicating. As a result, the bit clock output to the terminal SSIBCK stops by the automatic operation when the communication stop is done. When you will restart the communication, set 0 to BCKASTP with idle state (SSICR.IIRQ=1) and the AUDIO_MCK supply state (SSIFCR.AUCKE = 1)

When master communication (SSICR.MST=1) and idle state (SSICR.IIRQ=1).

Table 24.11 BCKASTP state, and Output state of SSIBCK pin.

BCKASTP bit	output state of SSIBCK pin
0	Enable
1	Disable

Note: When the opposing device who is the slave needs the clock of the terminal SSIBCK before the communication operates, it is not possible to use it. Use it to stop clocking after the communication ends (Figure 24.33). Refer to Figure 24.32 for timing that the function becomes effective.

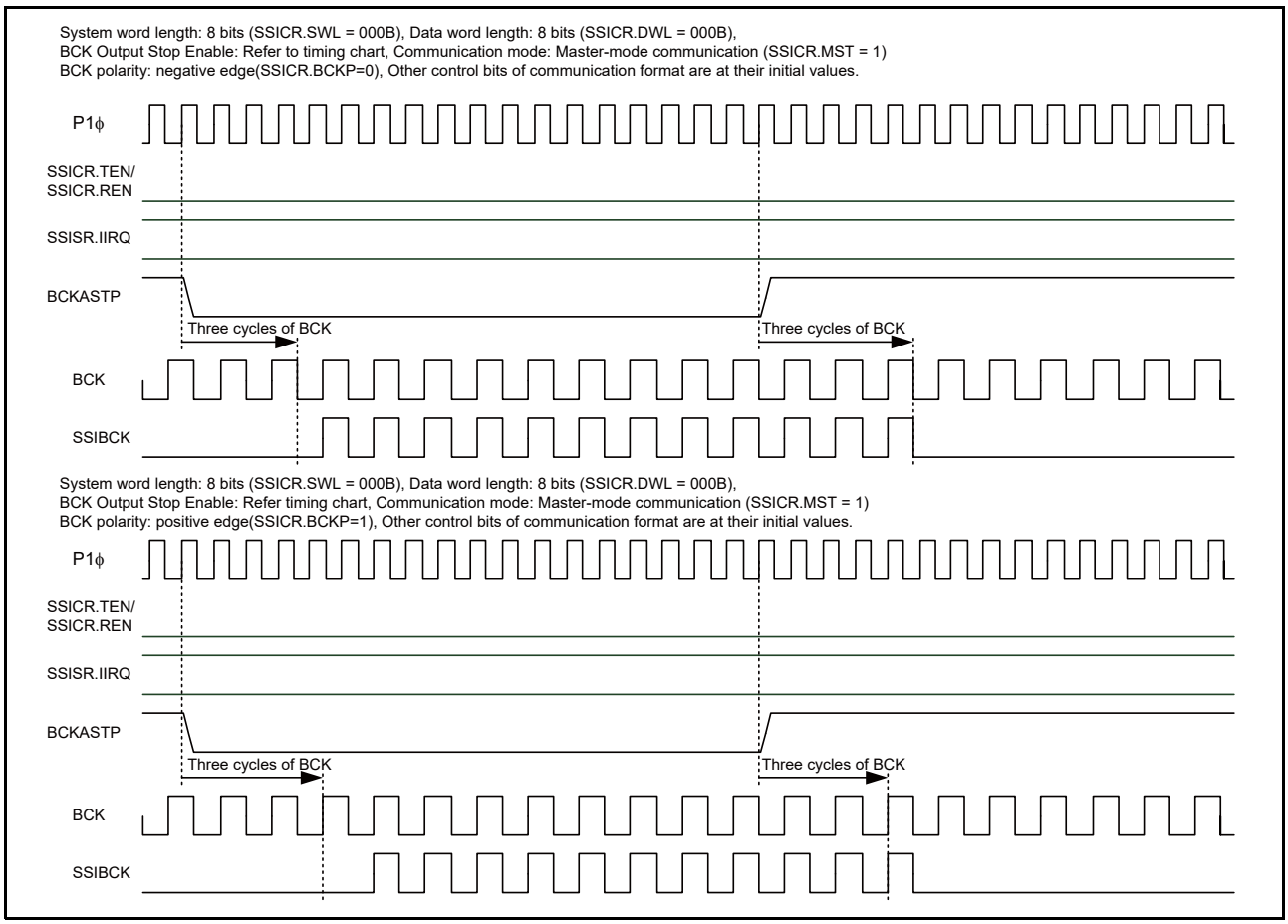


Figure 24.32 Example Operation of the BCKASTP Bit (in idle state)

When master communication (SSICR.MST=1) and the BCK output automatic operation stop function (BCKASTP = 1) (Figure 24.33).

Details of the BCK output to SSIBCK pin are as follows.

- Output start timing: To generate an effective edge in timing that LRCK/FS is changed into the valid value, BCK is output.
- Output stop timing: Frame boundary from 1 to 1.5 clock

Refer to the timing chart of Figure 24.33 for detailed timing.

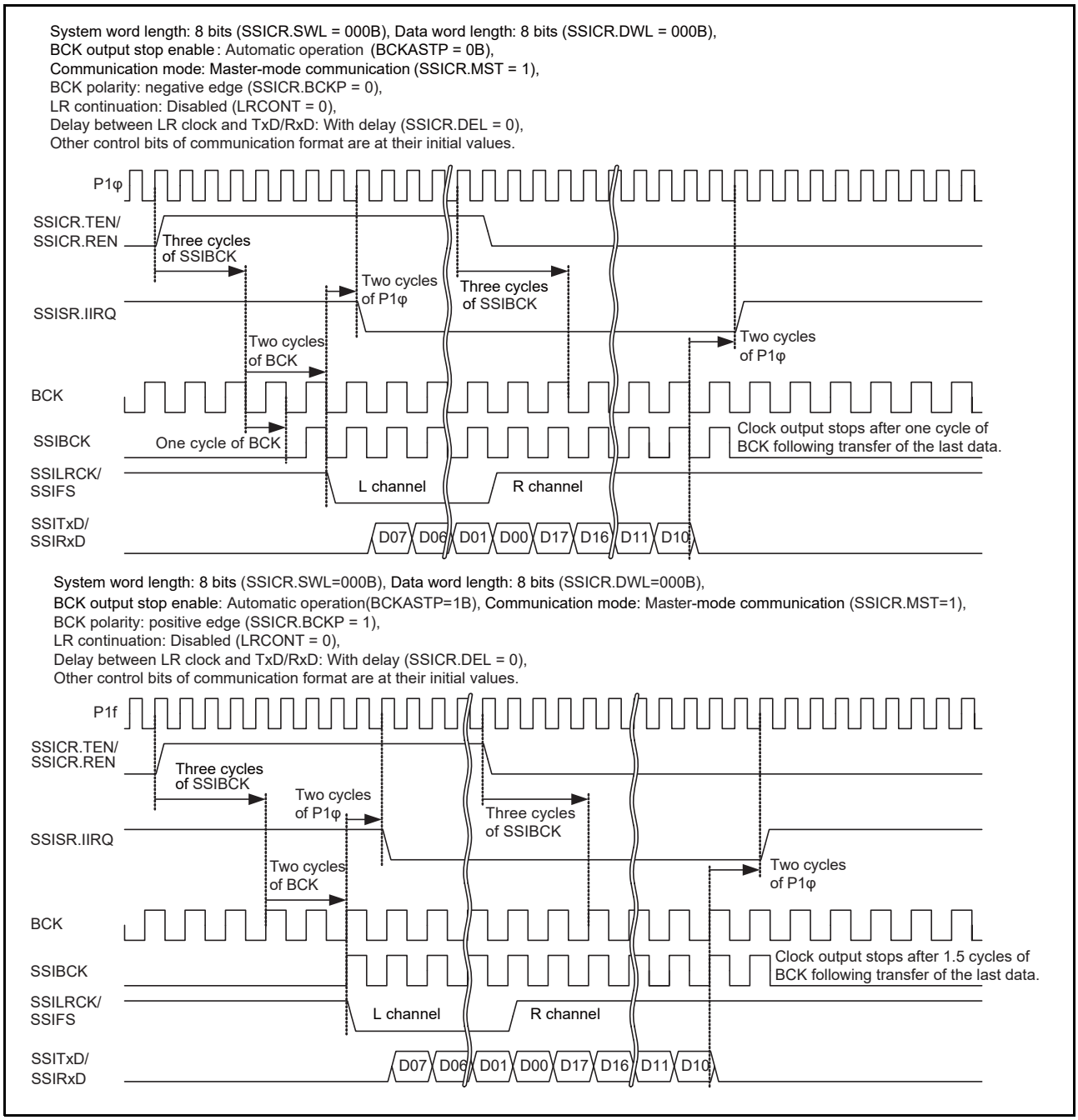


Figure 24.33 Example Operation of the BCKASTP Bit (Communication behavior in BCKASTP=1)

- LRCONT Bit

This bit enables/disables the output from the SSILRCK/SSIFS pin when SSIF-2 is master communication (SSICR.MST=1) and idle state (SSISR.IIRQ=1). Writing 1 to this bit (enables LRCK/FS continuation) in master mode (SSICR.MST = 1) enables continuation of the output from the SSILRCK/SSIFS pins even when idle state.

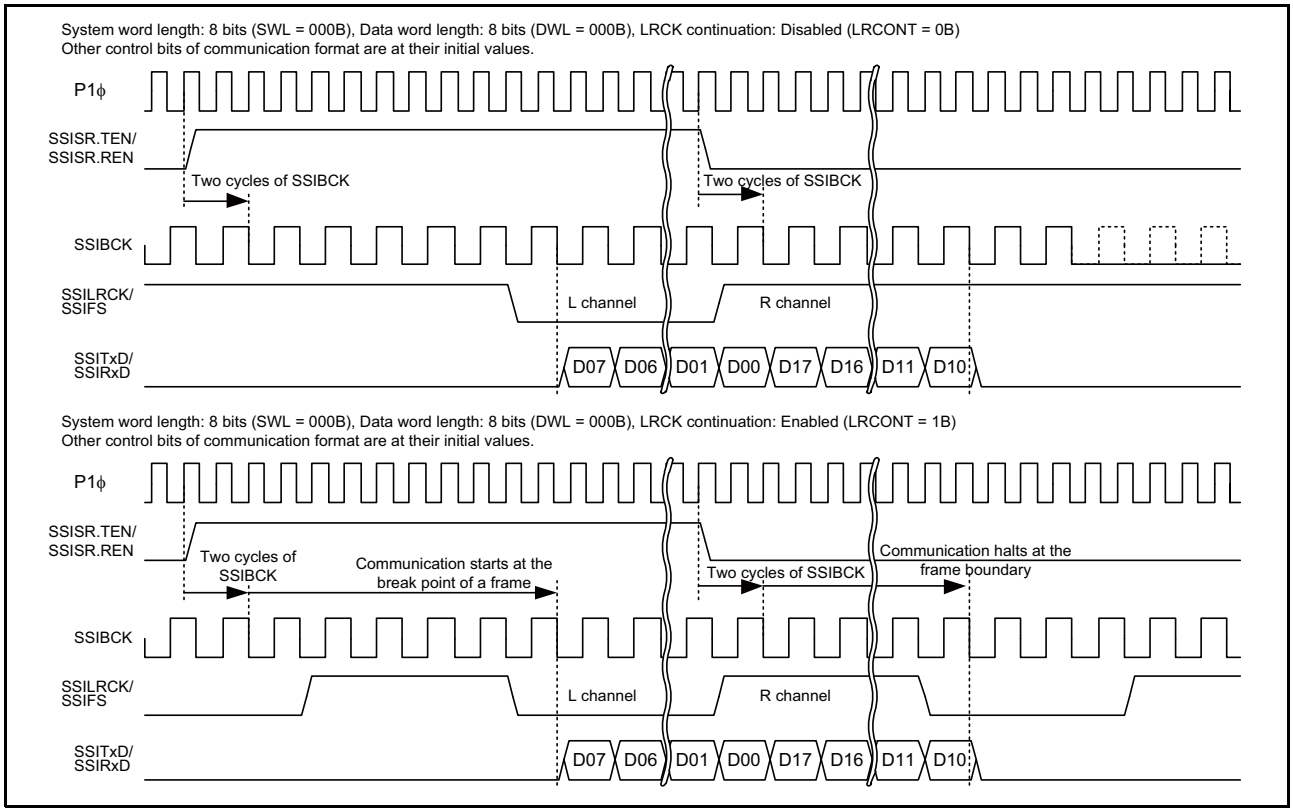


Figure 24.34 Operation Example of LRCK/FS Continuation (in idle state)

- OMOD[1:0] Bits

These bits set an audio format.

Rewrite this bit when the LR clock supply to the terminal SSILRCK stops. Refer to the LRCONT bit details explanation in section 24.4.1.7, Audio Format Register (SSIOFR) for the output operation of the LR clock.

24.4.1.8 Status Control Register (SSISCR)

This is a readable/writable 32-bit register. It sets the operation of the TDE and RDF flags.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	R/W	Initial Value	Description
31 to 13	—	R	00000H	Reserved. Write 0. The read value is 0.
12 to 8	TDES[4:0]	RW	0H	TDE Setting Condition Select*1 TDES[4:0]: Operation 00000B: SSIFTDR has one stage or more free space 00001B: SSIFTDR has two stages or more free space (snip) 11110B: SSIFTDR has thirty-one stages or more free space 11111B: SSIFTDR has thirty-two stages or more free space
7 to 5	—	R	0H	Reserved. Write 0. The read value is 0.
4 to 0	RDFS[4:0]	RW	0H	RDF Setting Condition Select*1 RDFS[4:0]: Operation 00000B: SSIFRDR has one stage or more data size 00001B: SSIFRDR has two stages or more data size (snip) 01110B: SSIFRDR has thirty-one or more data size 01111B: SSIFRDR has thirty-two stages or more data size

Note 1. Writing to these bits while SSIF-2 is in a communication state (SSISR.IIRQ = 0) is prohibited. If written, the operation performed immediately after writing is not guaranteed.

- TDES[4:0] Bits

These bits set the setting condition of the transmit data empty flag (TDE). Set these bits in byte units.

- RDFS[4:0] Bits

These bits set the setting condition of the receive data full flag (RDF). Set these bits in byte units.

24.4.2 Communication Formats

SSIF-2 supports three communication formats (Table 24.12).

Table 24.12 Supported Communication Formats

Communication Format	SSIOFR.OMOD[1:0]
I²S format	00
TDM format	01
Monaural format	10

The following figure shows the serial data configuration common to communication formats. Serial data is determined according to the system word length (SSICR.SWL[2:0]) and the data word length (SSICR.DWL[2:0]). When the system word length is longer than the data word length, padding bits are transferred (Figure 24.35).

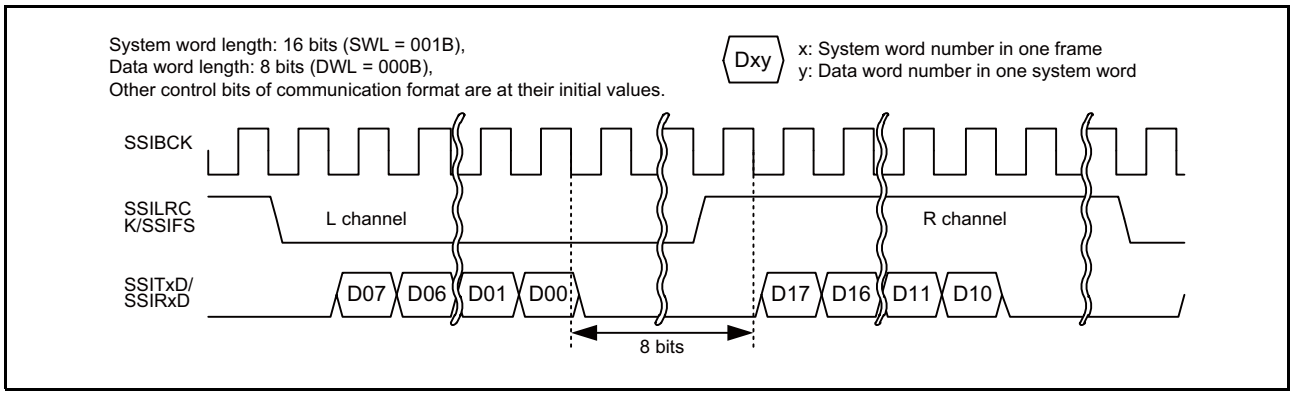


Figure 24.35 Example of Padding Bit Transfer (I²S Format: System Word Length > Data Word Length)

Table 24.13 lists the number of padding bits to be transferred with each combination of system word length (SSICR.SWL[2:0]) and data word length (SSICR.DWL[2:0]). "-" indicates that the setting is prohibited.

Table 24.13 Number of Padding Bits

	SSICR.DWL[2:0]	000B	001B	010B	011B	100B	101B	110B	111B
SSICR.SWL[2:0]	System Word Length	8	16	18	20	22	24	32	Set prohibition
000B	8	0	—	—	—	—	—	—	—
001B	16	8	0	—	—	—	—	—	—
010B	24	16	8	6	4	2	0	—	—
011B	32	24	16	14	12	10	8	0	—
100B	48	40	32	30	28	26	24	16	—
101B	64	56	48	46	44	42	40	32	—
110B	128	120	112	110	108	106	104	96	—
111B	256	248	240	238	236	234	232	224	—

24.4.2.1 I²S Format

The I²S format is a communication format used for connection with I²S-compatible serial devices. With this format setting (SSIOFR.OMOD[1:0] = 00B), one frame is configured with two system words, one for the channel L and the other for channel R. The SSILRCK/SSIFS signals are at a low level for the channel L and at a high level for the channel R. Set the polarity of the signals with the SSICR.LRCKP bit. Figure 24.36 shows the I²S format without padding. See Figure 24.35 for the format with padding.

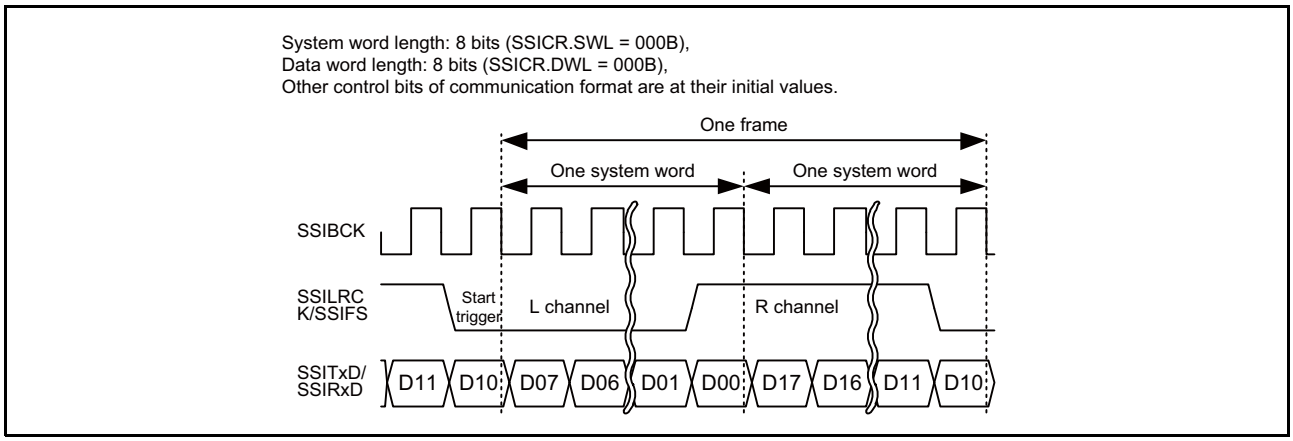


Figure 24.36 I²S Format (Without Padding: System Word Length = Data Word Length)

For the state of external pins when SSIF-2 is in the idle state, see section 24.5.1.1, Idle State.

Note: SSIF-2 has the terminal SSILRCK/SSIFS that shows the synchronization of the communication. Match communication format of the opposing device and SSIF-2 when SSIF-2 is slave communication (SSICR.MST=0). SSIF-2 uses the signal of SSILRCK/SSIFS only as a communication start trigger.

24.4.2.2 Monaural Format

The monaural format is a communication format used for connection with monaural-compatible serial devices. With this format setting (SSIOFR.OMOD[1:0] = 10B), one frame is configured with the number of system words set with the SSICR.FLM[1:0] bits (Figure 24.37). A rising of the SSILRCK/SSIFS signal means a start trigger. Figure 24.37 and Figure 24.38 respectively show the monaural formats without and with padding.

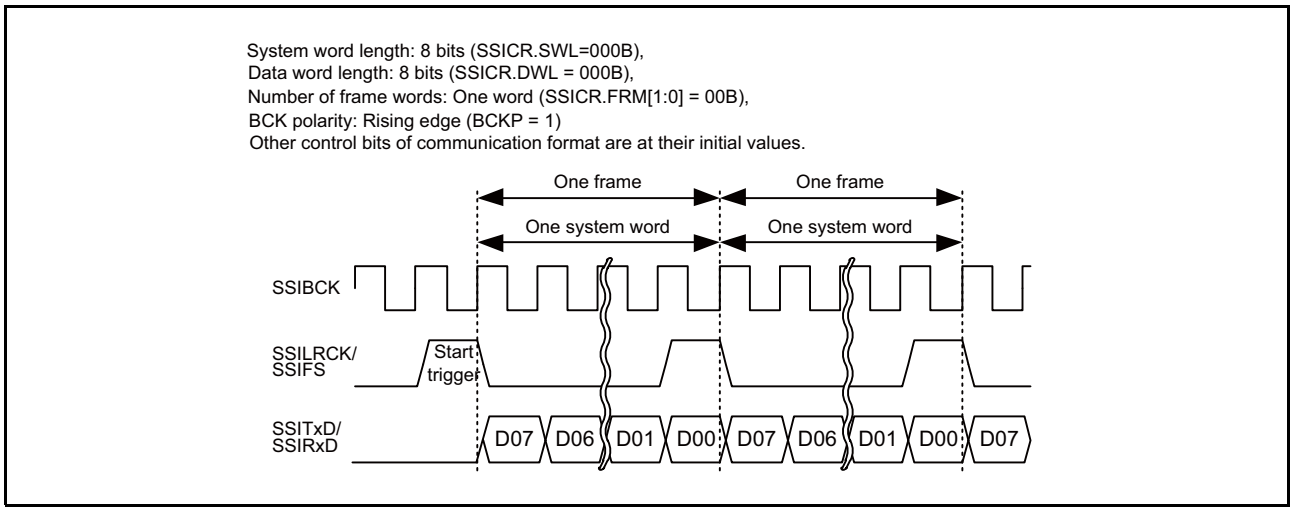


Figure 24.37 Short Frame in Monaural Format (Without Padding: System Word Length = Data Word Length)

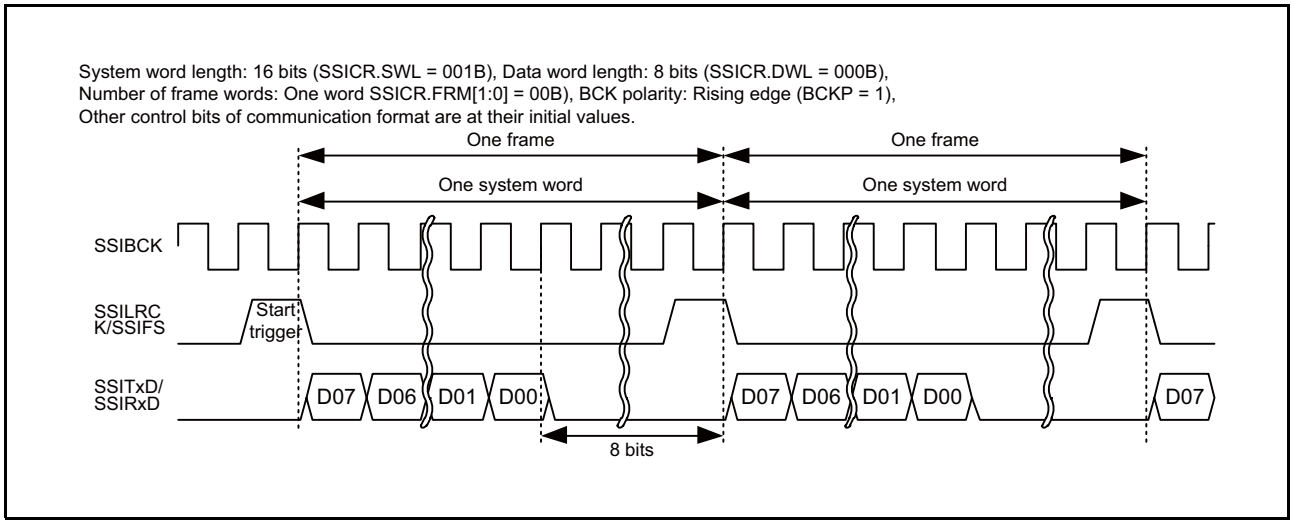


Figure 24.38 Short Frame in Monaural Format (With Padding: System Word Length > Data Word Length)

The monaural formats supported by SSIF-2 consist of short frames and long frames. See section 24.4.2.2, (1), Short Frame and (2), Long Frame for the difference between these two frames.
For the state of external pins state when SSIF-2 is in the idle state, see section 24.5.1.1, Idle State.

Note: SSIF-2 has the terminal SSILRCK/SSIFS that shows the synchronization of the communication. Match communication format of the opposing device and SSIF-2 when SSIF-2 is slave communication (SSICR.MST=0). SSIF-2 uses the signal of SSILRCK/SSIFS only as a communication start trigger.

(1) Short Frame

With the short frame (SSICR.DEL = 0), the SSILRCK/SSIFS signal that indicates a start of serial data is at a high level for one cycle of SSIBCK. Data transfer starts at the falling edge of the signal (Figure 24.37 and Figure 24.38).

(2) Long Frame

With the long frame (SSICR.DEL = 1), the SSILRCK/SSIFS signal that indicates a start of serial data is at a high level for two cycles of SSIBCK. Data transfer starts at the rising edge of the signal (Figure 24.39).

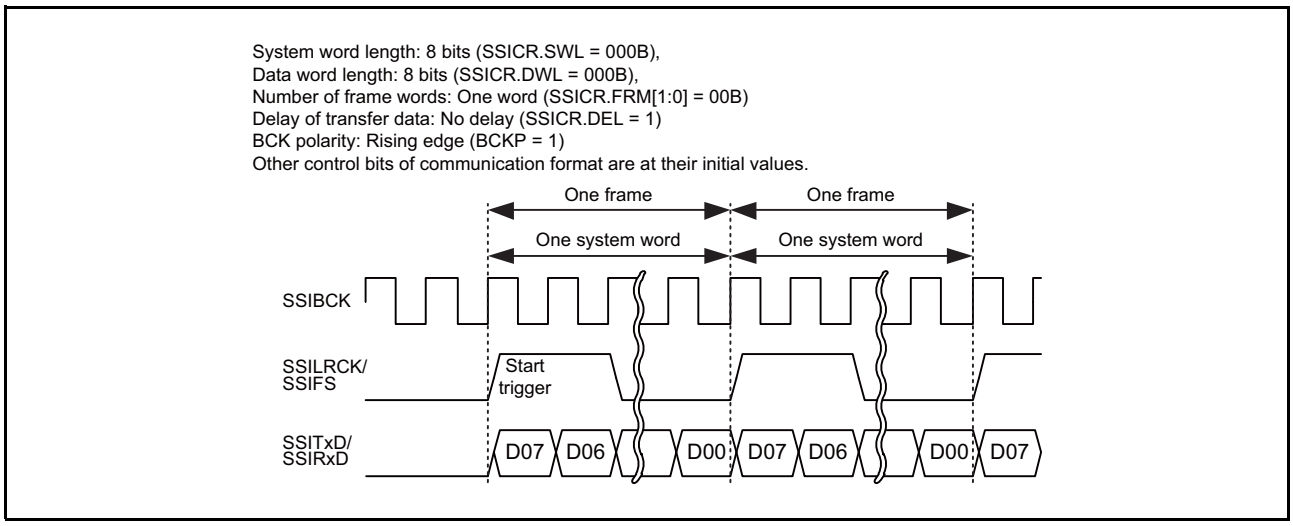


Figure 24.39 Long Frame in Monaural Format (Without Padding)

24.4.2.3 TDM Format

The TDM format is a communication format used for connection with TDM-compatible multi-channel devices. With this format setting (SSIOFR.OMOD[1:0] = 01B), one frame is configured with four to eight system words set with the SSICR.FRM[1:0] bits. With this format, the SSILRCK/SSIFS signal is at a high level for the first one system word and at a low level for the rest. The pulse generated on the SSILRCK/SSIFS signal is defined as the SYNC pulse and its rising edge means a start of one frame. Figure 24.40 and Figure 24.41 respectively show the TDM formats with and without padding.

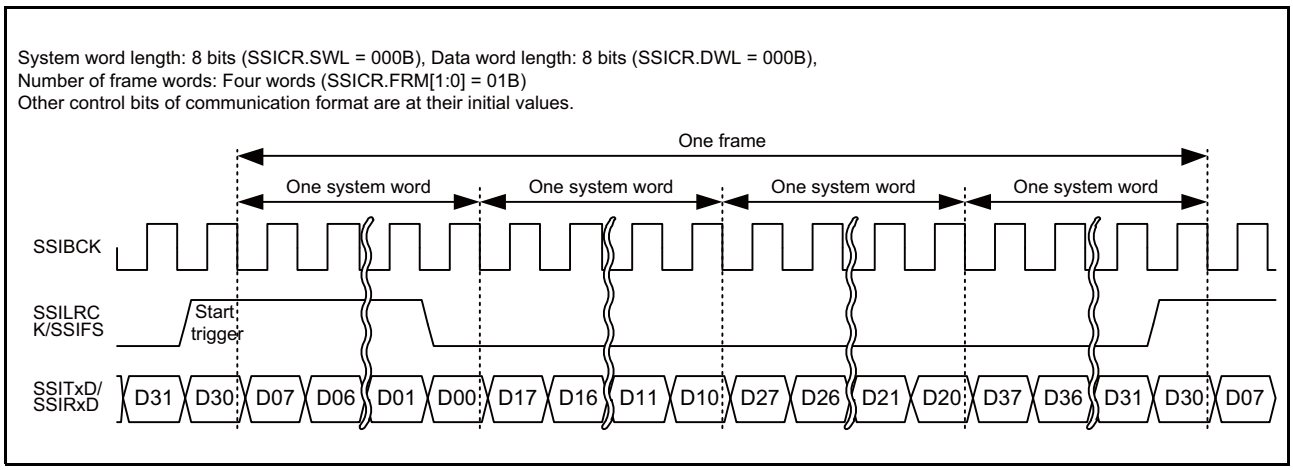


Figure 24.40 TDM Format (Without Padding: System Word Length = Data Word Length)

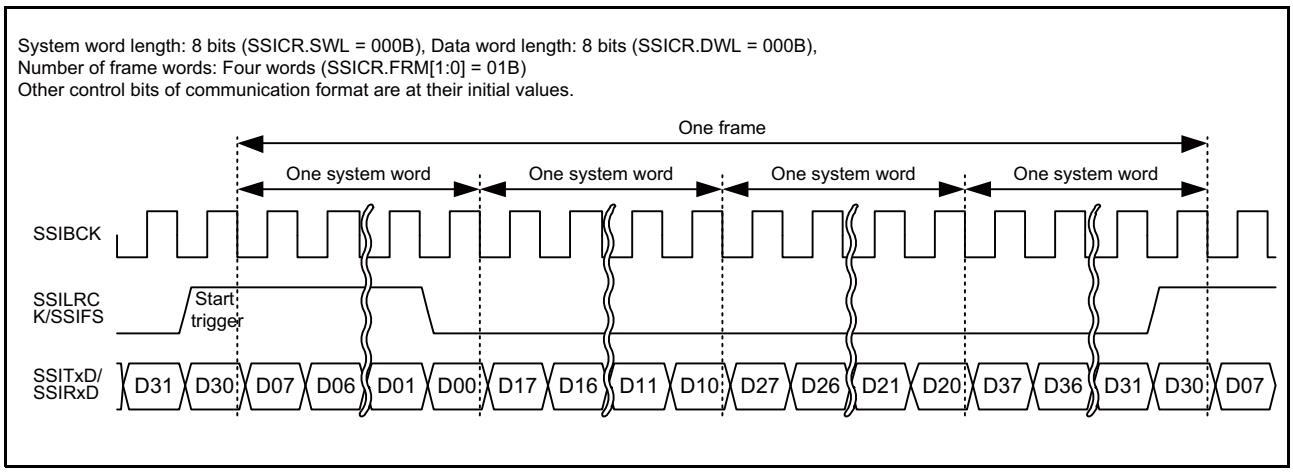


Figure 24.41 TDM Format (With Padding: System Word Length > Data Word Length)

For the state of external pins when SSIF-2 is in the idle state, see section 24.5.1.1, Idle State.

Note: SSIF-2 has the terminal SSILRCK/SSIFS that shows the synchronization of the communication. Match communication format of the opposing device and SSIF-2 when SSIF-2 is slave communication (SSICR.MST=0). SSIF-2 uses the signal of SSILRCK/SSIFS only as a communication start trigger.

24.4.3 Communication Modes

SSIF-2 supports the following communication modes. Table 24.15 lists the control bits that are not available with each communication mode. See section 24.4.3.1 to section 24.4.3.5 for details of these communication modes.

Table 24.14 Communication Modes

Communication Mode	SSICR.MST Bit	SSICR.REN Bit	SSICR.TEN Bit
Slave-mode transmission	0	0	1
Slave-mode reception	0	1	0
Slave-mode transmission and reception	0	1	1
Master-mode transmission	1	0	1
Master-mode reception	1	1	0
Master-mode transmission and reception	1	1	1

Table 24.15 Control Bits that Cannot be Used in Each Communication Mode

Communication / Mode Control Bit	Slave-mode Reception	Slave-mode Transmission	Slave-mode Transmission and Reception	Master-mode Reception	Master-mode Transmission	Master-mode Transmission and Reception
SSICR.CKS	Invalid	Invalid	Invalid	Available	Available	Available
SSICR.CKDV	Invalid	Invalid	Invalid	Available	Available	Available
SSICR.MUEN	Invalid	Available	Available	Invalid	Available	Available
SSICR.TEN	Invalid	Available	Available	Invalid	Available	Available
SSICR.REN	Available	Invalid	Available	Available	Invalid	Available
SSIFCR.AUCKEN	Invalid	Invalid	Invalid	Available	Available	Available
SSIFCR.BCKNCE	Available	Available*1	Available*2	Invalid	Invalid	Invalid
SSIFCR.LRCKNCE	Available	Available*1	Available*2	Invalid	Invalid	Invalid
SSIFCR.RXDNCE	Available	Invalid	Available	Invalid	Invalid	Invalid
SSIFCR.TIE	Invalid	Available	Available	Invalid	Available	Available
SSIFCR.RIE	Available	Invalid	Available	Available	Invalid	Available
SSIFCR.TFRST	Invalid	Available	Available	Invalid	Available	Available
SSIFCR.RFRST	Available	Invalid	Available	Available	Invalid	Available
SSIOFR.BCKASTP	Invalid	Invalid	Invalid	Available	Available	Available
SSIOFR.LRCONT	Invalid	Invalid	Invalid	Available	Available	Available
SSIOFR.OMOD	Available	Available	Available	Available	Available	Available
SSISCR.TDES	Invalid	Available	Available	Invalid	Available	Available
SSISCR.RDFS	Available	Invalid	Available	Available	Invalid	Available

Note 1. Disabled when SSICR.DEL = '1'
Note 2. Disabled during transmission (SSICR.TEN = '1') when SSICR.DEL = '1'

24.4.3.1 Slave-mode Communication

SSIF-2 operates in slave mode with SSICR.MST = 0. Supply the SSIBCK signal and the SSILRCK/SSIFS signal used for the serial data communication from external devices. If these signals do not match the communication format set for SSIF-2, operation is not guaranteed.

24.4.3.2 Master-mode Communication

SSIF-2 operates in master mode with SSICR.MST = 1. The SSIBCK signal and SSILRCK/SSIFS signal used for the serial data communication are internally generated from the audio clock. These signals use the format according to the setting of SSIF-2. If the communication format of another device working as the slave device does not match the communication format set for SSIF-2, operation is not guaranteed.

24.4.3.3 Transmission

SSIF-2 transmits serial data to opposing device when SSICR.TEN = 1 and SSICR.REN = 0. If the communication format of the opposing device does not match the communication format set for SSIF-2, operation is not guaranteed.

24.4.3.4 Reception

SSIF-2 receives serial data from opposing device when SSICR.TEN = 0 and SSICR.REN = 1. If the communication format of the opposing device does not match the communication format set for SSIF-2, operation is not guaranteed.

24.4.3.5 Transmission and Reception

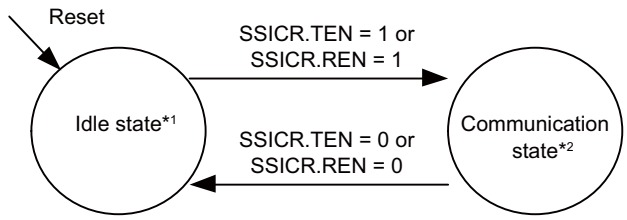
SSIF-2 transmits and receives serial data to and from opposing device when SSICR.TEN = 1, SSICR.REN = 1. If the communication format of opposing device does not match the communication format set for SSIF-2, operation is not guaranteed.

24.5 Operation

24.5.1 Operational State

SSIF-2 has the following two main operation states (Figure 24.42).

- Idle state (SSISR.IIRQ = 1)
- Communication state (SSISR.IIRQ = 0)



Notes: 1. See section 24.5.1.1 for details of the idle state.
2. See section 24.5.1.2 for details of the communicationstate.

Figure 24.42 SSIF-2 State Transition

24.5.1.1 Idle State

In this state, communication of SSIF-2 is halted. However, when SSICR.MST = 1, output of BCK and LRCK/FS to external pins is enabled depending on the settings in the SSIOFR.BCKASTP bit and the SSIOFR.LRCONT bit (Table 24.16). This function is common to all formats.

Table 24.16 Output from External Pins in the Idle State

			Output from Pins		
SSICR.MST	SSIOFR.BCKASTP	SSIOFR.LRCONT	SSIBCK	SSILRCK/SSIFS	SSITxD
0	—	—	Stop	Stop	Stop
1	0	0	Supply	Stop	Stop
1	0	1	Supply	Supply	Stop
1	1	0	Stop	Stop	Stop
1	1	1	Stop	Supply	Stop

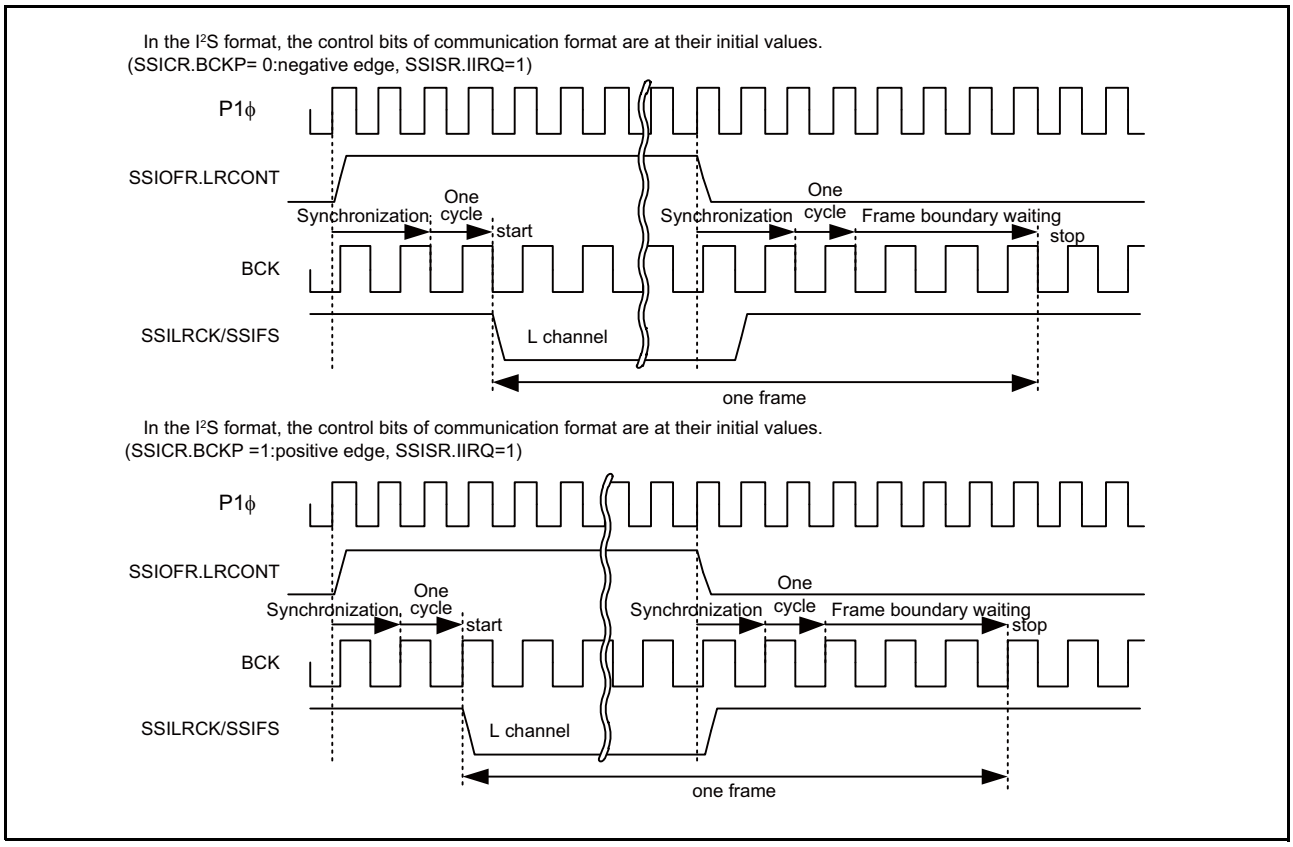


Figure 24.43 Example of LRCK/FS Continuation Release with SSIOFR.LRCONT

Note: To stop the output to the SSILRCK/SSIFS pin with SSIOFR.LRCONT when SSIF-2 is in the idle state in master-mode communication (SSICR.MST = 1), note the following: The output stops when 0 is written from 1 to SSIOFR.LRCONT. (Figure 24.43). Make sure that the remote device is not affected.

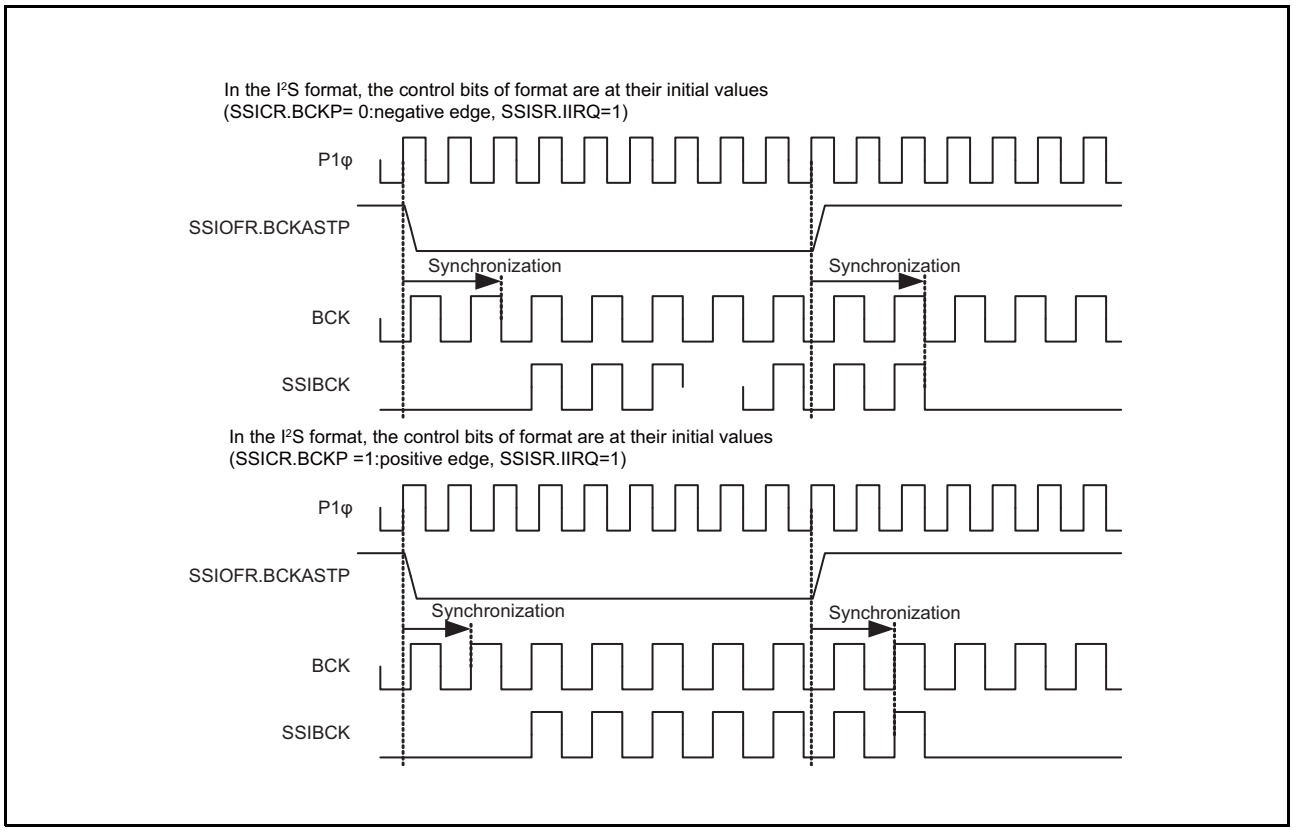


Figure 24.44 Example of Stopping SSIBCK with SSIOFR.BCKASTP

Note: To stop the output to the SSIBCK pin with SSIOFR.BCKASTP in master-mode communication (SSICR.MST = 1) and while SSIF-2 is in the idle state, note the following; By writing 0 to SSIOFR.BCKASTP while it is 1, the output stops immediately (Figure 24.44). So, make sure that the remote device is not affected.

24.5.1.2 Communication States

In this state, SSIF-2 is during communication.Figure 24.45 shows transitions of communication states and Table 24.17 lists the conditions for transition. If the transition condition is not satisfied, the state does not transit.

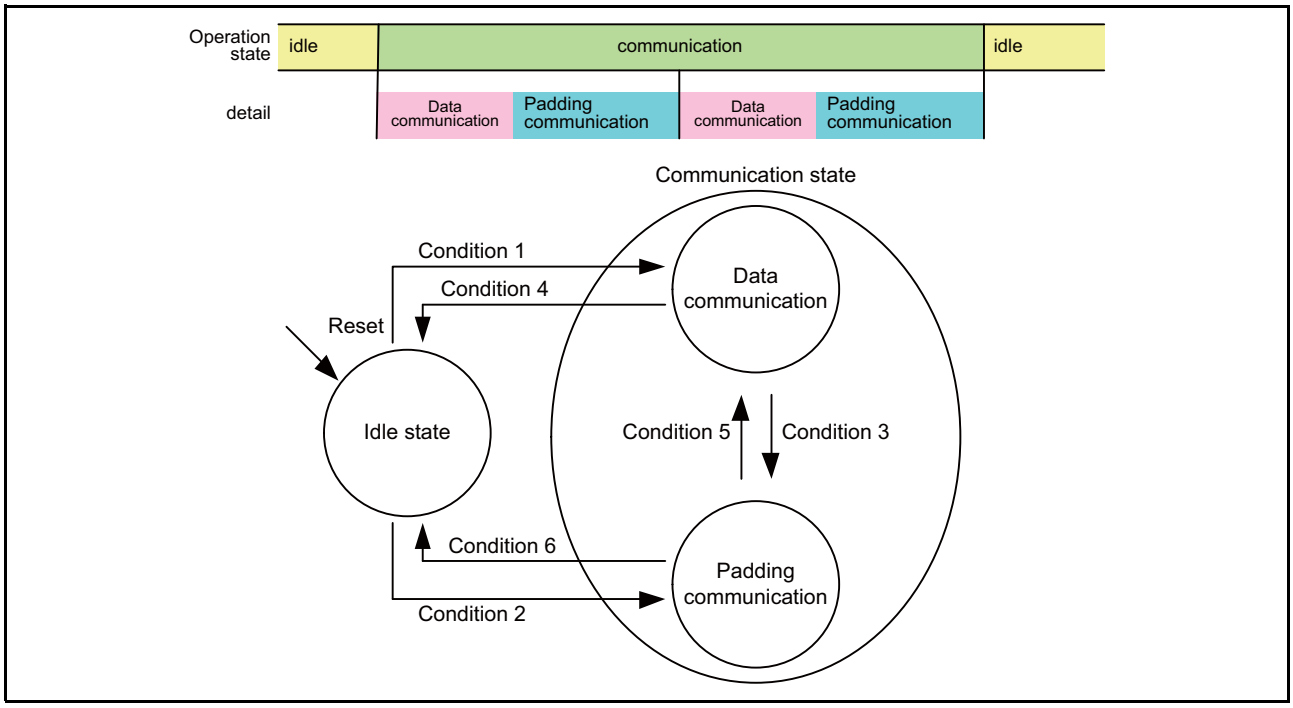


Figure 24.45 Communication State Transition

Table 24.17 Condition for Communication State Transition

Condition No.	Condition for Transition
1	Writing SSICR.TEN = 1 or SSICR.REN = 1 while SSICR.SDTA = 0 or in the setting without padding bits.
2	Writing SSICR.TEN = 1 or SSICR.REN = 1 while SSICR.SDTA = 1 and in the setting with padding bits.
3	The following all of three conditions are satisfied: • SSICR.TEN = 1 or SSICR.REN = 1 • In the setting without padding bits • Transfer data words of the last data bit is completed.
4	The following all of two conditions are satisfied: • SSICR.SDTA = 1 or without padding bits • Transfer data words of the last data bit of one frame is completed while SSICR.TEN = 0 and SSICR.REN = 0
5	Transfer of the last padding bit is completed while SSICR.TEN = 1 or SSICR.REN = 1
6	The following all of two conditions are satisfied: • SSICR.SDTA = 0 and with padding bits • Transfer of the last data bit of one frame is completed while SSICR.TEN = 0 and SSICR.REN = 0.

See Table 24.13 for the setting with/without padding bits.

(1) Data Communication State

In this state, SSIF-2 is during communication. Data of the data word length set with the SSICR.DWL[2:0] bits is transmitted, received, or transmitted and received.

- State Transition in the Setting without Padding Bits

During communication (SSISR.IIRQ = 0), SSIF-2 is during data communication for all the time (Figure 24.46). By disabling transmission and reception (SSICR.TEN = 0 and SSICR.REN = 0), SSIF-2 transits to the idle state (Figure 24.47).

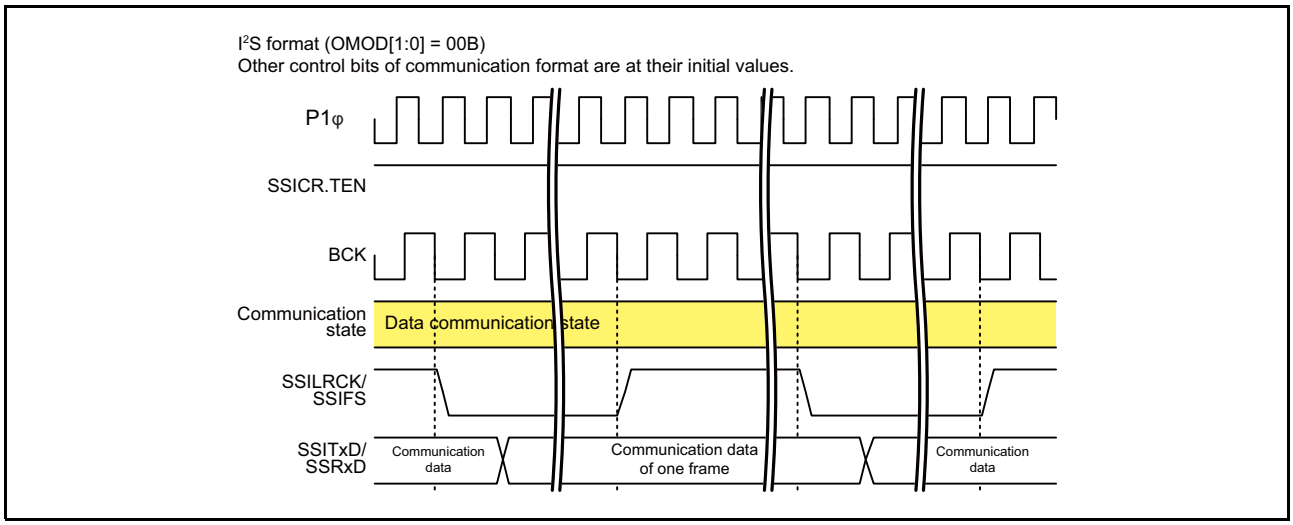


Figure 24.46 Continuation of the Data Communication State

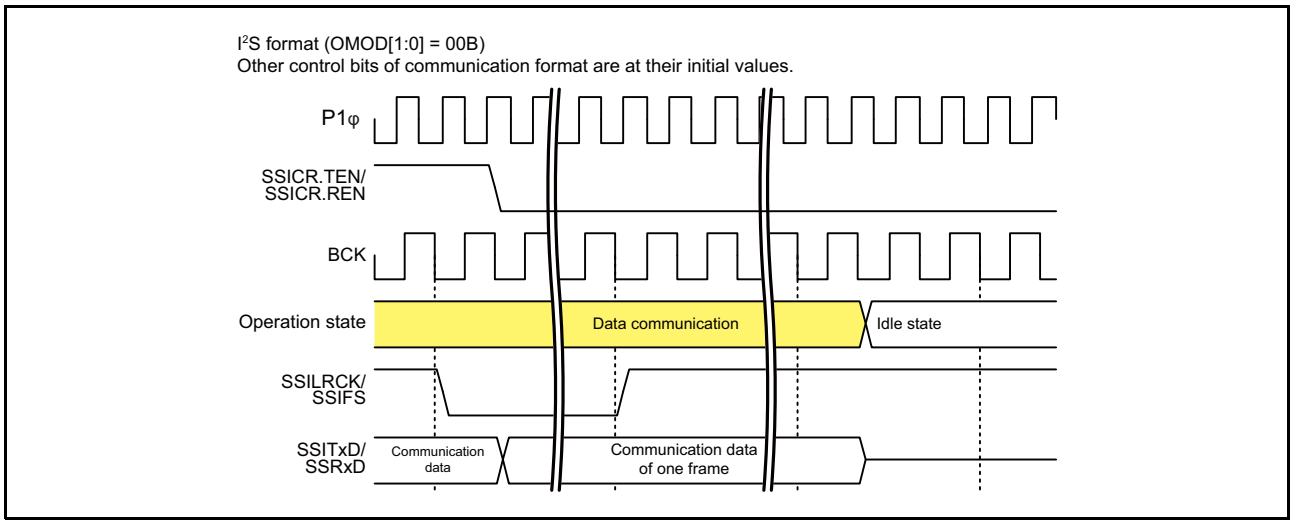


Figure 24.47 Halt from the Data Communication State (without Padding Bits)

- State Transition in the Setting with Padding Bits

During communication (SSISR.IIRQ = 0), after completing transfer of the last bit of a data word, SSIF-2 transits to the padding communication state from data communication state (Figure 24.48). By disabling transmission and reception (SSICR.TEN = 0 and SSICR.REN = 0) while SSICR.SDTA = 1, SSIF-2 transits to the idle state from data communication state when the communication stop is done. (Figure 24.50).

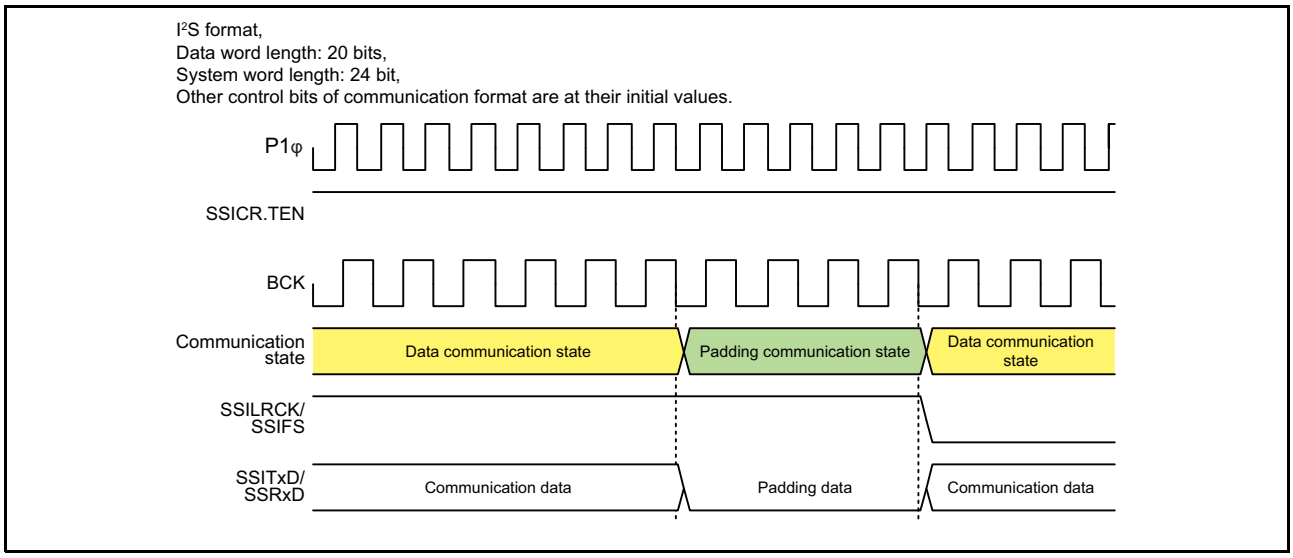


Figure 24.48 Transition from Data Communication to Padding Communication

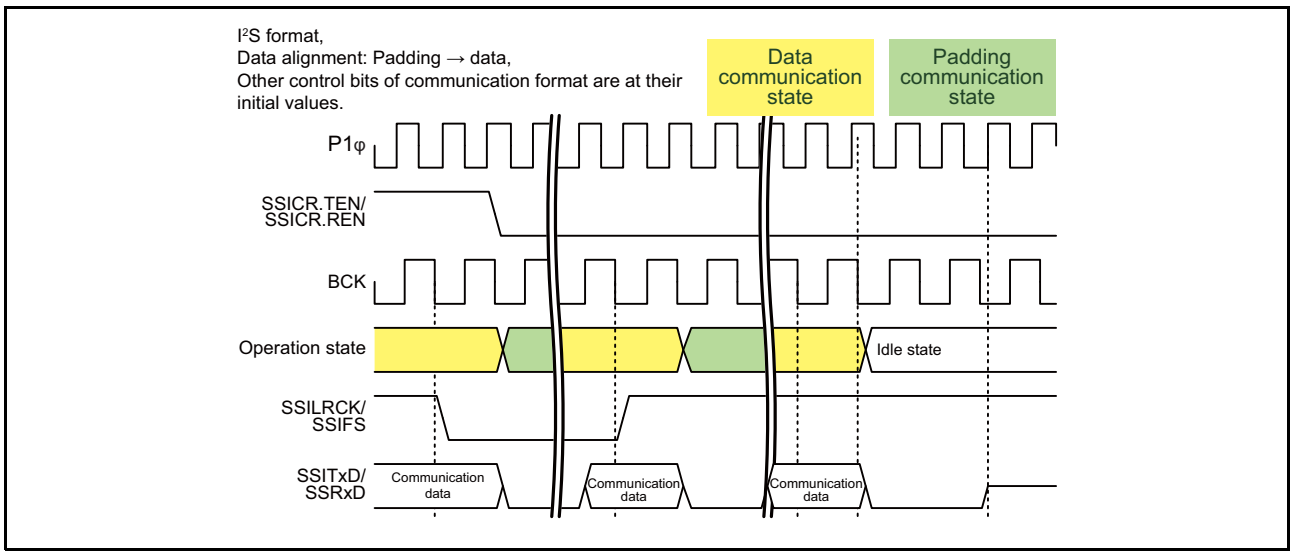


Figure 24.49 Halt from Data Communication (with Padding Bits)

(2) Padding Communication State

In this state, SSIF-2 is during communication. The padding bits set with the SSICR.SWL[2:0] bits and SSICR.DWL[2:0] bits are transmitted, received, or transmitted and received.

- State Transition in the Setting with Padding Bits

During communication (SSISR.IIRQ = 0), after completing transfer of the last padding bit, SSIF-2 transits to data communication state (Figure 24.48). By disabling transmission and reception (SSICR.TEN = 0 and SSICR.REN = 0) while SSICR.SDTA = 0, SSIF-2 transits to the idle state from padding communication when the communication stop is done. (Figure 24.50).

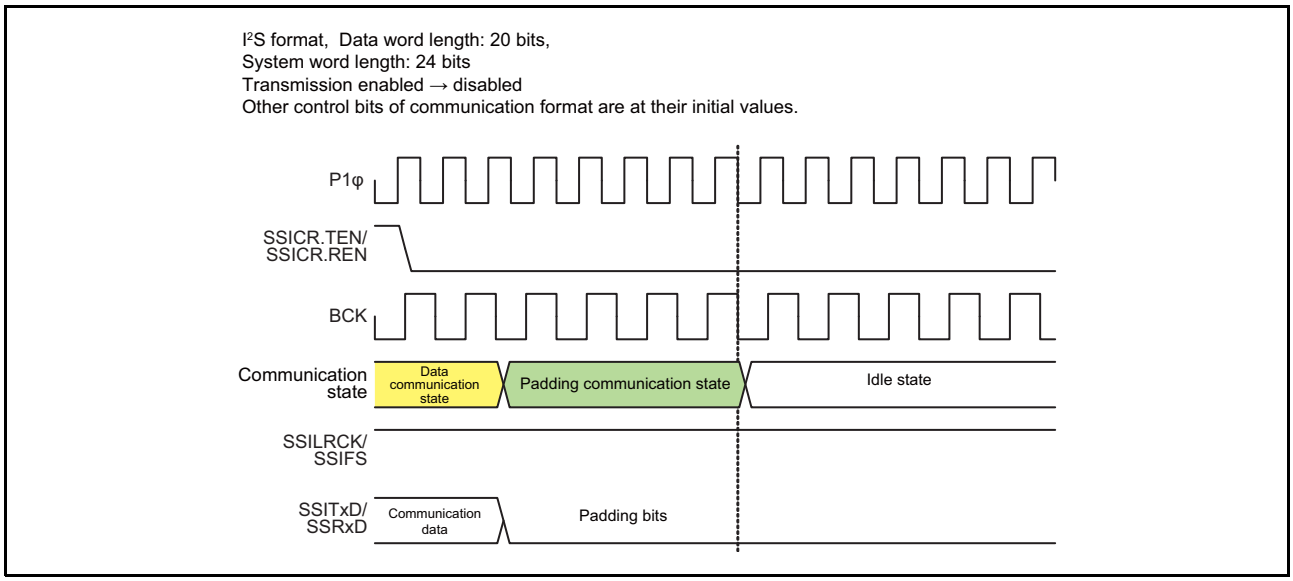


Figure 24.50 Halt from the Padding Communication State

24.5.2 Communication Operation

Figure 24.51 shows the communication flow of SSIF-2.

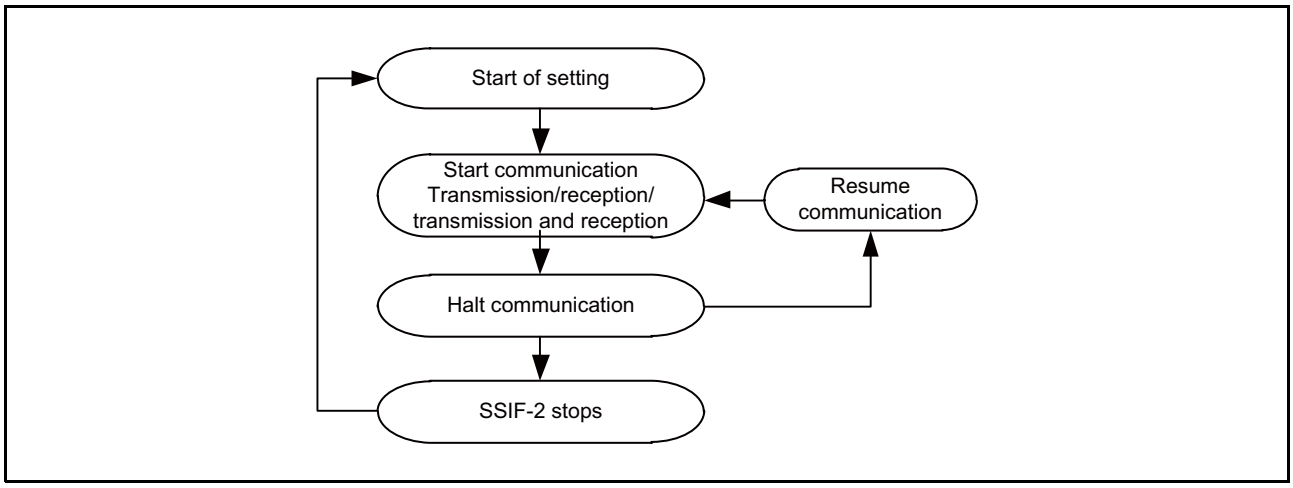


Figure 24.51 SSIF-2 Communication Operation

The procedure of each operation is described in section 24.5.2.1, Start Communication to section 24.5.2.7, Resume Communication.

24.5.2.1 Start Communication

This section describes how to start communication of SSIF-2. Figure 24.52 shows the procedure to start communication. Be sure to follow the procedure. See section 24.5.2.2 for transmission operation and section 24.5.2.3 for reception operation.

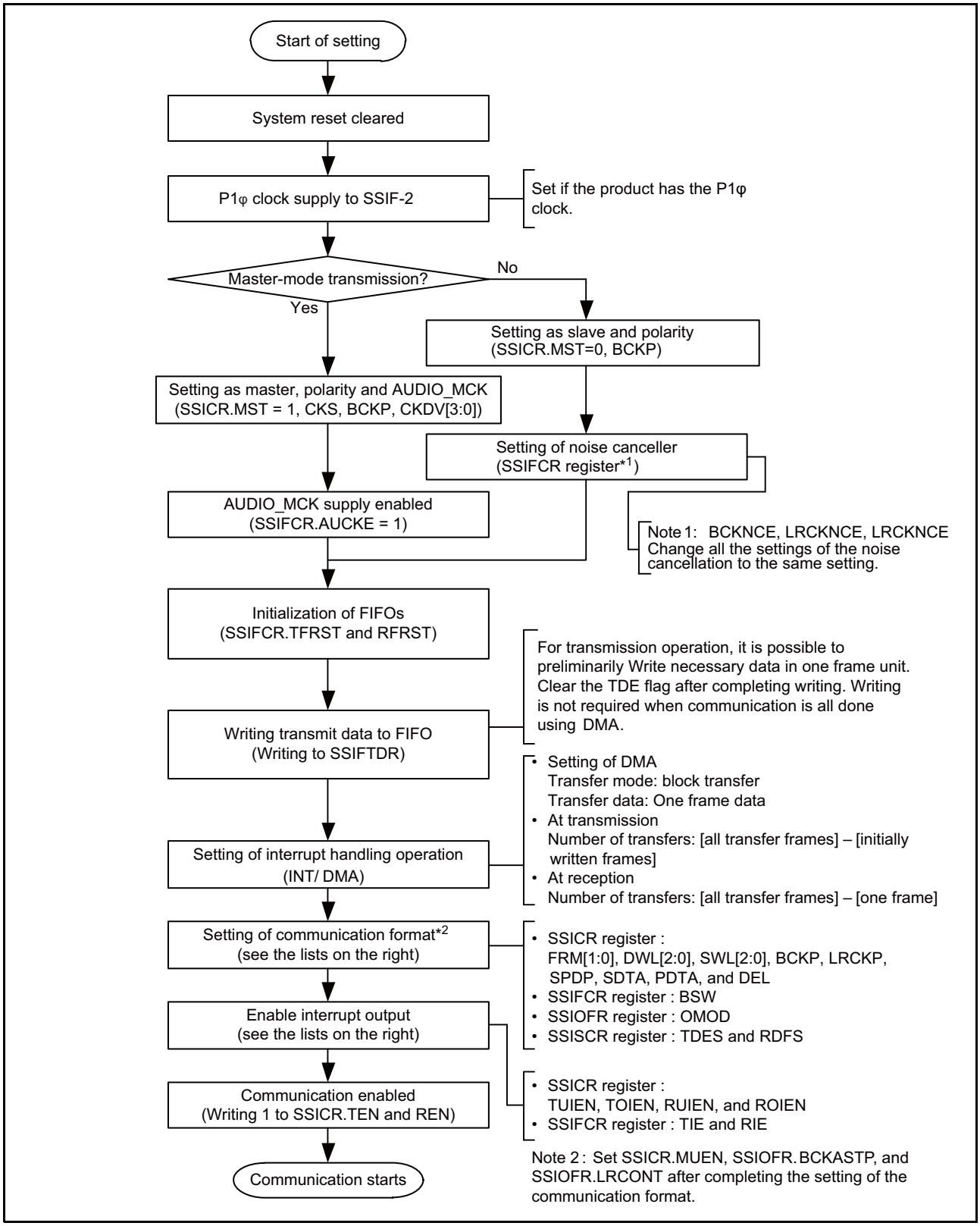


Figure 24.52 Procedure to Start Communication (CPU Procedure)

For SSIF-2 communication, continuous communication is possible with DMA interrupt handling. For transmission, write 1 to SSIFCR.TIE, SSICR.TUIEN, and SSICR.TOIEN. For reception, write 1 to SSIFCR.RIE, SSICR.RUIEN, and SSICR.ROIEN.

24.5.2.2 Transmission

Follow the transmission operation procedure (Figure 24.53) while the transmission is operating.
Transmission starts in synchronization with the start trigger of the SSILRCK/SSIFS signal while transmission is enabled (SSICR.TEN = 1 and SSICR.REN = 0) and while there is one or more frame of serial data in the transmit FIFO data register (SSIFTDR).
SSIF-2 outputs a transmit data empty interrupt to DMA according to the setting condition of TDE (SSISCR.TDES) and transmit data empty interrupt (SSIFCR.TIE) being enabled.
This interrupt requests writing to the transmit FIFO data register (SSIFTDR). By writing to SSIFTDR at each generation of an interrupt, serial data transmission is realized.
In the procedure to start communication, set DMA handling for transmit data empty interrupts as writing to the transmit FIFO data register (SSIFTDR). As a result, SSIF-2 is able not to mediate CPU and to transmit continuous data. When it becomes empty capacity of transmission FIFO data register set with SSISCR.TDES, transmission data empty interrupt is generated. Set the writing number of times according to empty capacity of transmission FIFO data register that transmission data empty interrupt shows. In a case of an error, stop receive procedure (section 24.5.2.5) and then take the error procedure (section 24.5.2.6).

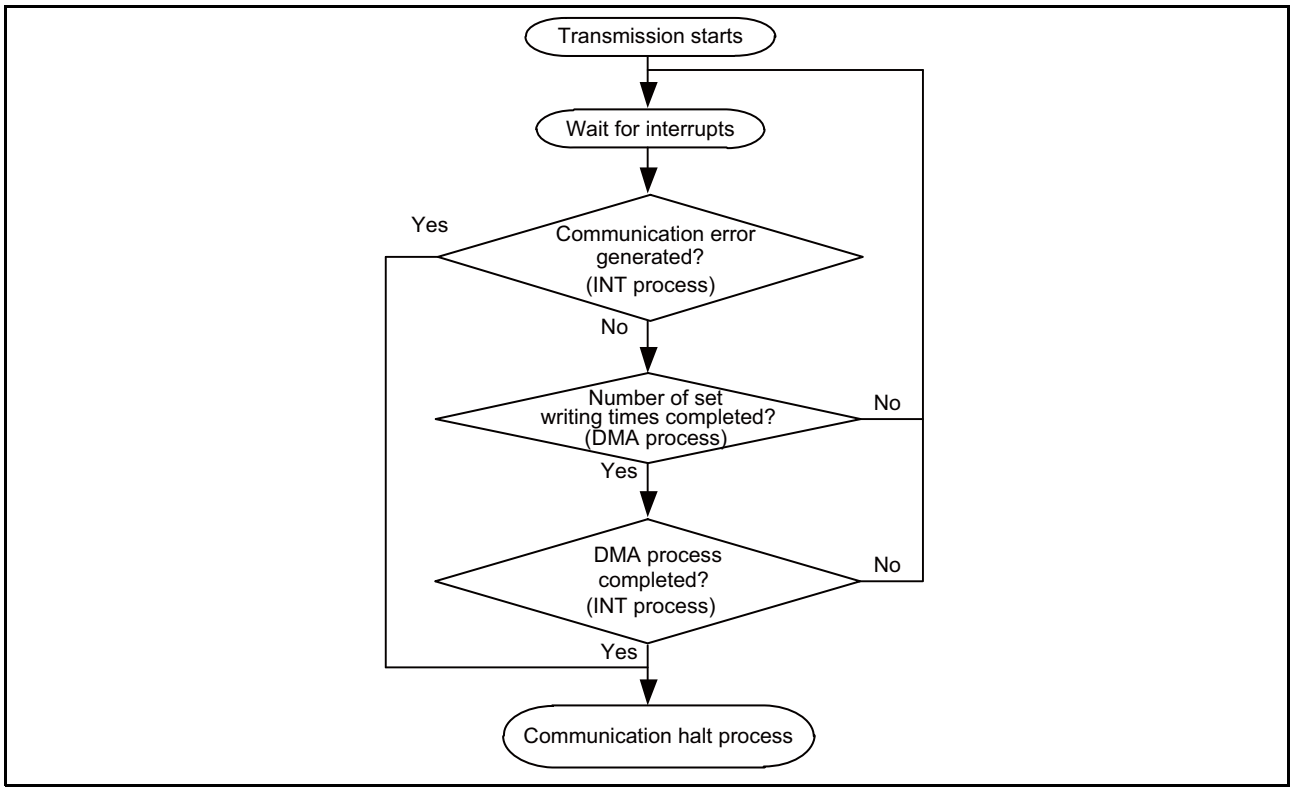


Figure 24.53 Transmission Procedure

Note: The communication flow defined in SSIF-2 uses the DMA. In the case the DMA is not used, observe one of SSIFSR.TDE, and write it in SSIFTDR. Follow empty capacity of transmission FIFO data register set with SSISCR.TDES about the number of times written in SSIFTDR detecting 1 of SSIFSR.TDE. Clear the transmission data that corresponds to empty capacity and clear the SSIFSR.TDE flag to SSIFTDR after writing. Transmission operation consecutive by the repeated thing is possible. SSIFSR.TDE is not cleared when not clearing.

24.5.2.3 Reception

Follow the reception operation procedure (Figure 24.54) while the reception is operating.
Reception starts in synchronization with the start trigger of the SSILRCK/SSIFS signal while reception is enabled (SSICR.TEN = 0 and SSICR.REN = 1). SSIF-2 outputs a receive data full interrupt to DMA according to the setting of RDF (SSISCR.RDFS) and receive data empty interrupt (SSIFCR.RIE) being enabled. This interrupt requests reading from the receive FIFO data register (SSIFRDR). By reading from SSIFRDR at each generation of an interrupt, serial data reception is realized.
In the procedure to start communication, set DMA handling for receive data full interrupts as reading from receive FIFO data register (DDIFRDR). As a result, SSIF-2 is able not to mediate CPU and to receive continuous data. If the data capacity of reception FIFO data register set with SSISCR.RDFS is stored, receive data full interrupt is generated. Set the writing number of times according to the data capacity of reception FIFO data register that receive data full interrupt shows.
In the case of an error, stop the receive operation (section 24.5.2.6) and then take the error procedure (section 24.5.2.6).

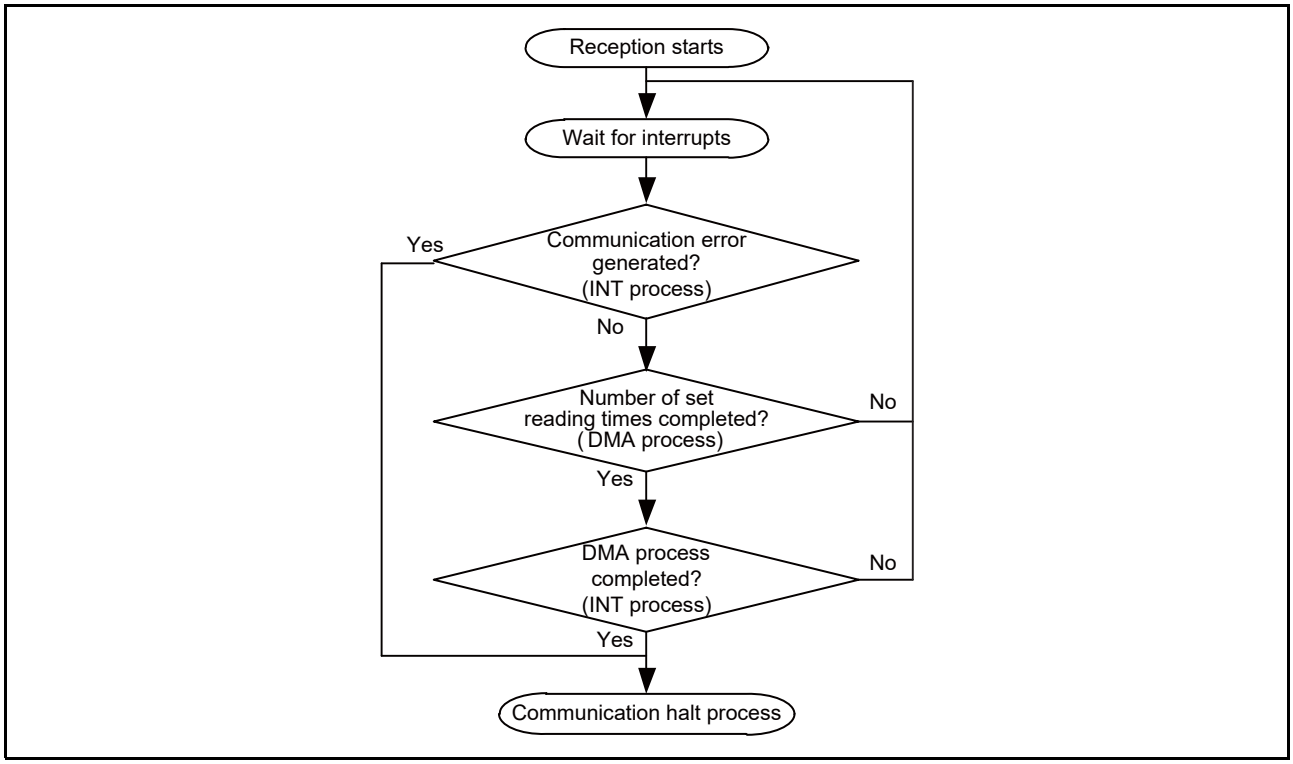


Figure 24.54 Reception Procedure

Note: The communication flow defined in SSIF-2 uses the DMA. In the case the DMA is not used, observe 1 of SSIFSR.RDF, and read SSIFRDR. Follow the receive data capacity of reception FIFO data register set with SSISCR.RDFS about the number of times in which SSIFRDR is read detecting 1 of SSIFSR.RDF. Clear data that does the reception completion and clear the SSIFSR.RDF flag from SSIFRDR after reading. Reception operation consecutive by the repeated thing is possible. SSIFSR.RDF is not cleared when not clearing.

24.5.2.4 Transmission and Reception

Transmission and reception starts in synchronization with the start trigger of the SSILRCK/SSIFS signal while transmission and reception are enabled (SSICR.TEN = 1 and SSICR.REN = 1) and while there is one or more frames of serial data in the transmit FIFO data register (SSIFTDR). The transmission and reception operation of SSIF-2 can be transmitted and received continuously by each doing the procedure for showing in the transmission operation (section 24.5.2.2) and the reception operation (section 24.5.2.3). As transmission and reception are independent in SSIF-2, see sections of transmission and reception respectively. See section 24.5.2.5, Halt Communication to halt communication.

24.5.2.5 Halt Communication

This section describes how to halt communication of SSIF-2. Figure 24.55 shows the procedure to halt communication. Be sure to follow the procedure.

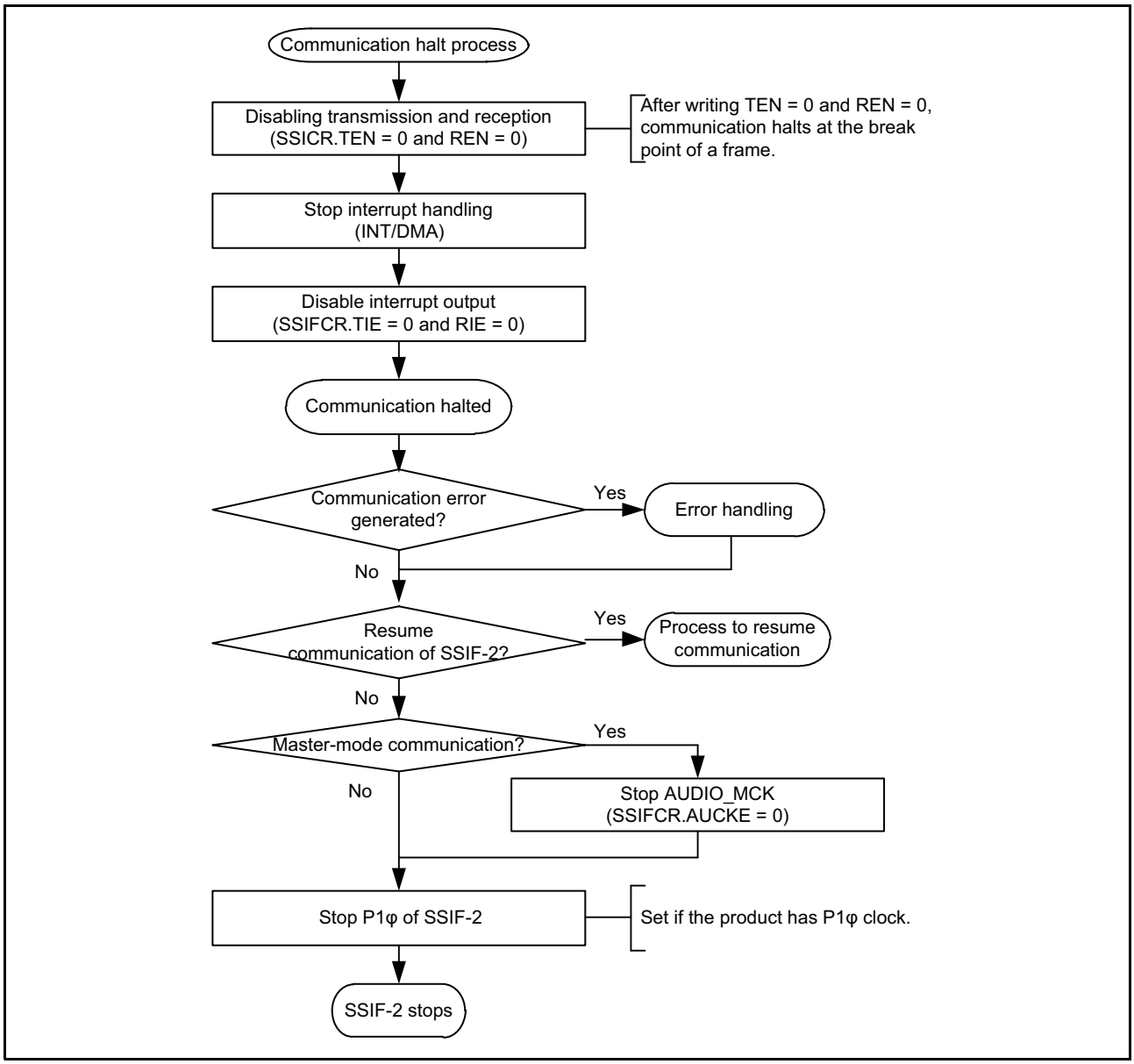


Figure 24.55 Procedure to Halt Communication (CPU Procedure)

To halt the communication of SSIF-2, supply of the following clocks are required until the SSISR.IIRQ bit indicates an idle state.

- Input clock from the SSIBCK pin when SSICR.MST = 0
- AUDIO_MCK when SSICR.MST = 1

To resume communication of SSIF-2 in the previous setting, see section 24.5.2.7, Resume Communication.

Note: When communication of SSIF-2 is halted according to the procedure to halt communication in Figure 24.55, resume communication according to the procedure to start communication in Figure 24.52.

24.5.2.6 Error Handling

SSIF-2 has the following four errors.

- Transmit underflow error
- Transmit overflow error
- Receive underflow error
- Receive overflow error

When an underflow error or overflow error is generated, SSIF-2 need to be restarted. Take the error procedure (Figure 24.56) according to the procedure to halt communication (Figure 24.55).

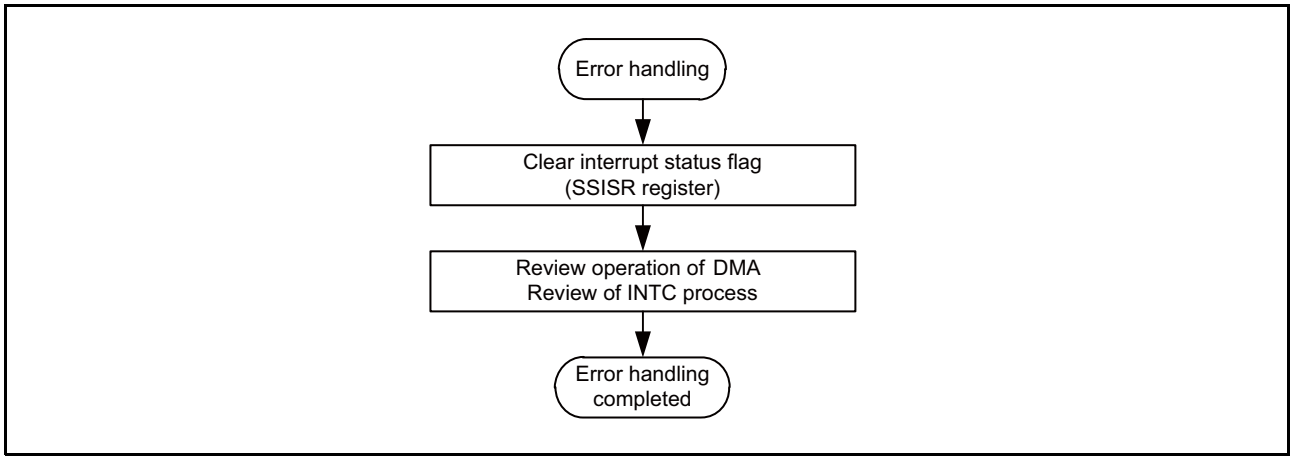


Figure 24.56 Error Handling Procedure

Four error operations are described as follows. When the interrupt output enable bit of the SSICR register is enabled and error flags are set, an error interrupt is generated. See descriptions of flags in section 24.4.1.2, Status Register (SSISR) for the setting conditions of error flags.

(1) Transmit Underflow Error

In the case a transmit underflow error is generated during transmission (SSICR.TEN = 1, SSISR.IIRQ = 0), review the number of write operations to the transmit FIFO data register (SSIFTDR) according to serial data empty interrupts. After generation of a transmit underflow error, data to be transmitted from SSIF-2 are 0 output. To transmit data written to SSIFTxD pin normally, run the procedure to halt communication (Figure 24.55) and then take the error procedure (Figure 24.56). When this error is generated, serial data is consumed as usual. To resume transmission of SSIF-2, write data from the beginning of one frame.

(2) Transmit Overflow Error

In the case a transmit overflow error is generated, review the number of write operations to the transmit FIFO data register (SSIFTDR) according to transmit data empty interrupts. Data is not written to SSIFTDR where a transmit overflow error is generated. This error is generated regardless of while the transmission is operating. To resume transmission of SSIF-2, take the procedure to halt communication (Figure 24.55) and then take the error procedure (Figure 24.56). When communication is resumed, consider the lost serial data.

(3) Receive Underflow Error

In the case a receive underflow error is generated, review the number of reading operations from the receive FIFO data register (SSIFRDR) according to receive data full interrupts. The value read from SSIFRDR is undefined where a receive underflow error is generated. This error is generated regardless of while the reception is operating. To resume reception

of SSIF-2, take the procedure to halt communication (Figure 24.55) and then take the error procedure (Figure 24.56).

(4) Receive Overflow Error

In the case a receive overflow error is generated during reception (SSICR.REN = 1, SSISR.IIRQ = 0), review the number of reading operations from the receive FIFO data register (SSIFRDR) according to receive data full interrupts. Data is not stored in SSIFRDR when a receive overflow error is generated. To resume reception of SSIF-2, take the procedure to halt communication (Figure 24.55) and then take the error handling procedure (Figure 24.56).

24.5.2.7 Resume Communication

To resume communication of SSIF-2, follow the procedure shown in Figure 24.57. After communication is halted, the same setting is applied when it is resumed. To change the settings of clocks and slave/master mode, follow the procedure to start communication (Figure 24.52). For communication after resume, see section 24.5.2.2 and section 24.5.2.3.

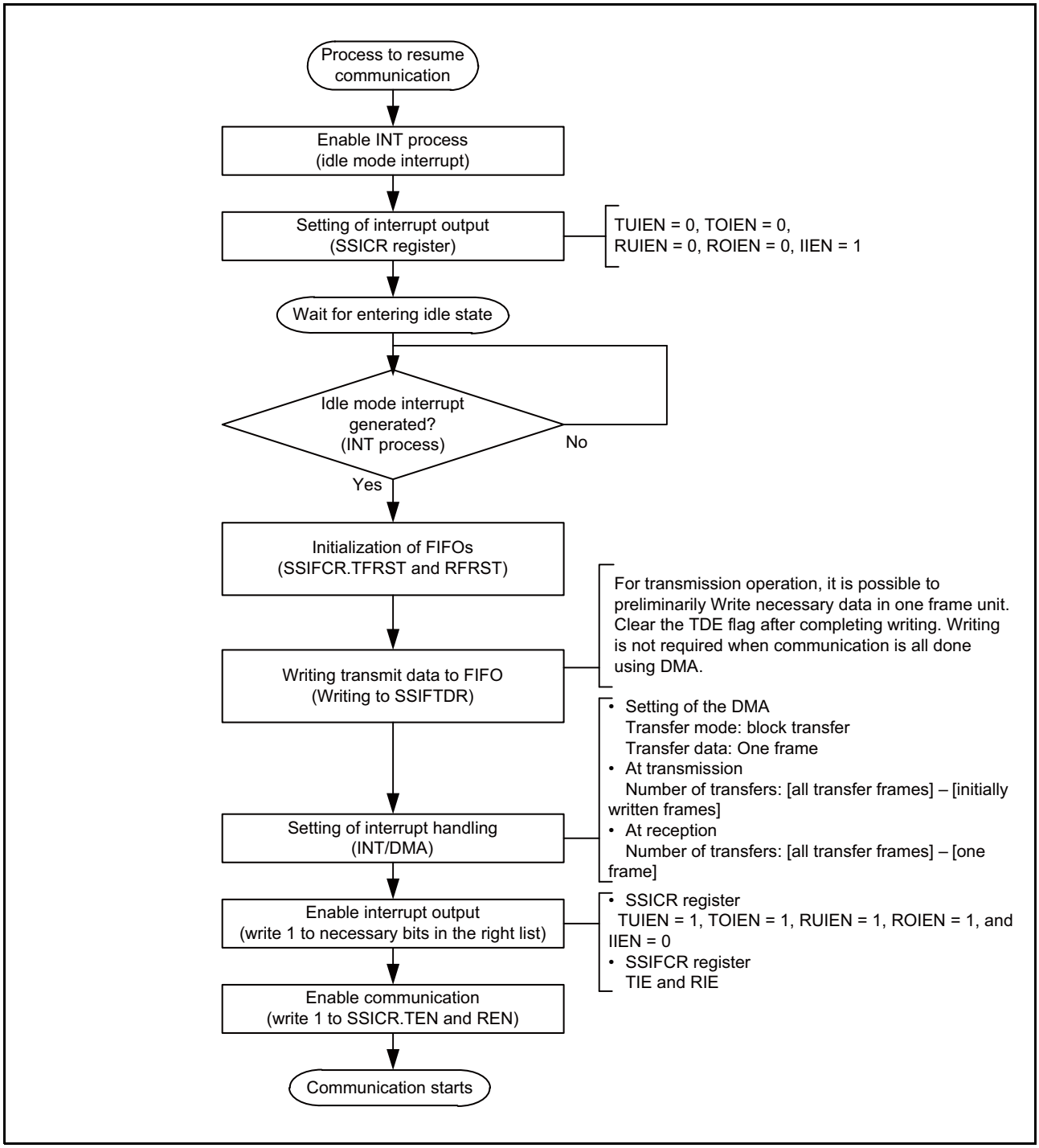


Figure 24.57 Procedure to Resume Communication (CPU Procedure)

24.5.3 Interrupt Sources

Table 24.18 lists the interrupt sources. Set enable/disable of interrupt output of each source with the TUIEN, TOIEN, RUIEN, ROIEN, and IIEN bits in the SSICR register and the TIE and RIE bits in the SSIFCR register.

Table 24.18 Interrupt Sources

Interrupt Source	Interrupt	Interrupt Flag	Interrupt Sense	DMA Activation
INT_ssif_int_req	Transmit underflow interrupt/ Transmit overflow interrupt/ Receive underflow interrupt/ Receive overflow interrupt/ Idle mode interrupt	SSISR.TUIRQ SSISR.TOIRQ SSISR.RUIRQ SSISR.ROIRQ SSISR.IIRQ	Level	Not available
INT_ssif_dma_tx	Transmit data empty interrupt	SSIFSR.TDE	Edge	Available
INT_ssif_dma_rx	Receive data full interrupt	SSIFSR.RDF	Edge	Available
INT_ssif_dma_rt	Receive data full interrupt/ Transmit data empty interrupt	SSIFSR.TDE/SSIFSR.RDF	Edge	Available

24.5.3.1 INT_ssif_int_req Interrupt

This interrupt source combines five interrupts. Enable output of necessary interrupts before using SSIF-2. This interrupt moves from the flag and the output enable permission that five factors have respectively (Figure 24.58). To clear an interrupt, set the interrupt enable to 0 or clear the interrupt flag to 0.

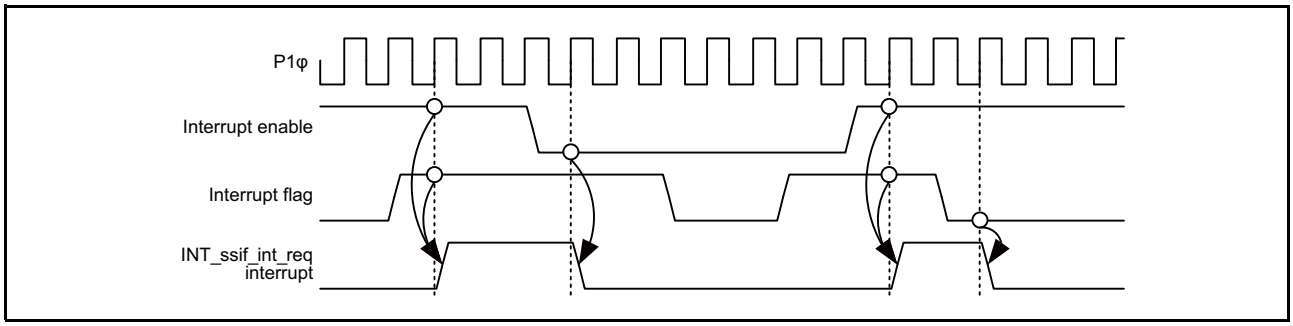


Figure 24.58 Timing Chart of the Common Interrupt Source, INT_ssif_int_req

(1) Transmit underflow interrupt

As the transmit underflow interrupt, SSISR.TUIRQ is output while SSICR.TUIEN = 1. To restart transmission (SSICR.TEN = 1), enable the output of this interrupt (SSICR.TUIRQ = 1). In the case this interrupt is generated, take the procedure to halt communication (Figure 24.55) and then take the error procedure (Figure 24.56).

(2) Transmit overflow interrupt

As the transmit overflow interrupt, SSISR.TOIRQ is output while SSICR.TOIRQ = 1. To restart transmission (SSICR.TEN = 1), enable the output of this interrupt (SSICR.TOIRQ = 1). In the case this interrupt is generated, take the procedure to halt communication (Figure 24.55) and then take the error procedure (Figure 24.56).

(3) Receive underflow interrupt

As the receive underflow interrupt, SSISR.RUIRQ is output while SSICR.RUIRQ = 1. To restart reception (SSICR.REN = 1), enable the output of this interrupt (SSICR. RUIRQ = 1). In the case this interrupt is generated, take the procedure to halt communication (Figure 24.55) and then take the error procedure (Figure 24.56).

(4) Receive overflow interrupt

As the receive overflow interrupt, SSISR.ROIRQ is output while SSICR.ROIRQ = 1. To restart reception (SSICR.REN = 1), enable the output of this interrupt (SSICR. ROIRQ = 1). In the case this interrupt is generated, take the procedure to halt communication (Figure 24.55) and then take the error procedure (Figure 24.56).

(5) Idle mode interrupt

As the idle mode interrupt, SSISR.IIRQ is output while SSICR.IIEN = 1. When the thing that the communication has stopped completely is confirmed, this interrupt is used. Figure 24.57).

24.5.3.2 INT_ssif_dma_tx Interrupt [full-duplex communication]

As the transmit data empty interrupt, a pulse interrupt is output by the following condition.

- SSIFSR.TDE and SSIFCR.TIE becomes 1.
  SSIF-2 operation: SSIFSR.TDE becomes 1 from 0 while SSIFCR.TIE = 1
  CPU operation: SSIFCR.TIE is changed to 1 from 0 while SSIFSR.TDE = 1

An interrupt suppression function is available for this interrupt to prevent the thing that cannot be taken. When the DMA is busy state (Interrupt cannot be accepted), and the interruption condition occurs, output of this interrupt is held. This interrupt is output after the DMA becomes acceptable (Figure 24.59).

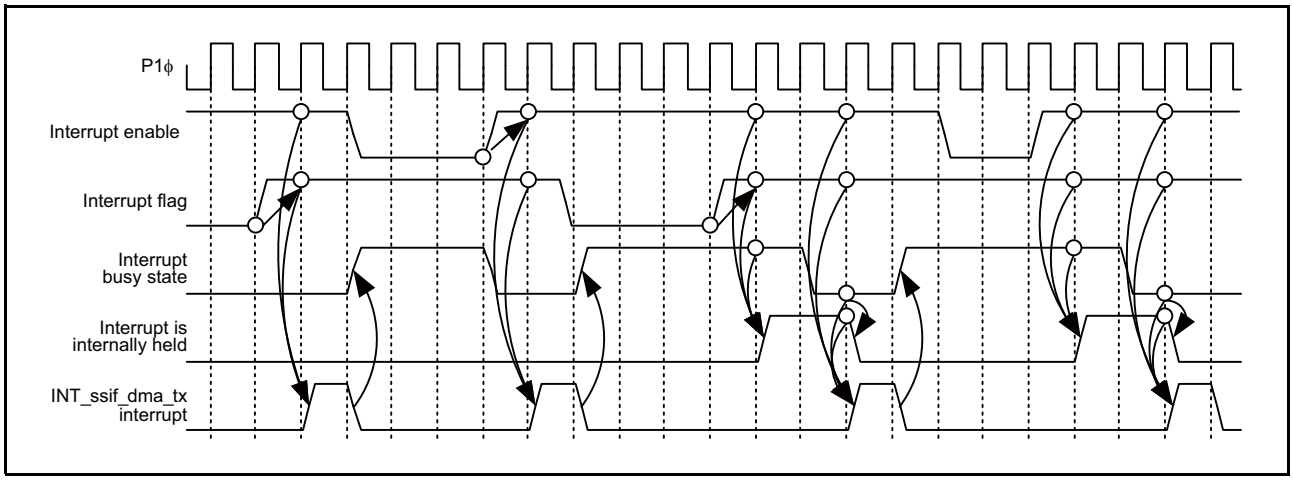


Figure 24.59 INT_ssif_dma_tx Interrupt Timing Chart

24.5.3.3 INT_ssif_dma_rx Interrupt [full-duplex communication]

As the receive data full interrupt, a pulse interrupt is output by the following condition.

- SSIFSR.RDF and SSIFCR.RIE becomes 1.
  SSIF-2 operation: SSIFSR.RDF becomes 1 from 0 while SSIFCR.RIE = 1.
  CPU operation: SSIFCR.RIE is changed to 1 from 0 while SSIFSR.RDF = 1.

An interrupt suppression function is available for this interrupt to prevent the thing that cannot be taken. When the DMA is busy state (Interrupt cannot be accepted), and the interruption condition occurs, output of this interrupt is held. This interrupt is output after the DMA becomes acceptable. It operates in the same way as in Figure 24.59.

24.5.3.4 INT_ssif_dma_rt Interrupt [half-duplex communication]

This interrupt is output by two sources, transmit data empty interrupt and receive data full interrupt. When this interrupt is generated, read the interrupt flag and specify the interrupt source.
An interrupt suppression function is available for this interrupt to prevent the thing that cannot be taken. When the DMA is busy state (Interrupt cannot be accepted), and the interruption condition occurs, output of this interrupt is held. This interrupt is output after the DMA becomes acceptable (Figure 24.59).

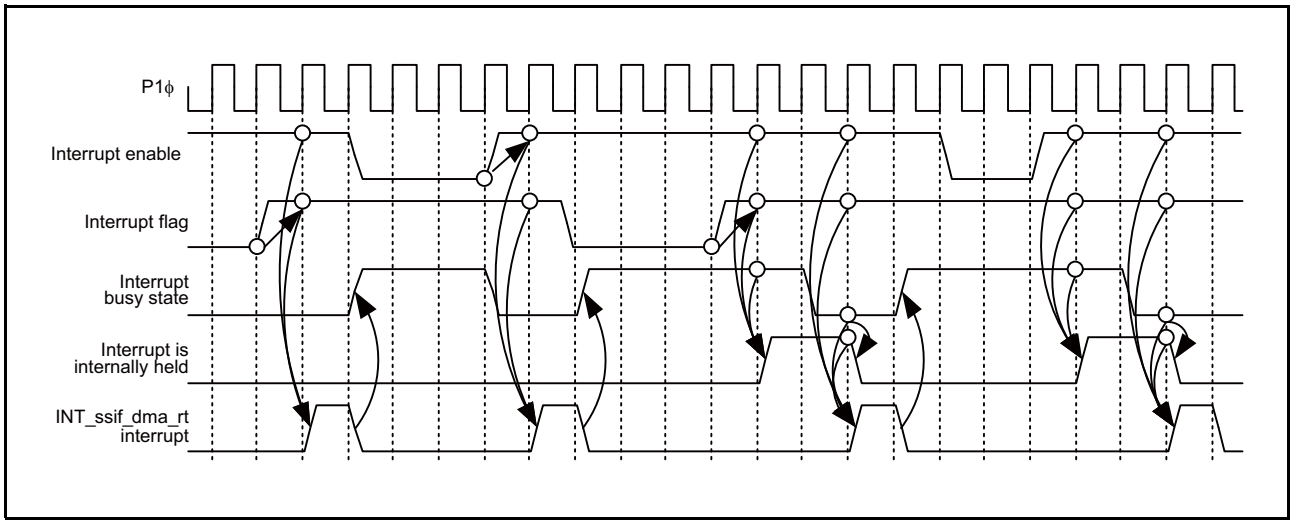


Figure 24.60 INT_ssif_dma_tx Interrupt Timing Chart

24.5.4 Software Resets

SSIF-2 has three software reset bits to reset its states.

- SSIF-2 software reset (SSIFCR.SSIRST)
- Transmit FIFO data register reset (SSICR.TFRST)
- Receive FIFO data register reset (SSIFCR.RFRST)

It explains the procedure of three software reset.

24.5.4.1 Software Reset Procedure

(1) SSIF-2 Software Reset

For the SSIF-2 software reset bit (SSIFCR.SSIRST), follow the procedure shown in Figure 24.61. After a reset, the same setting is applied when it is resumed. To change the settings of clocks and slave/master mode, follow the procedure to start communication (Figure 24.52). See section 24.5.2.2, Transmission and section 24.5.2.3, Reception respectively for transmission and reception after communication is resumed.

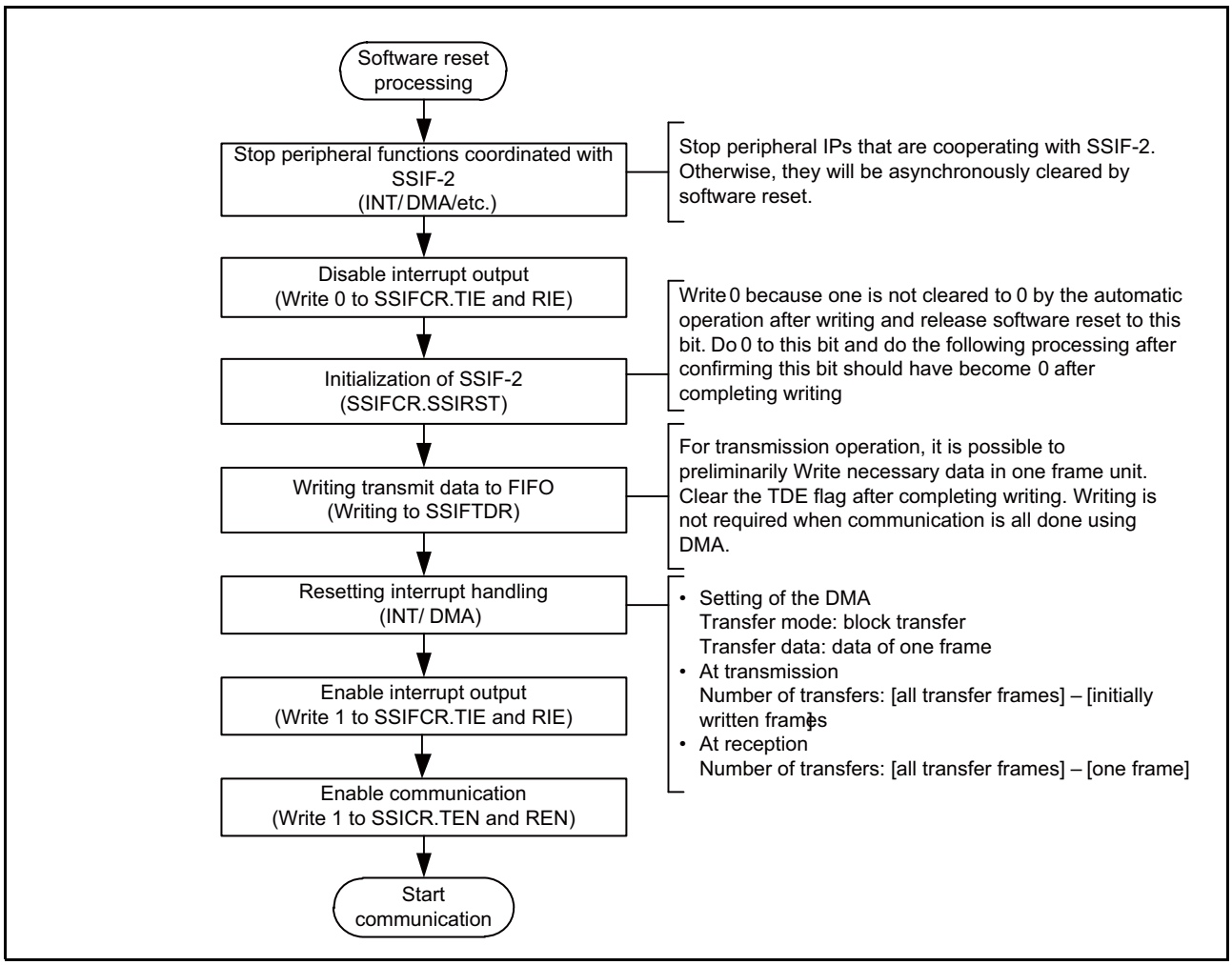


Figure 24.61 Software Reset Procedure (CPU Procedure)

(2) Transmit FIFO Data Register Reset

To initiate a transmit FIFO data register reset, follow the procedure to start communication (Figure 24.52) and resume communication (Figure 24.57).

(3) Receive FIFO Data Register Reset

To initiate a receive FIFO data register reset, follow the procedure to start communication (Figure 24.52) and resume communication (Figure 24.57).

24.6 Notes

24.6.1 Notes

24.6.1.1 Attention by communication using DMA

SSIF-2 assumes the DMA access as follows.

Table 24.19 Interrupt to expect DMA access

Communication operation	Interrupt	DMA access	LSI specification	Remarks
Transmission	INT_ssif_dma_tx	Write to SSIFTDR	Full duplex transmission	
	INT_ssif_dma_rt		Half duplex transmission	
Reception	INT_ssif_dma_rx	Read SSIFRDR	Full duplex transmission	
	INT_ssif_dma_rt		Half duplex transmission	
Transmission and Reception	INT_ssif_dma_tx	Write to SSIFTDR	Full duplex transmission	
	INT_ssif_dma_rx	Read SSIFRDR	Half duplex transmission	

Defend the following when communicating without using DMA.

- Clear SSIFSR.TDE when you access finality when it is writing in SSIFTDR with CPU. There is a possibility that the transmission overflow not intended occurs when not clearing.
- Clear SSIFSR.RDE when you access finality when you read SSIFRDR with CPU. There is a possibility that the reception overflow not intended occurs when not clearing.

24.6.1.2 Notes for Slave-mode Communication

(1) ADCKE Control

In slave-mode communication (SSICR.MST = 0), SSIF-2 needs supply of SSIBCK. To stop BCK on the master side, make sure that SSIF-2 is in the idle state (SSISR.IIRQ = 1). If BCK is stopped before SSIF-2 becomes idle, take the procedure to start communication (Figure 24.52) or wait for an idle state by taking the procedure to resume communication (Figure 24.57).

(2) SSILRCK/SSIFS

SSIF-2 has the terminal SSILRCK/SSIFS that shows the synchronization of the communication. Match communication format of the opposing device and SSIF-2 when SSIF-2 is slave communication (SSICR.MST=0). SSIF-2 uses the signal of SSILRCK/SSIFS only as a communication start trigger.

24.6.1.3 Notes for Master-mode Communication

(1) ADCKE Control

In master-mode communication (SSICR.MST = 1), SSIF-2 operates with the audio clock (AUDIO_MCK). To stop SSIF-2 completely, make sure that SSIF-2 is in the idle state (SSISR.IIRQ = 1) and then write 0 to SSIFCR.ADCKE. If 0 is written to SSIFCR.ADCKE before SSIF-2 becomes idle, take the procedure to start communication (Figure 24.52).

(2) LRCONT Control

To stop the output to the SSILRCK/SSIFS pin with SSIOFR.LRCONT when SSIF-2 is in the idle state in master-mode communication (SSICR.MST = 1), note the following: The output stops when 0 is written from 1 to SSIOFR.LRCONT. (Figure 24.43). Make sure that the remote device is not affected.

(3) BCKASTP Control

To stop the output to the SSIBCK pin with SSIOFR.BCKASTP in master-mode communication (SSICR.MST = 1) and while SSIF-2 is in the idle state, note the following; By writing 0 to SSIOFR.BCKASTP while it is 1, the output stops immediately (Figure 24.44). So, make sure that the remote device is not affected.
Note: When the opposing device who is the slave needs the clock of the terminal SSIBCK before the communication operates, it is not possible to use it.

24.6.1.4 Notes for Communication Flow

(1) When an Error Interrupt is Generated

SSIF-2 has the following four errors.

- Transmit underflow error
- Transmit overflow error
- Receive underflow error
- Receive overflow error

When an underflow error or overflow error is generated, SSIF-2 need to be restarted. Take the error procedure (Figure 24.56) according to the procedure to halt communication (Figure 24.55).

(a) Transmit Underflow Error

In the case a transmit underflow error is generated during transmission (SSICR.TEN = 1, SSISR.IIRQ = 0), review the number of write operations to the transmit FIFO data register (SSIFTDR) according to serial data empty interrupts. After generation of a transmit underflow error, data to be transmitted from SSIF-2 are 0 output. To transmit data written to SSIFTxD pin normally, run the procedure to halt communication (Figure 24.55) and then take the error procedure (Figure 24.56). When this error is generated, serial data is consumed as usual. To resume transmission of SSIF-2, write data from the beginning of one frame.

(b) Transmit Overflow Error

In the case a transmit overflow error is generated, review the number of write operations to the transmit FIFO data register (SSIFTDR) according to transmit data empty interrupts. Data is not written to SSIFTDR where a transmit overflow error is generated. This error is generated regardless of while the transmission is operating. To resume transmission of SSIF-2, take the procedure to halt communication (Figure 24.55) and then take the error procedure (Figure 24.56). When communication is resumed, consider the lost serial data.

(c) Receive Underflow Error

In the case a receive underflow error is generated, review the number of reading operations from the receive FIFO data register (SSIFRDR) according to receive data full interrupts. The value read from SSIFRDR is undefined where a receive underflow error is generated. This error is generated regardless of while the reception is operating. To resume reception

of SSIF-2, take the procedure to halt communication (Figure 24.55) and then take the error procedure (Figure 24.56).

(d) Receive Overflow Error

In the case a receive overflow error is generated during reception (SSICR.REN = 1, SSISR.IIRQ = 0), review the number of reading operations from the receive FIFO data register (SSIFRDR) according to receive data full interrupts. Data is not stored in SSIFRDR when a receive overflow error is generated. To resume reception of SSIF-2, take the procedure to halt communication (Figure 24.55) and then take the error handling procedure (Figure 24.56).

(2) Transmit Data Empty Interrupt

The communication flow defined in SSIF-2 uses the DMA. In the case the DMA is not used, observe one of SSIFSR.TDE, and write it in SSIFTDR. Follow empty capacity of transmission FIFO data register set with SSISCR.TDES about the number of times written in SSIFTDR detecting 1 of SSIFSR.TDE. Clear the transmission data that corresponds to empty capacity and clear the SSIFSR.TDE flag to SSIFTDR after writing. Transmission operation consecutive by the repeated thing is possible. SSIFSR.TDE is not cleared when not clearing.

(3) Receive Data Full Interrupt

The communication flow defined in SSIF-2 uses the DMA. In the case the DMA is not used, observe 1 of SSIFSR.RDF, and read SSIFRDR. Follow the receive data capacity of reception FIFO data register set with SSISCR.RDFS about the number of times in which SSIFRDR is read detecting 1 of SSIFSR.RDF. Clear data that does the reception completion and clear the SSIFSR.RDF flag from SSIFRDR after reading. Reception operation consecutive by the repeated thing is possible. SSIFSR.RDF is not cleared when not clearing.

(4) Switching Transfer Modes

1. For state transition from transmission, reception, and transmission and reception, disable transmission and reception (SSICR.TEN = 0, SSICR.REN = 0).
2. Confirm it is in the idle state (SSISR.IIRQ = 1).
3. In the idle state, set the SSICR.TEN bit and the SSICR.REN bit again and resume transfer.

(5) Resume Communication after Halting SSIF-2

When communication of SSIF-2 is halted according to the procedure to halt communication in Figure 24.55, resume communication according to the procedure to resume communication in Figure 24.52.

24.6.1.5 Write Access Restriction

(1) SSICR Register

If the TEN bit or REN bit is rewritten, make sure that the SSISR.IIRQ bit is in the desired status. If written, the operation performed immediately after writing is not guaranteed. For example, after enabling operation, make sure that SSISR.IIRQ = 0 and after disabling operation, make sure that SSISR.IIRQ = 1.

(a) TEN Bit and REN Bit

Writing 1 to these bits, transmission starts in synchronization with start trigger. Refer to section 24.5.2.2, Transmission, section 24.5.2.3, Reception, and section 24.5.2.4, Transmission and Reception for details. Writing 0 to these bits, transmission stops in synchronization with next frame boundary. Set 1 at the same time when you use SSIF-2 as transmission and reception. To stop communication, make sure to stop transmission and reception (TEN = 0 and REN = 0).

(2) SSICR Register

(a) Clear operation of TUIRQ and TOIRQ

After the communication is permitted (SSICR.TEN is written and '1' is written from '0'), the transmission error flag (TOIRQ and TUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the transmission error flag might not be able to be read.

(b) Clear operation of RUIRQ and ROIRQ

After the communication is permitted (SSICR.REN is written and '1' is written from '0'), the transmission error flag (ROIRQ and RUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the reception error flag might not be able to be read.

(3) Communication State

Writing to the bits with shaded area in Table 24.20 is prohibited. If written, the operation performed immediately after writing is not guaranteed.

Table 24.20 Bits Protected from Writing during Communication

Symbol	Address (Base+)		+0 31	+0 30	+0 29	+0 28	+0 27	+0 26	+0 25	+0 24	+1 23	+1 22	+1 21	+1 20	+1 19	+1 18	+1 17	+1 16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	00h	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	04h	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	10h	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	BCKNCE	LRCKNCE	RxDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	14h	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	18h	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	1ch	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR		+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	24h	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	TDES[5:0]						—	—	RDFS[5:0]					

25. CANFD Interface (RS-CANFD)

This section contains a generic description of the CANFD interface (RS-CANFD).
The first part of this section describes all this LSI specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of RS-CANFD.

25.1 Features of RS-CANFD

25.1.1 Number of Units and Channels

This product has the following number of RS-CANFD units.

Table 25.1 Number of Units

RS-CANFD	
Number of Units	1
Name	RSCFD0

This product has the CANFD interface channel listed below.

Table 25.2 Unit Configurations and Channels

RS-CANFD	
Number of Channels	2
Name	CAN0, CAN1

The RS-CANFD has two interface modes (classical CAN mode and CANFD mode) and uses different registers for each mode. There are two types of register names RSCANnXXX and RSCFDnCFDXXX (XXX: arbitrary) depending on interface modes. When explaining specifications common to two registers, register names are described as RSCFDn(CFD)XXX.

Table 25.3 Index

Index	Meaning
n	Throughout this section, the individual RS-CANFD units are generically indicated by the index “n” (n = 0); for example, RSCFDn(CFD)GCTR is the global control register of the RSCFDn unit.
m	Throughout this section, the individual channels of RS-CANFD units are generically indicated by the index “m” (m = 0, 1); for example, RSCFDn(CFD)CmSTS is the channel m status register.
j	The individual registers associated with receive rule table are generically indicated by the index “j” (j = 0 to 15); for example, RSCFDn(CFD)GAFLIDj is the receive rule ID register.
k	The individual transmit/receive FIFO buffers are generically indicated by the index “k” (k = 0 to 5); for example, RSCFDn(CFD)CFCCk is the transmit/receive FIFO buffer configuration/control register.
x	The individual receive FIFO buffers are generically identified by the index “x” (x = 0 to 7); for example, RSCFDn(CFD)RFSTSx is the receive FIFO buffer status register.
d	Data field registers of transmit/receive FIFO buffers and receive FIFO buffers are identified by “d” (classical CAN mode: d = 0, 1, CANFD mode: d = 0 to 15). For example, the transmit/receive FIFO buffer data field register is described as RSCFDn(CFD)CFDFd_k.
q	The individual receive buffers are generically indicated by the index “q” (q = 0 to 31); for example, RSCFDn(CFD)RMIDq is the receive buffer ID register.
p	The individual transmit buffers are generically indicated by the index “p” (p = 0 to 31); for example, RSCFDn(CFD)TMCp is the transmit buffer control register.
b	Data field registers of receive buffers and transmit buffers are identified by “b” (classical CAN mode: b = 0, 1, CANFD mode: b = 0 to 4). For example, the receive buffer data field register is described as RSCFDn(CFD)RMDFb_q.
r	The individual RAM tests for CAN are generically indicated by the index “r” (r = 0 to 63); for example, RSCFDn(CFD)RPGACCr is the RAM test page access register.
y	The registers not covered above are indicated by the letter “y” (y = 0); for example, RSCFDn(CFD)RMNDy is a receive buffer new data register.

Note: The functions and descriptions of registers in this section are for the RS-CANFDs that has 2 channels (m = 0, 1). When referring to information with indexes, regard the index values as the ones corresponding to your target product. Also, note that, if the value of an index exceeds the range described in this section due to your target product, write the value after reset when writing to bits outside the index range.

25.1.2 Register Base Address

RSCFDn base addresses are listed in the following table.
RSCFDn register addresses are given as offsets from the base addresses in general.

Table 25.4 Register Base Address

Base Address Name	Base Address
<RSCFD0_base>	E802 0000

25.1.3 Clock Supply

The RSCFDn clock supply is shown in the following table.

Table 25.5 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
RSCFDn	clk_xincan	CAN_CLK
	clkc	P1φ/2
	pclk	P1φ

The operating frequency of the RSCFDn depends on the transfer rate and the number of channels in use. Table 25.6 shows the range of the frequency.

Table 25.6 Range of Operating Frequency Depending on the Transfer Rate and the Number of Channels in Use in this LSI

Condition			Range of Operating Frequency		
Mode	Transfer Rate	No. of Channels in Use	pclk	clk_xincan*1	clkc*1,*2
CANFD	4.125 Mbps	2ch	pclk ≥ 64 MHz	32 MHz ≤ clk_xincan ≤ pclk/2	32 MHz ≤ clkc ≤ pclk/2
		1ch			
	2 Mbps	2ch	pclk ≥ 32 MHz	16 MHz ≤ clk_xincan ≤ pclk/2	16 MHz ≤ clkc ≤ pclk/2
		1ch			
CAN/CANFD	1 Mbps	2ch	pclk ≥ 30.6 MHz	8 MHz ≤ clk_xincan ≤ pclk/2	8 MHz ≤ clkc ≤ pclk/2
		1ch	pclk ≥ 27.5 MHz		
	500 kbps	2ch	pclk ≥ 27.5 MHz	4 MHz ≤ clk_xincan ≤ pclk/2	4 MHz ≤ clkc ≤ pclk/2
		1ch			
	125 kbps	2ch	pclk ≥ 27.5 MHz	1 MHz ≤ clk_xincan ≤ pclk/2	1 MHz ≤ clkc ≤ pclk/2
		1ch			

Note 1. Setting the DCS bit in the RSCFDn(CFD)GCFG register enables to select either clk_xincan or clkc. Set clocks less than or equal to pclk/2.
Note 2. When pclk< 25 MHz, select clk_xincan.

25.1.4 Interrupt Request

RSCFDn interrupt requests are listed in the following table.

Table 25.7 Interrupt Requests

Unit Interrupt Signal		Outline
INTRCANGERR		CAN global error interrupt
INTRCANGRECC		CAN receive FIFO interrupt
CAN0	INTRCAN0ERR	CAN0 error interrupt
	INTRCAN0REC	CAN0 transmit/receive FIFO receive completion interrupt
	INTRCAN0TRX	CAN0 transmit interrupt
CAN1	INTRCAN1ERR	CAN1 error interrupt
	INTRCAN1REC	CAN1 transmit/receive FIFO receive completion interrupt
	INTRCAN1TRX	CAN1 transmit interrupt

25.1.5 External Input/Output Signals

External input/output signals of RSCFDn are listed below.

Table 25.8 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal
CANmRX (m = 0, 1)	CANm receive data input	CANmRX (m = 0, 1)
CANmTX (m = 0, 1)	CANm transmit data output	CANmTX (m = 0, 1)
CANmRX_DATARATE_EN (m = 0, 1)	CANm receive data phase output	CANmRX_DATARATE_EN (m = 0, 1)
CANmTX_DATARATE_EN (m = 0, 1)	CANm transmit data phase output	CANmTX_DATARATE_EN (m = 0, 1)
CAN_CLK	Clock source for CAN communication	CAN_CLK

25.2 Overview

25.2.1 Functional Overview

Table 25.9 shows the RS-CANFD module specifications. Figure 25.1 shows the RS-CANFD module block diagram.

Table 25.9 RS-CANFD Module Specifications

Item	Specification
Number of channels	2
Protocol	ISO11898-1 compliant Using CANFD frames is selectable by switching interface modes.
Communication speed	Classical CAN mode: • Maximum 1 Mbps $\text{Communication speed (CANm bit time clock)} = \frac{1}{\text{CANm bit time}}$ CANm bit time = CANmTq × Tq count per bit $\text{CANmTq} = \frac{\text{(BRP[9:0] bits in the RSCANnCmCFG register + 1)}}{\text{fCAN}}$ fCAN: Frequency of CAN clock (selected by the DCS bit in the RSCANnGCFG register) CANFD mode: • Nominal bit rate: max.1 Mbps, data bit rate: max. 4.125 Mbps $\text{Transmission rate (CANm nominal bit time clock)} = \frac{1}{\text{CANm nominal bit time}}$ $\text{Transmission rate (CANm data bit time clock)} = \frac{1}{\text{CANm data bit time}}$ CANm nominal bit time = CANmTq(N) × Tq count per nominal bit CANm data bit time = CANmTq(D) × Tq count per data bit $\text{CANmTq(N)} = \frac{\text{(NBRP[9:0] bits in the RSCFDnCFDCmNCFG register + 1)}}{\text{fCAN}}$ $\text{CANmTq(D)} = \frac{\text{(DBRP[7:0] bits in the RSCFDnCFDCmDCFG register + 1)}}{\text{fCAN}}$ fCAN: Frequency of CAN clock (selected by the DCS bit in the RSCFDnCFDGCFG register) m = 0, 1 Tq: Time quantum
Buffer	160 buffers in total • Individual buffers: 32 buffers (16 buffers × 2 channels) Transmit buffer: 16 buffers per channel Transmit queue: Single queue per channel (shared with the transmit buffer; up to 16 buffers allocatable) • Shared buffers: 128 buffers for all channels Receive buffer: 0 to 32 buffers Receive FIFO buffer: 8 FIFO buffers (up to 128 buffers allocatable to each) Transmit/receive FIFO buffer: 3 FIFO buffers per channel (up to 128 buffers allocatable to each)
Reception function	• Receives data frames and remote frames. • Selects ID format (standard ID, extended ID, or both IDs) to be received. • Sets interrupt enable/disable for each FIFO. • Mirror function (reception of messages transmitted from the own CAN node) • Timestamp function (to record message reception time as a 16-bit timer value)
Reception filter function	• Selects receive messages according to 128 receive rules. • Sets the number of receive rules (0 to 128) for each channel. • Acceptance filter processing: Sets ID and mask for each receive rule. • DLC filter processing: Enables DLC filter check for each acceptance rule.

Item	Specification
Receive message transfer function	• Routing function Transfers receive messages to arbitrary destinations (can be transferred to up to 8 buffers) Transfer destination: Receive buffer, receive FIFO buffer, and/or transmit/receive FIFO buffer • Label addition function Stores label information together with a message in a receive buffer and FIFO buffer.
Transmission function	• Transmits data frames and remote frames. • Selects ID format (standard ID, extended ID, or both IDs) to be transmitted. • Sets interrupt enable/disable for each transmit buffer and transmit/receive FIFO buffer. • Selects ID priority transmission or transmit buffer number priority transmission. • Transmit request can be aborted (possible to confirm with a flag) • One-shot transmission function
Interval transmission function	Transmit messages at configurable intervals (transmit mode or gateway mode of transmit/receive FIFO buffers)
Transmit queue function	Transmits all stored messages according to the ID priority.
Transmit history function	Stores the history information of transmission-completed messages Adds timestamp (recording message transmission time as a 16-bit timer value) to the history information.
Gateway function	Transmits a received message automatically.
Bus off recovery mode selection	Selects the method for returning from bus off state. • ISO11898-1 compliant • Automatic entry to channel halt mode at bus-off entry • Automatic entry to channel halt mode at bus-off end • Transition to channel halt mode by program request • Transition to the error-active state by program request (forcible return from the bus off state)
Error status monitoring	• Monitors CAN protocol errors (stuff error, form error, ACK error, CRC error, bit error, ACK delimiter error, and bus dominant lock). • Detects error status transitions (error warning, error passive, bus off entry, and bus off recovery) • Reads the error counter. • Monitors DLC errors.
Interrupt source	8 sources • Global Interrupts (2 sources) Receive FIFO interrupt Global error interrupt • Channel interrupts (3 sources/channel) CANm transmit interrupt (m = 0, 1) - CANm transmit complete interrupt - CANm transmit abort interrupt - CANm transmit/receive FIFO transmit complete interrupt (in transmit mode, gateway mode) - CANm transmit history interrupt - CANm transmit queue interrupt CANm transmit/receive FIFO receive complete interrupt (in receive mode, gateway mode) CANm error interrupt
CAN stop mode	Reduces power consumption by stopping clock supply to the RS-CANFD module.
CAN clock source	Selects the clkc or the clk_xincan. As for the range of operating frequency, see Table 25.6
Test function	Test function for user evaluation • Listen-only mode • Self-test mode 0 (external loopback) • Self-test mode 1 (internal loopback) • Restricted operation mode • RAM test (read/write test) • Inter-channel communication test [CRC error test enabled]

25.2.2 Interface Modes

The RS-CANFD has two interface modes.

- Classical CAN mode: Handles only classical CAN frames.
- CANFD mode: Handles classical CAN frames and CANFD frames.

These two modes use different register maps with the same base address. Register maps change by switching interface modes.

Interface modes are switched by the RCMC bit in the RSCFDn(CFD)GRMCFG register.

25.2.3 Block Diagram

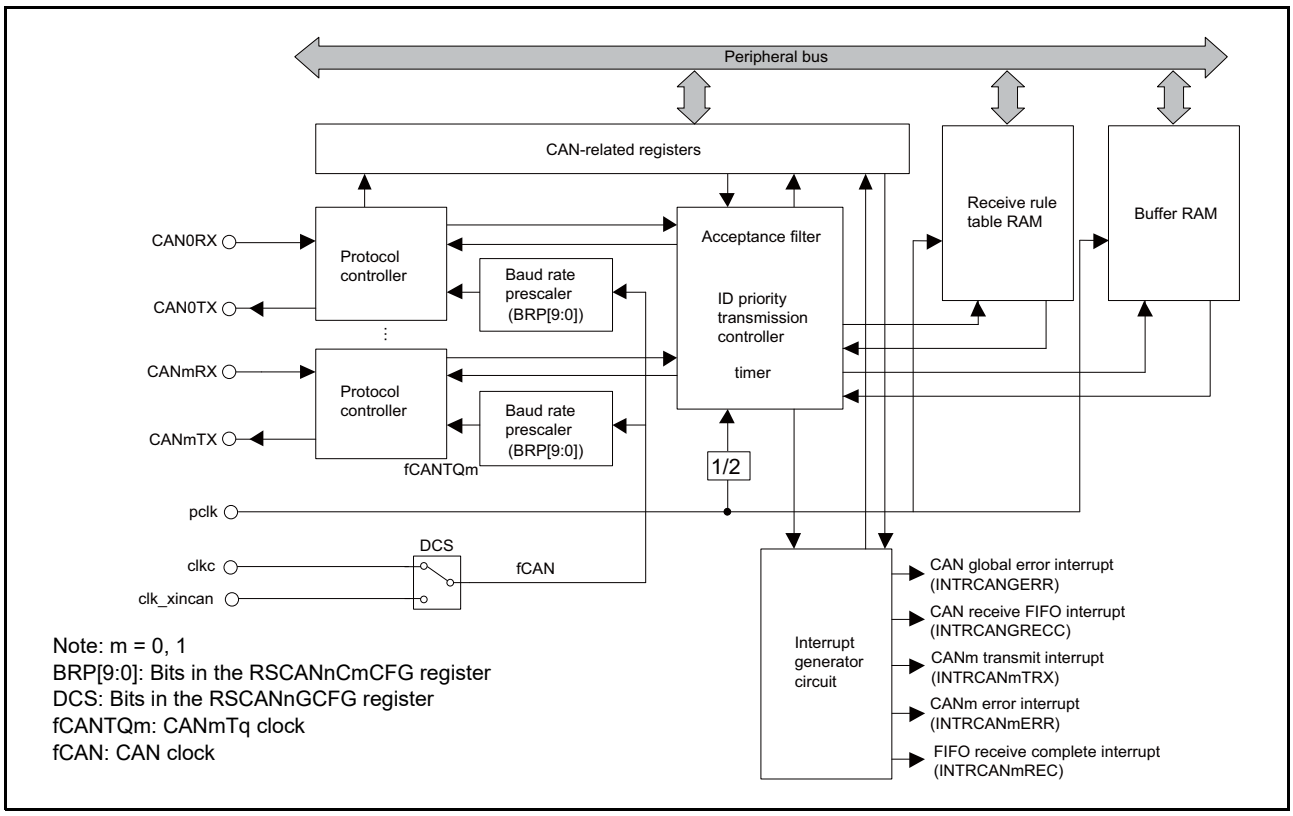


Figure 25.1 RS-CANFD Module Block Diagram (Classical CAN Mode)

In CANFD mode, different clock signals are input to the baud rate prescaler and the protocol controller respectively. See section 25.11.1.3, Communication Speed Setting.

25.3 Registers (Classical CAN Mode)

25.3.1 List of Registers

The following tables list RS-CANFD registers to be used in classical CAN mode.
For details about <RSCFDn_base>, see section 25.1.2, Register Base Address.
For registers initialized in global reset mode and channel reset mode, refer to the following.

- Table 25.176, Registers Initialized in Global Reset Mode or Channel Reset Mode
- Table 25.177, Registers Initialized Only in Global Reset Mode

Table 25.10 Registers

Module	Register	Symbol	Address
Interface mode-related registers			
RSCANn	Global interface mode select register	RSCANnGRMCFG	<RSCFDn_base> + $04FC_H$
Channel-related registers			
RSCANn	Channel m Configuration Register	RSCANnCmCFG	<RSCFDn_base> + 0000_H + (10_H × m)
RSCANn	Channel m control register	RSCANnCmCTR	<RSCFDn_base> + 0004_H + (10_H × m)
RSCANn	Channel m status register	RSCANnCmSTS	<RSCFDn_base> + 0008_H + (10_H × m)
RSCANn	Channel m error flag register	RSCANnCmERFL	<RSCFDn_base> + $000C_H$ + (10_H × m)
Global-related registers			
RSCANn	Global configuration register	RSCANnGCFG	<RSCFDn_base> + 0084_H
RSCANn	Global control register	RSCANnGCTR	<RSCFDn_base> + 0088_H
RSCANn	Global status register	RSCANnGSTS	<RSCFDn_base> + $008C_H$
RSCANn	Global error flag register	RSCANnGERFL	<RSCFDn_base> + 0090_H
RSCANn	Global timestamp counter register	RSCANnGTSC	<RSCFDn_base> + 0094_H
RSCANn	Global TX Interrupt Status Register 0	RSCANnGTINTSTS0	<RSCFDn_base> + 0460_H
RSCANn	Global FD Configuration Register	RSCANnGFDCFG	<RSCFDn_base> + 0474_H
Receive rule-related registers			
RSCANn	Receive Rule Entry Control Register	RSCANnGAFLECTR	<RSCFDn_base> + 0098_H
RSCANn	Receive Rule Configuration Register 0	RSCANnGAFLCFG0	<RSCFDn_base> + $009C_H$
RSCANn	Receive Rule ID Register j	RSCANnGAFLIDj	<RSCFDn_base> + 0500_H + (10_H × j)
RSCANn	Receive Rule Mask Register j	RSCANnGAFLMj	<RSCFDn_base> + 0504_H + (10_H × j)
RSCANn	Receive Rule Pointer 0 Register j	RSCANnGAFLP0_j	<RSCFDn_base> + 0508_H + (10_H × j)
RSCANn	Receive Rule Pointer 1 Register j	RSCANnGAFLP1_j	<RSCFDn_base> + $050C_H$ + (10_H × j)
Receive buffer-related registers			
RSCANn	Receive Buffer Number Register	RSCANnRMNB	<RSCFDn_base> + $00A4_H$
RSCANn	Receive Buffer New Data Register y	RSCANnRMNDy	<RSCFDn_base> + $00A8_H$ + (04_H × y)
RSCANn	Receive Buffer ID Register q	RSCANnRMIDq	<RSCFDn_base> + 0600_H + (10_H × q)
RSCANn	Receive Buffer Pointer Register q	RSCANnRMPTRq	<RSCFDn_base> + 0604_H + (10_H × q)
RSCANn	Receive Buffer Data Field 0 Register q	RSCANnRMDF0_q	<RSCFDn_base> + 0608_H + (10_H × q)
RSCANn	Receive Buffer Data Field 1 Register q	RSCANnRMDF1_q	<RSCFDn_base> + $060C_H$ + (10_H × q)
Receive FIFO buffer-related registers			
RSCANn	Receive FIFO Buffer Configuration and Control Register x	RSCANnRFCCx	<RSCFDn_base> + $00B8_H$ + (04_H × x)
RSCANn	Receive FIFO Buffer Status Register x	RSCANnRFSTSx	<RSCFDn_base> + $00D8_H$ + (04_H × x)
RSCANn	Receive FIFO Buffer Pointer Control Register x	RSCANnRFPCTRx	<RSCFDn_base> + $00F8_H$ + (04_H × x)
RSCANn	Receive FIFO Buffer Access ID Register x	RSCANnRFIDx	<RSCFDn_base> + $0E00_H$ + (10_H × x)
RSCANn	Receive FIFO Buffer Access Pointer Register x	RSCANnRFPTRx	<RSCFDn_base> + $0E04_H$ + (10_H × x)

Module	Register	Symbol	Address
RSCANn	Receive FIFO Buffer Access Data Field 0 Register x	RSCANnRFDF0_x	<RSCFDn_base> + $0E08_H$ + (10_H × x)
RSCANn	Receive FIFO Buffer Access Data Field 1 Register x	RSCANnRFDF1_x	<RSCFDn_base> + $0E0C_H$ + (10_H × x)
Transmit/Receive FIFO buffer related registers			
RSCANn	Transmit/receive FIFO Buffer Configuration and Control Register k	RSCANnCFCCk	<RSCFDn_base> + 0118_H + (04_H × k)
RSCANn	Transmit/receive FIFO Buffer Status Register k	RSCANnCFSTSk	<RSCFDn_base> + 0178_H + (04_H × k)
RSCANn	Transmit/receive FIFO Buffer Pointer Control Register k	RSCANnCFPCTRk	<RSCFDn_base> + $01D8_H$ + (04_H × k)
RSCANn	Transmit/receive FIFO Buffer Access ID Register k	RSCANnCFIDk	<RSCFDn_base> + $0E80_H$ + (10_H × k)
RSCANn	Transmit/receive FIFO Buffer Access Pointer Register k	RSCANnCFPTRk	<RSCFDn_base> + $0E84_H$ + (10_H × k)
RSCANn	Transmit/receive FIFO Buffer Access Data Field 0 Register k	RSCANnCFDF0_k	<RSCFDn_base> + $0E88_H$ + (10_H × k)
RSCANn	Transmit/receive FIFO Buffer Access Data Field 1 Register k	RSCANnCFDF1_k	<RSCFDn_base> + $0E8C_H$ + (10_H × k)
FIFO status-related registers			
RSCANn	FIFO Empty Status Register	RSCANnFESTS	<RSCFDn_base> + 0238_H
RSCANn	FIFO Full Status Register	RSCANnFFSTS	<RSCFDn_base> + $023C_H$
RSCANn	FIFO Message Lost Status Register	RSCANnFMSTS	<RSCFDn_base> + 0240_H
RSCANn	RSCANn FIFO Receive FIFO Buffer Interrupt Flag Status Register	RSCANnRFISTS	<RSCFDn_base> + 0244_H
RSCANn	Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register	RSCANnCFRISTS	<RSCFDn_base> + 0248_H
RSCANn	Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register	RSCANnCFTISTS	<RSCFDn_base> + $024C_H$
Transmit buffer-related registers			
RSCANn	Transmit Buffer Control Register p	RSCANnTMCp	<RSCFDn_base> + 0250_H + (01_H × p)
RSCANn	Transmit Buffer Status Register p	RSCANnTMSTSp	<RSCFDn_base> + $02D0_H$ + (01_H × p)
RSCANn	Transmit Buffer ID Register p	RSCANnTMIDp	<RSCFDn_base> + 1000_H + (10_H × p)
RSCANn	Transmit Buffer Pointer Register p	RSCANnTMPTRp	<RSCFDn_base> + 1004_H + (10_H × p)
RSCANn	Transmit Buffer Data Field 0 Register p	RSCANnTMDF0_p	<RSCFDn_base> + 1008_H + (10_H × p)
RSCANn	Transmit Buffer Data Field 1 Register p	RSCANnTMDF1_p	<RSCFDn_base> + $100C_H$ + (10_H × p)
RSCANn	Transmit Buffer Interrupt Enable Configuration Register y	RSCANnTMIECy	<RSCFDn_base> + 0390_H + (04_H × y)
Transmit buffer status-related registers			
RSCANn	Transmit Buffer Transmit Request Status Register y	RSCANnTMTRSTSy	<RSCFDn_base> + 0350_H + (04_H × y)
RSCANn	Transmit Buffer Transmit Abort Request Status Register y	RSCANnTMTARSTSy	<RSCFDn_base> + 0360_H + (04_H × y)
RSCANn	Transmit Buffer Transmit Complete Status Register y	RSCANnTMTCSTSy	<RSCFDn_base> + 0370_H + (04_H × y)
RSCANn	Transmit Buffer Transmit Abort Status Register y	RSCANnTMTASTSy	<RSCFDn_base> + 0380_H + (04_H × y)
Transmit queue-related registers			
RSCANn	Transmit Queue Configuration and Control Register m	RSCANnTXQCCm	<RSCFDn_base> + $03A0_H$ + (04_H × m)
RSCANn	Transmit Queue Status Register m	RSCANnTXQSTSm	<RSCFDn_base> + $03C0_H$ + (04_H × m)
RSCANn	Transmit Queue Pointer Control Register m	RSCANnTXQPCTRm	<RSCFDn_base> + $03E0_H$ + (04_H × m)
Transmit history-related registers			
RSCANn	Transmit History Configuration and Control Register m	RSCANnTHLCCm	<RSCFDn_base> + 0400_H + (04_H × m)
RSCANn	Transmit History Status Register m	RSCANnTHLSTSm	<RSCFDn_base> + 0420_H + (04_H × m)
RSCANn	Transmit History Pointer Control Register m	RSCANnTHLPCTRm	<RSCFDn_base> + 0440_H + (04_H × m)
RSCANn	Transmit History Access Register m	RSCANnTHLACCm	<RSCFDn_base> + 1800_H + (04_H × m)

Module	Register	Symbol	Address
Test-related registers			
RSCANn	Global Test Configuration Register	RSCANnGTSTCFG	<RSCFDn_base> + 0468_H
RSCANn	Global Test Control Register	RSCANnGTSTCTR	<RSCFDn_base> + $046C_H$
RSCANn	Global Lock Key Register	RSCANnGLOCKK	<RSCFDn_base> + $047C_H$
RSCANn	RAM Test Page Access Register r	RSCANnRPGACCr	<RSCFDn_base> + 1900_H + (04_H × r)

Table 25.11 Transmit Buffer p Allocated to Each Channel

	CANm
Transmit buffer p	Transmit buffer 16 × m + 0
	Transmit buffer 16 × m + 1
	Transmit buffer 16 × m + 2
	Transmit buffer 16 × m + 3
	Transmit buffer 16 × m + 4
	Transmit buffer 16 × m + 5
	Transmit buffer 16 × m + 6
	Transmit buffer 16 × m + 7
	Transmit buffer 16 × m + 8
	Transmit buffer 16 × m + 9
	Transmit buffer 16 × m + 10
	Transmit buffer 16 × m + 11
	Transmit buffer 16 × m + 12
	Transmit buffer 16 × m + 13
	Transmit buffer 16 × m + 14
	Transmit buffer 16 × m + 15

Table 25.12 Transmit/Receive FIFO Buffer k Allocated to Each Channel

	CANm
Transmit/receive FIFO buffer k	Transmit/receive FIFO buffer 3 × m + 0
	Transmit/receive FIFO buffer 3 × m + 1
	Transmit/receive FIFO buffer 3 × m + 2

Table 25.13 Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer by the Setting of Bits CFTML[3:0]

Setting of Bits CFTML[3:0]	Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer
0000_B	Transmit buffer 16 × m + 0
0001_B	Transmit buffer 16 × m + 1
0010_B	Transmit buffer 16 × m + 2
0011_B	Transmit buffer 16 × m + 3
0100_B	Transmit buffer 16 × m + 4
0101_B	Transmit buffer 16 × m + 5
0110_B	Transmit buffer 16 × m + 6
0111_B	Transmit buffer 16 × m + 7
1000_B	Transmit buffer 16 × m + 8
1001_B	Transmit buffer 16 × m + 9
1010_B	Transmit buffer 16 × m + 10
1011_B	Transmit buffer 16 × m + 11
1100_B	Transmit buffer 16 × m + 12

Setting of Bits CFTML[3:0]	Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer
1101_B	Transmit buffer 16 × m + 13
1110_B	Transmit buffer 16 × m + 14
1111_B	Transmit buffer 16 × m + 15

Table 25.14 Transmit Buffer p Allocated to the Transmit Queue of Each Channel

Setting of Bits TXQDC[3:0]	Transmit Buffer p Allocated to the Transmit Queue
0000_B	Setting prohibited
0001_B	Setting prohibited
0010_B	Transmit buffer 16 × m + 15 to 16 × m + 13
0011_B	Transmit buffer 16 × m + 15 to 16 × m + 12
0100_B	Transmit buffer 16 × m + 15 to 16 × m + 11
0101_B	Transmit buffer 16 × m + 15 to 16 × m + 10
0110_B	Transmit buffer 16 × m + 15 to 16 × m + 9
0111_B	Transmit buffer 16 × m + 15 to 16 × m + 8
1000_B	Transmit buffer 16 × m + 15 to 16 × m + 7
1001_B	Transmit buffer 16 × m + 15 to 16 × m + 6
1010_B	Transmit buffer 16 × m + 15 to 16 × m + 5
1011_B	Transmit buffer 16 × m + 15 to 16 × m + 4
1100_B	Transmit buffer 16 × m + 15 to 16 × m + 3
1101_B	Transmit buffer 16 × m + 15 to 16 × m + 2
1110_B	Transmit buffer 16 × m + 15 to 16 × m + 1
1111_B	Transmit buffer 16 × m + 15 to 16 × m + 0

25.3.2 Details of Interface Mode-related Registers

25.3.2.1 RSCANnGRMCFG — Global Interface Mode Select Register

Access: RSCANnGRMCFG register can be read/written in 32-bit units
RSCANnGRMCFGL, RSCANnGRMCFGHregisters can be read/written in 16-bit units
RSCANnGRMCFGLL, RSCANnGRMCFGLH, RSCANnGRMCFGHL, RSCANnGRMCFGHHregisters can be read/written in 8-bit units

Address: RSCANnGRMCFG: <RSCFDn_base> + $04FC_H$,
RSCANnGRMCFGL: <RSCFDn_base> + $04FC_H$,
RSCANnGRMCFGH: <RSCFDn_base> + $04FE_H$,
RSCANnGRMCFGLL: <RSCFDn_base> + $04FC_H$,
RSCANnGRMCFGLH: <RSCFDn_base> + $04FD_H$,
RSCANnGRMCFGHL: <RSCFDn_base> + $04FE_H$,
RSCANnGRMCFGHH: <RSCFDn_base> + $04FF_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RCMC
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 25.15 RSCANnGRMCFG Register Contents

Bit	Bit Name	Function
31 to 1	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
0	RCMC	Interface Mode Select 0: Classical CAN mode 1: CANFD mode

Note 1. The RSCANnGRMCFG register and the RSCFDnCFDGRMCFG register are the same register. Therefore, set either one of the registers.

Modify the RSCANnGRMCFG register only in global reset mode. Before setting other RS-CANFD registers, set this register.

RCMC Bit

Setting this bit to 0 places the RS-CANFD module in classical CAN mode.
Setting this bit to 1 places the RS-CANFD module in CANFD mode. To switch CANFD mode to classical CAN mode, set the value after reset in all registers and bits allocated to the register map of CANFD mode and then modify the RSCANnGRMCFG register.

25.3.3 Details of Channel-related Registers

25.3.3.1 RSCANnCmCFG — Channel Configuration Register (m = 0, 1)

Access: RSCANnCmCFG register can be read/written in 32-bit units
RSCANnCmCFGL, RSCANnCmCFGH registers can be read/written in 16-bit units
RSCANnCmCFGLL, RSCANnCmCFGLH, RSCANnCmCFGHL, RSCANnCmCFGHH registers can be read/written in 8-bit unitsa

Address: RSCANnCmCFG: <RSCFDn_base> + 0000_H + (10_H × m)
RSCANnCmCFGL: <RSCFDn_base> + 0000_H + (10_H × m),
RSCANnCmCFGH: <RSCFDn_base> + 0002_H + (10_H × m)
RSCANnCmCFGLL: <RSCFDn_base> + 0000_H + (10_H × m),
RSCANnCmCFGLH: <RSCFDn_base> + 0001_H + (10_H × m),
RSCANnCmCFGHL: <RSCFDn_base> + 0002_H + (10_H × m),
RSCANnCmCFGHH: <RSCFDn_base> + 0003_H + (10_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	SJW[1:0]		—	TSEG2[2:0]			TSEG1[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	BRP[9:0]									
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.16 RSCANnGRMCFG Register Contents

Bit	Bit Name	Function
31 to 26	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
25, 24	SJW[1:0]	Resynchronization Jump Width Control b25 b24 0 0: 1 Tq 0 1: 2 Tq 1 0: 3 Tq 1 1: 4 Tq
23	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
22 to 20	TSEG2[2:0]	Time Segment 2 Control b22 b21 b20 0 0 0: Setting prohibited 0 0 1: 2 Tq 0 1 0: 3 Tq 0 1 1: 4 Tq 1 0 0: 5 Tq 1 0 1: 6 Tq 1 1 0: 7 Tq 1 1 1: 8 Tq

Bit	Bit Name	Function
19 to 16	TSEG1[3:0]	Time Segment 1 Control b19 b18 b17 b16 0 0 0 0: Setting prohibited 0 0 0 1: Setting prohibited 0 0 1 0: Setting prohibited 0 0 1 1: 4 Tq 0 1 0 0: 5 Tq 0 1 0 1: 6 Tq 0 1 1 0: 7 Tq 0 1 1 1: 8 Tq 1 0 0 0: 9 Tq 1 0 0 1: 10 Tq 1 0 1 0: 11 Tq 1 0 1 1: 12 Tq 1 1 0 0: 13 Tq 1 1 0 1: 14 Tq 1 1 1 0: 15 Tq 1 1 1 1: 16 Tq
15 to 10	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
9 to 0	BRP[9:0]	Prescaler Division Ratio Set When these bits are set to P (0 to 1023), the baud rate prescaler divides fCAN by P + 1.

Modify the RSCANnCmCFG register in channel reset mode or channel halt mode. Set this register in channel reset mode before shifting to channel communication mode or channel halt mode. For a description of the bit timing parameters and settings, see section 25.11.1, Initial Settings.

SJW[1:0] Bits

These bits are used to specify a Tq value for the resynchronization jump width. Allowed values are 1 Tq to 4 Tq, inclusive.
Set a value less than or equal to the value of the TSEG2 bits.

TSEG2[2:0] Bits

These bits are used to specify a Tq value for the length of phase segment 2 (PHASE_SEG2).
Allowed values are 2 Tq to 8 Tq, inclusive.
Set a value smaller than the value of the TSEG1 bits.

TSEG1[3:0] Bits

These bits are used to specify a Tq value for the total length of the propagation segment (PROP_SEG) and phase segment 1 (PHASE_SEG1).
Allowed values are 4 Tq to 16 Tq, inclusive.

BRP[9:0] Bits

The CANmTq clock (fCANTQm) is calculated by dividing the CAN clock (fCAN) by the baud rate prescaler, ((BRP[9:0]) + 1). One clock cycle of the CANmTq clock is 1 Time Quantum (Tq).

25.3.3.2 RSCANnCmCTR — Channel Control Register (m = 0, 1)

Access: RSCANnCmCTR register can be read/written in 32-bit units
RSCANnCmCTRL, RSCANnCmCTRH registers can be read/written in 16-bit units
RSCANnCmCTRLL, RSCANnCmCTRLH, RSCANnCmCTRHL, RSCANnCmCTRHH registers can be read/written in 8-bit units

Address: RSCANnCmCTR: <RSCFDn_base> + 0004_H + (10_H × m)
RSCANnCmCTRL: <RSCFDn_base> + 0004_H + (10_H × m),
RSCANnCmCTRH: <RSCFDn_base> + 0006_H + (10_H × m)
RSCANnCmCTRLL: <RSCFDn_base> + 0004_H + (10_H × m),
RSCANnCmCTRLH: <RSCFDn_base> + 0005_H + (10_H × m),
RSCANnCmCTRHL: <RSCFDn_base> + 0006_H + (10_H × m),
RSCANnCmCTRHH: <RSCFDn_base> + 0007_H + (10_H × m)

Value after reset: $0000\ 0005_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CRCT	—	—	—	CTMS[1:0]		CTME	ERRD	BOM[1:0]		—	—	—	—	TAIE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE	—	—	—	—	RTBO	CSLPR	CHMDC[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 25.17 RSCANnCmCTR Register Contents

Bit	Bit Name	Function
31	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
30	CRCT	CRC Error Test Enable 0: The first bit of the reception ID field is not inverted. 1: The first bit of the reception ID field is inverted.
29 to 27	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
26, 25	CTMS[1:0]	Communication Test Mode Select b26 b25 0 0: Standard test mode 0 1: Listen-only mode 1 0: Self-test mode 0 (external loopback mode) 1 1: Self-test mode 1 (internal loopback mode)
24	CTME	Communication Test Mode Enable 0: Communication test mode is disabled. 1: Communication test mode is enabled.
23	ERRD	Error Display Mode Select 0: Error flags are displayed only for the first error information after bits 14 to 8 in the RSCANnCmERFL register are all cleared. 1: Error flags for all error information are displayed.
22, 21	BOM[1:0]	Bus Off Recovery Mode Select b22 b21 0 0: ISO11898-1 compliant 0 1: Entry to channel halt mode automatically at bus-off entry 1 0: Entry to channel halt mode automatically at bus-off end 1 1: Entry to channel halt mode (in bus-off state) by program request
20 to 17	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
16	TAIE	Transmit Abort Interrupt Enable 0: Transmit abort interrupt is disabled. 1: Transmit abort interrupt is enabled.
15	ALIE	Arbitration Lost Interrupt Enable 0: Arbitration lost interrupt is disabled. 1: Arbitration lost interrupt is enabled.

Bit	Bit Name	Function
14	BLIE	Bus Lock Interrupt Enable 0: Bus lock interrupt is disabled. 1: Bus lock interrupt is enabled.
13	OLIE	Overload Frame Transmit Interrupt Enable 0: Overload frame transmit interrupt is disabled. 1: Overload frame transmit interrupt is enabled.
12	BORIE	Bus Off Recovery Interrupt Enable 0: Bus off recovery interrupt is disabled. 1: Bus off recovery interrupt is enabled.
11	BOEIE	Bus Off Entry Interrupt Enable 0: Bus off entry interrupt is disabled. 1: Bus off entry interrupt is enabled.
10	EPIE	Error Passive Interrupt Enable 0: Error passive interrupt is disabled. 1: Error passive interrupt is enabled.
9	EWIE	Error Warning Interrupt Enable 0: Error warning interrupt is disabled. 1: Error warning interrupt is enabled.
8	BEIE	Bus Error Interrupt Enable 0: Bus error interrupt is disabled. 1: Bus error interrupt is enabled.
7 to 4	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
3	RTBO	Forcible Return from Bus-off When this bit is set to 1, forcible return from the bus off state is made. This bit is always read as 0.
2	CSLPR	Channel Stop Mode 0: Other than channel stop mode 1: Channel stop mode
1, 0	CHMDC[1:0]	Mode Select b1 b0 0 0: Channel communication mode 0 1: Channel reset mode 1 0: Channel halt mode 1 1: Setting prohibited

CRCT Bit

This bit is used to test the CRC generation circuit in the RS-CANFD module. Setting this bit to 1 inverts the first bit of the ID field when a message is received. With this inversion of bit, the CRC calculation result does not match the normal CRC value of the received frame, which can detect a CRC error (the CERR bit in the RSCANnCmERFL register is 1). When using this function, note the following.

- This function is available while the CTME bit in the RSCANnCmCTR register is 1 (communication test mode enabled).
- This function cannot communicate with other CAN nodes. Use this function for inter-channel communication test (the CmICBCE bit in the RSCANnGTSTCFG register is 1).
- Bit inversion in the ID field may cause bit stuffing rule violation. In that case, no CRC error is detected but a stuff error is detected.

Modify this bit only in channel halt mode. This bit is always 0 in channel reset mode.

CTMS[1:0] Bit

These bits are used to select a communication test mode. Modify these bits in channel halt mode only. These bits are set to 0 in channel reset mode.

CTME Bit

Setting this bit to 1 enables communication test mode. Modify this bit in channel halt mode. This bit is set to 0 in channel reset mode.

ERRD Bit

This bit is used to control the display mode of bits 14 to 8 in the RSCANnCmERFL register.
When this bit is clear to 0, if any error is detected while the flags of bits 14-8 in the RSCANnCmERFL register are all 0, only the flags of the first error event are set to 1. If two or more errors occur in the first error event, all the flags of the detected errors are set to 1.
When this bit is set to 1, all the flags of errors that have occurred are set to 1 regardless of the error occurrence order.
Modify this bit only in channel reset mode or channel halt mode.

BOM[1:0] Bit

These bits are used to select the bus off recovery mode of the RS-CANFD module.
When the BOM[1:0] bits are set to 00B, return from the bus off state to the error active state is compliant with the CAN specifications. That is, the RS-CANFD module reenters the CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. A bus off recovery interrupt request is generated at the time of return from the bus off state. Even if the CHMDC[1:0] bits are set to 10B (channel halt mode) before recessive bits are detected 128 times, the RS-CANFD module does not transition to channel halt mode until recessive bits are detected 128 times.
When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to 01B, the CHMDC[1:0] bits in the RSCFDnCmCTRRSCANnCmCTR register (m = 0, 1) are set to 10B and the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated and the TEC[7:0] and REC[7:0] bits in the RSCFDnCmSTSRSCANnCmSTS register are cleared to 00H.
When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to 10B, the CHMDC[1:0] bits are set to 10B and the RS-CANFD module transitions to channel halt mode after return from the bus off state (11 consecutive recessive bits are detected 128 times). A bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00H.
When the BOM[1:0] bits are set to 11B and the CHMDC[1:0] bits are set to 10B while the RS-CANFD module is in the bus off state, the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00H. However, if 11 consecutive recessive bits are detected 128 times and the RS-CANFD module has recovered to the error active state from the bus off state before the CHMDC[1:0] bits are set to 10B, a bus off recovery interrupt request is generated.
If a program writes to the CHMDC[1:0] bit at the same time as the RS-CANFD module transition to channel halt mode (at bus off entry when the BOM[1:0] bits are 01B or at bus off end when the BOM[1:0] bits are 10B), the program's writing takes precedence. Modify the BOM[1:0] bits only in channel reset mode.

TAIE Bit

When transmit abort of the transmit buffer is completed with the TAIE bit set to 1, an interrupt request is generated.
Modify this bit only in channel reset mode.

ALIE Bit

When the ALF flag in the RSCANnCmERFL register is set to 1 with the ALIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BLIE Bit

When the BLF flag in the RSCANnCmERFL register is set to 1 with the BLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

OLIE Bit

When the OVLF flag in the RSCANnCmERFL register is set to 1 with the OLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BORIE Bit

When the BORF flag in the RSCANnCmERFL register is set to 1 with the BORIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BOEIE Bit

When the BOEF flag in the RSCANnCmERFL register is set to 1 with the BOEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EPIE Bit

When the EPF flag in the RSCANnCmERFL register is set to 1 with the EPIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EWIE Bit

When the EWF flag in the RSCANnCmERFL register is set to 1 with the EWIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BEIE Bit

When the BEF flag in the RSCANnCmERFL register is set to 1 with the BEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

RTBO Bit

Setting this bit to 1 in the bus off state forcibly returns the state from the bus off state to the error active state. This bit is automatically cleared to 0. Setting this bit to 1 clears the TEC[7:0] and REC[7:0] bits in the RSCANnCmSTS register to 00H and also clears the BOSTS flag in the RSCANnCmSTS register to 0 (not in bus off state). The other registers remain unchanged. No bus off recovery interrupt request is generated upon return from the bus off state in this case. Use this bit only when the BOM[1:0] bits in theRSCANnCmCTR register are 00B (ISO11898-1 compliant).
A delay of up to 1 CAN bit time occurs after the RTBO bit is set to 1 until the RSCAN module transitions to the error active state. Set this bit to 1 in channel communication mode.

CSLPR Bit

Setting this bit to 1 places the channel into channel stop mode.
Clearing this bit to 0 makes the channel exit channel stop mode.
This bit should not be modified in channel communication mode or channel halt mode.

CHMDC[1:0] Bit

These bits are used to select a channel mode (channel communication mode, channel reset mode, or channel halt mode). For details, see section 25.6.2, Channel Modes. Setting the CSLPR bit to 1 in channel reset mode allows transition to channel stop mode. Do not set the CHMDC[1:0] bits to 11B. When the CAN module has automatically transitioned to channel halt mode based on the setting of the BOM[1:0] bits, the CHMDC[1:0] bits automatically become 10_B.

25.3.3.3 RSCANnCmSTS — Channel Status Register (m = 0, 1)

Access: RSCANnCmSTS register can be read only in 32-bit units
RSCANnCmSTSL, RSCANnCmSTSH registers can be read only in 16-bit units
RSCANnCmSTSLL, RSCANnCmSTSLH, RSCANnCmSTSHL, RSCANnCmSTSHH registers can be read only in 8-bit units

Address: RSCANnCmSTS: <RSCFDn_base> + 0008_H + (10_H × m)
RSCANnCmSTSL: <RSCFDn_base> + 0008_H + (10_H × m),
RSCANnCmSTSH: <RSCFDn_base> + $000A_H$ + (10_H × m)
RSCANnCmSTSLL: <RSCFDn_base> + 0008_H + (10_H × m),
RSCANnCmSTSLH: <RSCFDn_base> + 0009_H + (10_H × m),
RSCANnCmSTSHL: <RSCFDn_base> + $000A_H$ + (10_H × m),
RSCANnCmSTSHH: <RSCFDn_base> + $000B_H$ + (10_H × m)

Value after reset: $0000\ 0005_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEC[7:0]								REC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	COM STS	REC STS	TRM STS	BO STS	EP STS	CSLP STS	CHLT STS	CRST STS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.18 RSCANnCmSTS Register Contents

Bit	Bit Name	Function
31 to 24	TEC[7:0]	The transmit error counter (TEC) can be read.
23 to 16	REC[7:0]	The receive error counter (REC) can be read.
15 to 8	Reserved	These bits are read as the value after reset.
7	COMSTS	Communication Status Flag 0: Communication is not ready. 1: Communication is ready.
6	RECSTS	Receive Status Flag 0: Bus idle, in transmission or bus off state 1: In reception
5	TRMSTS	Transmit Status Flag 0: Bus idle or in reception 1: In transmission or bus off state
4	BOSTS	Bus Off Status Flag 0: Not in bus off state 1: In bus off state
3	EPSTS	Error Passive Status Flag 0: Not in error passive state 1: In error passive state
2	CSLPSTS	Channel Stop Status Flag 0: Not in channel stop mode 1: In channel stop mode
1	CHLTSTS	Channel Halt Status Flag 0: Not in channel halt mode 1: In channel halt mode
0	CRSTSTS	Channel Reset Status Flag 0: Not in channel reset mode 1: In channel reset mode

TEC[7:0] Bits

These bits contain the transmit error counter value. For transmit error counter increment/decrement conditions, see the CAN specification (ISO11898-1).
These bits are cleared to 0 in channel reset mode.

REC[7:0] Bits

These bits contain the receive error counter value. For receive error counter increment/decrement conditions, see the CAN specifications (ISO11898-1).
These bits are cleared to 0 in channel reset mode.

COMSTS Flag

This bit indicates that communication is ready.
This flag becomes 1 when the CAN module has detected 11 consecutive recessive bits after it has transitioned from channel reset mode or channel halt mode to channel communication mode. This flag is cleared to 0 in channel reset mode or channel halt mode.

RECSTS Flag

This flag is set to 1 when reception has started, and is cleared to 0 when the bus has become idle or transmission has started.

TRMSTS Flag

This flag is set to 1 when transmission has started, and is cleared to 0 when the bus has become idle or reception has started. This flag remains 1 in the bus off state.

BOSTS Flag

This flag is set to 1 when the bus off state (TEC[7:0] > 255) is entered. It is cleared to 0 when the CAN module has exited the bus off state.

EPSTS Flag

This flag is set to 1 when the RS-CANFD module has entered the error passive state ((128 ≤ TEC[7:0] ≤ 255) or (128 ≤ REC[7:0])), It is cleared to 0 when the RS-CANFD module has exited the error passive state or has entered channel reset mode.

CSLPSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel stop mode, and is cleared to 0 when the CAN module has returned from channel stop mode.

CHLTSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel halt mode, and is cleared to 0 when the CAN module has returned from channel halt mode.

CRSTSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel reset mode, and is cleared to 0 when the CAN module has transitioned to channel communication mode or channel halt mode. This flag remains 1 when the CAN module transitions from channel reset mode to channel stop mode.

25.3.3.4 RSCANnCmERFL — Channel Error Flag Register (m = 0, 1)

Access: RSCANnCmERFL register can be read/written in 32-bit units
RSCANnCmERFLL, RSCANnCmERFLH registers can be read/written in 16-bit units
RSCANnCmERFLLL, RSCANnCmERFLLH, RSCANnCmERFLHL, RSCANnCmERFLHH registers can be read/written in 8-bit units

Address: RSCANnCmERFL: <RSCFDn_base> + $000C_H$ + (10_H × m)
RSCANnCmERFLL: <RSCFDn_base> + $000C_H$ + (10_H × m),
RSCANnCmERFLH: <RSCFDn_base> + $000E_H$ + (10_H × m)
RSCANnCmERFLLL: <RSCFDn_base> + $000C_H$ + (10_H × m),
RSCANnCmERFLLH: <RSCFDn_base> + $000D_H$ + (10_H × m),
RSCANnCmERFLHL: <RSCFDn_base> + $000E_H$ + (10_H × m),
RSCANnCmERFLHH: <RSCFDn_base> + $000F_H$ + (10_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CRCREG[14:0]														
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR	ALF	BLF	OVLF	BORF	BOEF	EPF	EWF	BEF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.19 RSCANnCmERFL Register Contents

Bit	Bit Name	Function
31	Reserved	When read, the value after reset is returned. When writing to this bit, write the value after reset.
30 to 16	CRCREG [14:0]	CRC Calculation Data A CRC value calculated based on the transmit message or receive message is indicated.
15	Reserved	When read, the value after reset is returned. When writing to this bit, write the value after reset.
14	ADERR	ACK Delimiter Error Flag 0: No ACK delimiter error is detected. 1: ACK delimiter error is detected.
13	B0ERR	Dominant Bit Error Flag 0: No dominant bit error is detected. 1: Dominant bit error is detected.
12	B1ERR	Recessive Bit Error Flag 0: No recessive bit error is detected. 1: Recessive bit error is detected.
11	CERR	CRC Error Flag 0: No CRC error is detected. 1: CRC error is detected.
10	AERR	ACK Error Flag 0: No ACK error is detected. 1: ACK error is detected.
9	FERR	Form Error Flag 0: No form error is detected. 1: Form error is detected.
8	SERR	Stuff Error Flag 0: No stuff error is detected. 1: Stuff error is detected.
7	ALF	Arbitration-lost Flag 0: No arbitration-lost is detected. 1: Arbitration-lost is detected.

Bit	Bit Name	Function
6	BLF	Bus Lock Flag 0: No channel bus is detected. 1: Channel bus is detected.
5	OVLF	Overload Flag 0: No overload is detected. 1: Overload is detected.
4	BORF	Bus Off Recovery Flag 0: No bus off recovery is detected. 1: Bus off recovery is detected.
3	BOEF	Bus Off Entry Flag 0: No bus off entry is detected. 1: Bus off entry is detected.
2	EPF	Error Passive Flag 0: No error passive is detected. 1: Error passive is detected.
1	EWF	Error Warning Flag 0: No error warning is detected. 1: Error warning is detected.
0	BEF	Bus Error Flag 0: No channel bus error is detected. 1: Channel bus error is detected.

See the CAN specification (ISO11898-1) for a description of error occurrence conditions. To clear each flag of this register, the program must write a 0 to the corresponding bit. These flags cannot be set to 1 by the program. If any of these flags is set to 0 at the same time that the program writes 0 to the flag, the flag is still set to 1. The channel reset mode transition clears all of these flags to 0.
If the ERRD bit in the RSCANnCmCTR register is set to 0 (ie, only the flags for the first error event are displayed) and an error related to bits 14 to 8 of RSCANnCmERFL is detected, the flag bits are only set by the error event if bits 14 to 8 were all 0 at the when time the error occurred.

CRCREG[14:0] Flag

When the CTME bit in the RSCANnCmCTR register is set to 1 (communication test mode is enabled), the CRC value calculated based on the transmit or receive message can be read. When the CTME bit is set to 0 (communication test mode is disabled), these bits are always read as 0. This bit is always 0 in channel reset mode.

ADERR Flag

This flag is set to 1 when a form error has been detected in the ACK delimiter during transmission.

B0ERR Flag

This flag is set to 1 when a recessive bit has been detected though a dominant bit was transmitted.

B1ERR Flag

This flag is set to 1 when a dominant bit has been detected though a recessive bit was transmitted.

CERR Flag

This flag is set to 1 when a CRC error has been detected.

AERR Flag

This flag is set to 1 when an ACK error has been detected.

FERR Flag

This flag is set to 1 when a form error has been detected.

SERR Flag

This flag is set to 1 when a stuff error has been detected.

ALF Flag

This flag is set to 1 when an arbitration-lost has been detected.

BLF Flag

This flag is set to 1 when 32 consecutive dominant bits have been detected on the CAN bus in channel communication mode. After that, detection of a dominant lock is restarted when either of the following conditions is met.

- A recessive bit is detected after the BLF bit has been cleared from 1 to 0.
- The CAN module transitions to channel reset mode and returns to channel communication mode after the BLF bit has been cleared from 1 to 0

OVLF Flag

This flag is set to 1 when the overload frame transmit condition has been detected when performing reception or transmission.

BORF Flag

This flag is set to 1 when 11 consecutive recessive bits have been detected 128 times and the CAN module returns from the bus off state. However, this flag is not set to 1 if the CAN module returns from the bus off state in any of the following ways before 11 consecutive recessive bits are detected 128 times.

- The CHMDC[1:0] bits in the RSCANnCmCTR register are set to 01B (channel reset mode).
- The RTBO bit in the RSCANnCmCTR register is set to 1 (forcible return from the bus off state is made).
- The BOM[1:0] bits in the RSCANnCmCTR register are set to 01B (transition to channel halt mode at bus off entry).
- The CHMDC[1:0] bits in the RSCANnCmCTR register are set to 10B (channel halt mode) before 11 consecutive recessive bits are detected 128 times with the BOM[1:0] bits set to 11B (transition to channel halt mode upon a request from the program during bus off).

BOEF Flag

This flag is set to 1 when the bus off state is reached (TEC[7:0] value > 255). This flag is also set to 1 if the bus off state is reached when the BOM[1:0] bits in the RSCANnCmCTR register (m = 0, 1) set to 01B (transition to channel halt mode at bus off entry).

EPF Flag

This flag becomes 1 when the error passive state is reached (REC[7:0] or TEC[7:0] value > 127).
This flag becomes 1 only when the REC[7:0] or TEC[7:0] value first exceeds 127. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 127, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 127 or less and then the REC[7:0] or TEC[7:0] value exceeds 127 again.

EWF Flag

This flag is set to 1 only when the REC[7:0] or TEC[7:0] value first exceeds 95. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 95, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 95 or less and then the REC[7:0] or TEC[7:0] value exceeds 95 again.

BEF Flag

This flag is set to 1 when any one of the ADERR, B0ERR, B1ERR, CERR, AERR, FERR, and SERR flags in the RSCANnCmERFL register is set to 1.

Note: To clear the flag of this register to 0, use a store instruction to write “0” to the given flag and “1” to the other flags.

25.3.4 Details of Global-related Registers

25.3.4.1 RSCANnGCFG — Global Configuration Register

Access: RSCANnGCFG register can be read/written in 32-bit units
RSCANnGCFGL, RSCANnGCFGH registers can be read/written in 16-bit units
RSCANnGCFGLL, RSCANnGCFGLH, RSCANnGCFGHL, RSCANnGCFGHH registers can be read/written in 8-bit units

Address: RSCANnGCFG: <RSCFDn_base> + 0084_H
RSCANnGCFGL: <RSCFDn_base> + 0084_H, RSCANnGCFGH: <RSCFDn_base> + 0086_H
RSCANnGCFGLL: <RSCFDn_base> + 0084_H, RSCANnGCFGLH: <RSCFDn_base> + 0085_H,
RSCANnGCFGHL: <RSCFDn_base> + 0086_H, RSCANnGCFGHH: <RSCFDn_base> + 0087_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ITRCP[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSBTCS[2:0]			TSSS	TSP[3:0]				TMTSCE	—	—	DCS	MME	DRE	DCE	TPRI
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Table 25.20 RSCANnGCFG Register Contents

Bit	Bit Name	Function
31 to 16	ITRCP[15:0]	Interval Timer Prescaler Set When these bits are set to M, the pclk is divided by M. Setting 0000H is prohibited when the interval timer is in use.
15 to 13	TSBTCS[2:0]	Timestamp Clock Source Select b15 b14 b13 0 0 0: Channel 0 bit time clock 0 0 1: Channel 1 bit time clock 0 1 0: Setting prohibited 0 1 1: Setting prohibited 1 0 0: Setting prohibited 1 0 1: Setting prohibited 1 1 0: Setting prohibited 1 1 1: Setting prohibited
12	TSSS	Timestamp Source Select 0: pclk/2*1 1: Bit time clock
11 to 8	TSP[3:0]	Timestamp Clock Source Division b11 b10 b9 b8 0 0 0 0: Not divided 0 0 0 1: Divided by 2 0 0 1 0: Divided by 4 0 0 1 1: Divided by 8 0 1 0 0: Divided by 16 0 1 0 1: Divided by 32 0 1 1 0: Divided by 64 0 1 1 1: Divided by 128 1 0 0 0: Divided by 256 1 0 0 1: Divided by 512 1 0 1 0: Divided by 1024 1 0 1 1: Divided by 2048 1 1 0 0: Divided by 4096 1 1 0 1: Divided by 8192 1 1 1 0: Divided by 16384 1 1 1 1: Divided by 32768

Bit	Bit Name	Function
7	TMTSCE	Transmission Timestamp Enable 0: Transmission timestamp is disabled. 1: Transmission timestamp is enabled.
6, 5	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
4	DCS	CAN Clock Source Select*2 0: clkc 1: clk_xincan
3	MME	Mirror Function Enable 0: Mirror function is disabled. 1: Mirror function is enabled.
2	DRE	DLC Replacement Enable 0: DLC replacement is disabled. 1: DLC replacement is enabled.
1	DCE	DLC Check Enable 0: DLC check is disabled. 1: DLC check is enabled.
0	TPRI	Transmit Priority Select 0: ID priority 1: Transmit buffer number priority

Note 1. When specifying pclk/2 as the timestamp counter count source, set bits TSBTCS[2:0] to 000B.
Note 2. For the CAN clock frequency settings, see section 25.1.3, Clock Supply.

Modify the RSCANnGCFG register only in global reset mode.

ITRCP[15:0] Bits

These bits are used to set a clock source division value of the interval timer for FIFO buffers. See section 25.8.3.1, Interval Transmission Function.

TSBTCS[2:0] Bits

When the TSSS bit is 1, these bits are used to select the channel of the bit time clock that will be the clock source of the timestamp counter.

TSSS Bits

This bit is used to select a clock source of the timestamp counter.

TSP[3:0] Bits

A clock obtained by dividing the clock source selected with the TSBTCS[2:0] bits and TSSS bit according to the TSP[3:0] bits is used as the timestamp counter count source.

TMTSCE Bits

Setting this bit to 1 makes it possible to store the timestamp of a transmitted message in the transmit history buffer. The timestamp is stored in TMTS[15:0] bits in the RSCANnTHLACCm register.

DCS Bits

When this bit is set to 0, clkc is used as the clock source of the CAN clock (fCAN).
When this bit is set to 1, clk_xincan is used as the clock source of the CAN clock (fCAN).
For the CAN clock frequency settings, see Table 25.6.

MME Bits

Setting this bit to 1 makes the mirror function available.

DRE Bits

When the DRE bit is set to 1, the DLC value of the receive rule is stored in the buffer instead of the DLC value of the received message after the DLC value has passed through the DLC filter. In this case, a value of 00H is stored in each data byte beyond the DLC value of the receive rule.
The DLC replacement function is only available when the DCE bit is set to 1 (DLC check is enabled).

DCE Bits

Setting this bit to 1 makes the DLC check function available. When disabling the DLC check function, set the GAFLDLC[3:0] bits in the RSCANnGAFLP0_j register to 0000B before clearing the DCE bit in the RSCANnGCFG register to 0.

TPRI Bits

This bit is used to set the transmit priority.
When this bit is set to 0, ID priority is selected and the transmit priority complies with the CAN bus arbitration rule (ISO11898-1 specifications). When this bit is set to 1, transmit buffer number priority is selected and the lowest transmit buffer number of those has the highest priority.
While the transmit queue is in use, this bit should be set to 0.

25.3.4.2 RSCANnGCTR — Global Control Register

Access: RSCANnGCTR register can be read/written in 32-bit units
RSCANnGCTRL, RSCANnGCTRH registers can be read/written in 16-bit units
RSCANnGCTRLL, RSCANnGCTRLH, RSCANnGCTRHL, RSCANnGCTRHH registers can be read/written in 8-bit units

Address: RSCANnGCTR: <RSCFDn_base> + 0088_H
RSCANnGCTRL: <RSCFDn_base> + 0088_H, RSCANnGCTRH: <RSCFDn_base> + $008A_H$
RSCANnGCTRLL: <RSCFDn_base> + 0088_H, RSCANnGCTRLH: <RSCFDn_base> + 0089_H,
RSCANnGCTRHL: <RSCFDn_base> + $008A_H$, RSCANnGCTRHH: <RSCFDn_base> + $008B_H$

Value after reset: $0000\ 0005_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSRST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLEIE	MEIE	DEIE	—	—	—	—	—	GSLPR	GMDC[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.21 RSCANnGCTR Register Contents

Bit	Bit Name	Function
31 to 17	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
16	TSRST	Timestamp Counter Reset Setting the TSRST bit to 1 resets the timestamp counter. This bit is always read as 0.
15 to 11	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
10	THLEIE	Transmit History Buffer Overflow Interrupt Enable 0: Transmit history buffer overflow interrupt is disabled. 1: Transmit history buffer overflow interrupt is enabled.
9	MEIE	FIFO Message Lost Interrupt Enable 0: FIFO message lost interrupt is disabled. 1: FIFO message lost interrupt is enabled.
8	DEIE	DLC Error Interrupt Enable 0: DLC error interrupt is disabled. 1: DLC error interrupt is enabled.
7 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	GSLPR	Global Stop Mode 0: Other than global stop mode 1: Global stop mode
1, 0	GMDC[1:0]	Global Mode Select b1 b0 0 0: Global operating mode 0 1: Global reset mode 1 0: Global test mode 1 1: Setting prohibited

TSRST Bit

This bit is used to reset the timestamp counter. When this bit is set to 1, the RSCANnGTSCregister is cleared to 0000H.

THLEIE Bit

When the THLEIE bit is set to 1 and the THLES flag in theRSCANnGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

MEIE Bit

When the MEIE bit is set to 1 and the MES flag in theRSCANnGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

DEIE Bit

When the DEIE bit is set to 1 and the DEF flag in theRSCANnGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

GSLPR Bit

Setting this bit to 1 places the RSCAN module into global stop mode.
Clearing this bit to 0 makes the RSCAN module leave from global stop mode.
This bit should not be modified in global operating mode or global test mode.

GMDC[1:0] Bit

These bits are used to select the mode of entire RS-CANFD module (global operating mode, global reset mode, or global test mode). For details, see section 25.6.1, Global Modes. Setting the GSLPR bit to 1 when in global reset mode places the RS-CANFD module into global stop mode.

25.3.4.3 RSCANnGSTS — Global Status Register

Access: RSCANnGSTS register can be read only in 32-bit units
RSCANnGSTSL, RSCANnGSTSH registers can be read only in 16-bit units
RSCANnGSTSLL, RSCANnGSTSLH, RSCANnGSTSHL, RSCANnGSTSHH registers can be read only in 8-bit units

Address: RSCANnGSTS: <RSCFDn_base> + $008C_H$
RSCANnGSTSL: <RSCFDn_base> + $008C_H$, RSCANnGSTSH: <RSCFDn_base> + $008E_H$
RSCANnGSTSLL: <RSCFDn_base> + $008C_H$, RSCANnGSTSLH: <RSCFDn_base> + $008D_H$,
RSCANnGSTSHL: <RSCFDn_base> + $008E_H$, RSCANnGSTSHH: <RSCFDn_base> + $008F_H$

Value after reset: $0000\ 000D_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	GRAM INIT	GSLP STS	GHLT STS	GRST STS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.22 RSCANnGSTS Register Contents

Bit	Bit Name	Function
31 to 4	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
3	GRAMINIT	CAN RAM Initialization Status Flag 0: CAN RAM initialization is completed. 1: CAN RAM initialization is ongoing.
2	GSLPSTS	Global Stop Status Flag 0: Not in global stop mode 1: In global stop mode
1	GHLTSTS	Global Test Status Flag 0: Not in global test mode 1: In global test mode
0	GRSTSTS	Global Reset Status Flag 0: Not in global reset mode 1: In global reset mode

GRAMINIT Flag

This flag indicates the initialization status of the CAN RAM.
This flag is set to 1 after the MCU has been reset, and is cleared to 0 when CAN RAM initialization is completed.

GSLPSTS Flag

This flag is set to 1 when the CAN module has transitioned to global stop mode, and is cleared to 0 when the CAN module has returned from global stop mode.

GHLTSTS Flag

This flag is set to 1 when the CAN module has transitioned to global test mode, and is cleared to 0 when the CAN module has exited global test mode.

GRSTSTS Flag

This flag is set to 1 when the CAN module has transitioned to global reset mode, and is cleared to 0 when the CAN module has exited global reset mode. This flag remains 1 even when the CAN module has transitioned from global reset mode to global stop mode.

25.3.4.4 RSCANnGERFL — Global Error Flag Register

Access: RSCANnGERFL register can be read/written in 32-bit units
RSCANnGERFLL, RSCANnGERFLH registers can be read/written in 16-bit units
RSCANnGERFLLL, RSCANnGERFLLH, RSCANnGERFLHL, RSCANnGERFLHH registers can be read/written in 8-bit units

Address: RSCANnGERFL: <RSCFDn_base> + 0090_H
RSCANnGERFLL: <RSCFDn_base> + 0090_H, RSCANnGERFLH: <RSCFDn_base> + 0092_H
RSCANnGERFLLL: <RSCFDn_base> + 0090_H, RSCANnGERFLLH: <RSCFDn_base> + 0091_H,
RSCANnGERFLHL: <RSCFDn_base> + 0092_H, RSCANnGERFLHH: <RSCFDn_base> + 0093_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	THLES	MES	DEF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*

Note: The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.23 RSCANnGERFL Register Contents

Bit	Bit Name	Function
31 to 14, 7, 6, 4, 3	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
13 to 8, 5	Reserved	When read, an undefined value is returned.The write value should be the value after reset.
2	THLES	Transmit History Buffer Overflow Status Flag 0: No transmit history buffer overflow has occurred. 1: A transmit history buffer overflow has occurred.
1	MES	FIFO Message Lost Status Flag 0: No FIFO message lost error has occurred. 1: A FIFO message lost error has occurred.
0	DEF	DLC Error Flag 0: No DLC error has occurred. 1: A DLC error has occurred.

All flags in the RSCANnGERFL register are cleared to 0 in global reset mode.

THLES Flag

The THLES flag is set to 1 when any one of the THLELT flags in the RSCANnTHLSTSm register (m = 0, 1) is set to 1.
This flag is cleared to 0 when the THLELT flags of all channels are set to 0.

MES Flag

The MES flag is set to 1 when any one of the RFMLT flags in theRSCANnRFSTSx register (x = 0 to 7) or the CFMLT flags in the RSCANnCFSTSk register (k = 0 to 5) is set to 1.
This flag is cleared to 0 when all RFMLT flags and CFMLT flags are set to 0.

DEF Flag

The DEF flag is set to 1 when an error has been detected during the DLC check. The program can clear this flag by writing 0 to this bit.

Note: To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”

25.3.4.5 RSCANnGTSC — Global Timestamp Counter Register

Access: RSCANnGTSC register can be read only in 32-bit units.
RSCANnGTSCL, RSCANnGTSCH registers can be read only in 16-bit units.

Address: RSCANnGTSC: <RSCFDn_base> + 0094_H
RSCANnGTSCL: <RSCFDn_base> + 0094_H, RSCANnGTSCH: <RSCFDn_base> + 0096_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.24 RSCANnGTSC Register Contents

Bit	Bit Name	Function
31 to 16	Reserved	These bits are read as the value after reset.
15 to 0	TS[15:0]	Timestamp Value The timestamp counter value can be read. Counter Value: 0000_H to $FFFF_H$

TS[15:0] Bits

When the TS[15:0] bits are read, the read value shows the timestamp counter (16-bit free-running counter) value at that time. When the SOF is detected, the TS[15:0] value is captured and later stored in the receive buffer or the FIFO buffer. Furthermore, while the TMTSCE bit in the RSCANnGCFG register is 1, the TS[15:0] value is stored in the transmit history buffer. The timestamp counter is initialized in global reset mode.
The timestamp counter start timing and stop timing depend on the count source.

- When the TSSS bit in the GRSCANnGCFG register is 0 (pclk):
  The timestamp counter starts counting when the RSCAN module has transitioned to global operating mode.
  This counter stops counting when the RSCAN module has transitioned to global stop mode or global test mode.
- When the TSSS bit is 1 (CANm bit time clock):
  The timestamp counter starts counting when the corresponding channel has transitioned to channel communication mode.
  This counter stops counting when the corresponding channel has transitioned to channel reset mode or channel halt mode.

25.3.4.6 RSCANnGTINTSTS0 — Global TX Interrupt Status Register 0

Access: RSCANnGTINTSTS0 register can be read only in 32-bit units
RSCANnGTINTSTS0L, RSCANnGTINTSTS0H registers can be read only in 16-bit units
RSCANnGTINTSTS0LL, RSCANnGTINTSTS0LH, RSCANnGTINTSTS0HL, RSCANnGTINTSTS0HH registers can be read only in 8-bit units

Address: RSCANnGTINTSTS0: <RSCFDn_base> + 0460_H
RSCANnGTINTSTS0L: <RSCFDn_base> + 0460_H, RSCANnGTINTSTS0H: <RSCFDn_base> + 0462_H
RSCANnGTINTSTS0LL: <RSCFDn_base> + 0460_H, RSCANnGTINTSTS0LH: <RSCFDn_base> + 0461_H,
RSCANnGTINTSTS0HL: <RSCFDn_base> + 0462_H, RSCANnGTINTSTS0HH: <RSCFDn_base> + 0463_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1	—	—	—	THIF0	CFTIF0	TQIF0	TAIF0	TSIF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Note 1. This bit is automatically cleared in the global reset or channel reset mode.

Table 25.25 RSCANnGTINTSTS0 Register Contents

Bit	Bit Name	Function
31 to 13	Reserved	These bits are read as the value after reset.
12	THIF1	Channel 1 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
11	CFTIF1	Channel 1 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
10	TQIF1	Channel 1 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
9	TAIF1	Channel 1 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
8	TSIF1	Channel 1 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.
7 to 5	Reserved	These bits are read as the value after reset.
4	THIF0	Channel 0 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
3	CFTIF0	Channel 0 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
2	TQIF0	Channel 0 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
1	TAIF0	Channel 0 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.

Bit	Bit Name	Function
0	TSIF0	Channel 0 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.

TSIFm Bits

The TSIFm bit is set to 1 when the TMIEp bit in the RSCANnTMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCANnTMSTSp register are set to 10B (transmit completed without abort request) or 11B (transmit completed with abort request).
When the TMTRF[1:0] flags are cleared to 00B under the condition that the TSIFm bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIEp bit to 0 also clears this flag to 0.

TAIFm Bits

The TAIFm bit is set to 1 when the TAIE bit in the RSCANnCmCTR register is 1 (transmit abort interrupt enabled) and the TMTRF[1:0] flags in the RSCANnTMSTSp register are set to 01B (transmit abort completed).
This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to 00B after the transmit abort is completed.

TQIFm Bits

When the TXQIE bit in the RSCANnTXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCANnTXQSTSm register is set to 1 (transmit queue interrupt request), the TQIFm bit is set to 1.
When the TXQIF bit (transmit queue interrupt request) in the RSCANnTXQSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the TXQIE bit is cleared to 0.

CFTIFm Bits

When the CFTXIE bit in the RSCANnCFCCk register is set to 1 (transmit/receive FIFO transmit interrupt enabled) and the CFTXIF bit in the RSCANnCFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), the CFTIFm bit is set to 1.
When the CFTXIF bit is cleared to 0 under the conditions that the CFTIFm bit can be set to 1, this bit is cleared to 0. This flag is also cleared to 0 when the CFTXIE bit is cleared to 0.

THIFm Bits

When the THLIE bit in the RSCANnTHLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCANnTHLSTSm register is set to 1 (transmit history interrupt request), the THIFm bit is set to 1.
When the THLIF bit in the RSCANnTHLSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the THLIE bit is cleared to 0.

25.3.4.7 RSCANnGFDCFG — Global FD configuration register

Access: RSCANnGFDCFG register can be read/written in 32-bit unit
RSCANnGFDCFGL and RSCANnGFDCFGH registers can be read/written in 16-bit unit
RSCANnGFDCFGLL, RSCANnGFDCFGLH, RSCANnGFDCFGHL, RSCANnGFDCFGHH registers can be read/written in 8-bit unit

Address: RSCANnGFDCFG: <RSCFDn_base> + 0474_H
RSCANnGFDCFG L: <RSCFDn_base> + 0474_H, RSCANnGFDCFG H: <RSCFDn_base> + 0476_H
RSCANnGFDCFG LL: <RSCFDn_base> + 0474_H, RSCANnGFDCFG LH: <RSCFDn_base> + 0475_H,
RSCANnGFDCFG HL: <RSCFDn_base> + 0476_H, RSCANnGFDCFG HH: <RSCFDn_base> + 0477_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSCCFG[1:0]		—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.26 RSCANnGFDCFG Register Contents

Bit	Bit Name	Function
31 to 10	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
9, 8	TSCCFG[1:0]	Time-stamp capture setting bit b9 b8 0 0: Captured at a sample point in the SOF bit. 0 1: Captured when a valid frame has been transmitted/received. 1 0: Setting prohibited 1 1: Setting prohibited
7 to 1	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
0	Reserved	When read, an undefined value is returned.The write value should be the value after reset.

TSCCFG bit

Select a point where a time-stamp value is captured. Modify this bit only in global reset mode.

25.3.5 Details of Receive Rule-related Registers

25.3.5.1 RSCANnGAFLECTR — Receive Rule Entry Control Register

Access: RSCANnGAFLECTR register can be read/written in 32-bit units
RSCANnGAFLECTRL, RSCANnGAFLECTRH registers can be read/written in 16-bit units
RSCANnGAFLECTRLL, RSCANnGAFLECTRLH, RSCANnGAFLECTRHL, RSCANnGAFLECTRHH registers can be read/written in 8-bit units

Address: RSCANnGAFLECTR: <RSCFDn_base> + 0098_H
RSCANnGAFLECTRL: <RSCFDn_base> + 0098_H, RSCANnGAFLECTRH: <RSCFDn_base> + $009A_H$
RSCANnGAFLECTRLL: <RSCFDn_base> + 0098_H, RSCANnGAFLECTRLH: <RSCFDn_base> + 0099_H,
RSCANnGAFLECTRHL: <RSCFDn_base> + $009A_H$, RSCANnGAFLECTRHH: <RSCFDn_base> + $009B_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AFLDAE	—	—	—	AFLPN[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 25.27 RSCANnGAFLECTR Register Contents

Bit	Bit Name	Function
31 to 9	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
8	AFLDAE	Receive Rule Table Write Enable 0: Receive rule table write is disabled. 1: Receive rule table write is enabled.
7 to 5	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
4 to 0	AFLPN[4:0]	Receive Rule Table Page Number Configuration A page number can be selected from a range of page 0 (00000_B) to page 7 (00111_B).

AFLDAE Bit

Setting this bit to 0 disables the write to the receive rule table. After writes to the receive rule table are completed, set this bit to 0 to disable the write to the table. The receive rule table can be read regardless of the value of this bit.
Set the AFLDAE bit to 1 only in global reset mode.

AFLPN[4:0] Bits

These bits are used to set the page number of the receive rule table. Sixteen receive rules can be set per page.
Set these bits to a value within the range of 00000_B to 00111_B.

25.3.5.2 RSCANnGAFLCFG0 — Receive Rule Configuration Register 0

Access: RSCANnGAFLCFG0 register can be read/written in 32-bit units
RSCANnGAFLCFG0L, RSCANnGAFLCFG0H registers can be read/written in 16-bit units
RSCANnGAFLCFG0LL, RSCANnGAFLCFG0LH, RSCANnGAFLCFG0HL, RSCANnGAFLCFG0HH registers can be read/written in 8-bit units

Address: RSCANnGAFLCFG0: <RSCFDn_base> + $009C_H$
RSCANnGAFLCFG0L: <RSCFDn_base> + $009C_H$, RSCANnGAFLCFG0H: <RSCFDn_base> + $009E_H$
RSCANnGAFLCFG0LL: <RSCFDn_base> + $009C_H$, RSCANnGAFLCFG0LH: <RSCFDn_base> + $009D_H$,
RSCANnGAFLCFG0HL: <RSCFDn_base> + $009E_H$, RSCANnGAFLCFG0HH: <RSCFDn_base> + $009F_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RNC0[7:0]								RNC1[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.28 RSCANnGAFLCFG0 Register Contents

Bit	Bit Name	Function
31 to 24	RNC0[7:0]	Number of Rules for Channel 0 Set the number of receive rules exclusively used for channel 0.
23 to 16	RNC1[7:0]	Number of Rules for Channel 1 Set the number of receive rules exclusively used for channel 1.
15 to 0	Reserved	These bits are read as the value after reset. The write value should be the value after reset.

Modify the RSCANnGAFLCFG0 register only in global reset mode.
Up to 64 × (number of channels) rules can be registered in the receive rule table as the entire unit. The number of receive rules per channel should meet the following conditions.

- The maximum number of rules per channel is 128.
- The total of the number of rules allocated to each channel is not larger than the number of rules that can be registered in the entire unit.

RNC0[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 0 receive rule table.
Set these bits to a value within the range of 00_H to 80_H.

RNC1[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 1 receive rule table.
Set these bits to a value within the range of 00_H to 80_H.

25.3.5.3 RSCANnGAFLIDj — Receive Rule ID Register (j = 0 to 15)

Access: RSCANnGAFLIDj register can be read/written in 32-bit units
RSCANnGAFLIDjL, RSCANnGAFLIDjH registers can be read/written in 16-bit units
RSCANnGAFLIDjLL, RSCANnGAFLIDjLH, RSCANnGAFLIDjHL, RSCANnGAFLIDjHH registers can be read/written in 8-bit units

Address: RSCANnGAFLIDj: <RSCFDn_base> + 0500_H + (10_H × j)
RSCANnGAFLIDjL: <RSCFDn_base> + 0500_H + (10_H × j),
RSCANnGAFLIDjH: <RSCFDn_base> + 0502_H + (10_H × j)
RSCANnGAFLIDjLL: <RSCFDn_base> + 0500_H + (10_H × j),
RSCANnGAFLIDjLH: <RSCFDn_base> + 0501H + (10_H × j),
RSCANnGAFLIDjHL: <RSCFDn_base> + 0502H + (10_H × j),
RSCANnGAFLIDjHH: <RSCFDn_base> + 0503H + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDE	GAFLRTR	GAFLLB	GAFLID[28:16]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLID[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.29 RSCANnGAFLIDj Register Contents

Bit	Bit Name	Function
31	GAFLIDE	IDE Select 0: Standard ID 1: Extended ID
30	GAFLRTR	RTR Select 0: Data frame 1: Remote frame
29	GAFLLB	Receive Rule Target Message Select 0: When a message transmitted from another CAN node is received 1: When the own transmitted message is received
28 to 0	GAFLID [28:0]	ID Set the ID of the receive rule. For the standard ID, set the ID in bits b10 to b0 and set bits b28 to b11 to 0.

Modify the RSCANnGAFLIDj register when the AFLDAE bit in the RSCANnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDE Bit

This bit is used to select the ID format (standard ID or extended ID) of the receive rule. This bit is compared with the IDE bit in the received message during the acceptance filter processing.

GAFLRTR Bit

This bit is used to select the frame format (data frame or remote frame) of the receive rule. This bit is compared with the RTR bit in the received message during the acceptance filter processing.

GAFLLB Bit

When this bit is set to 0, data processing using the receive rule is performed when receiving messages transmitted from another CAN node.
When this bit is set to 1 when the mirror function is used, data processing using the receive rule is performed when the CAN node is receiving its own transmitted messages.

GAFLID[28:0] Bits

These bits are used to set the ID field of the receive rule. The ID value set by these bits is compared with the ID of the received message during the acceptance filter processing.

25.3.5.4 RSCANnGAFLMj — Receive Rule Mask Register (j = 0 to 15)

Access: RSCANnGAFLMj register can be read/written in 32-bit units
RSCANnGAFLMjL, RSCANnGAFLMjH registers can be read/written in 16-bit units
RSCANnGAFLMjLL, RSCANnGAFLMjLH, RSCANnGAFLMjHL, RSCANnGAFLMjHH registers can be read/written in 8-bit units

Address: RSCANnGAFLMj: <RSCFDn_base> + 0504_H+ (10_H × j)
RSCANnGAFLMjL: <RSCFDn_base> + 0504_H + (10_H × j),
RSCANnGAFLMjH: <RSCFDn_base> + 0506_H + (10_H × j)
RSCANnGAFLMjLL: <RSCFDn_base> + 0504_H + (10_H × j),
RSCANnGAFLMjLH: <RSCFDn_base> + 0505_H + (10_H × j),
RSCANnGAFLMjHL: <RSCFDn_base> + 0506_H + (10_H× j),
RSCANnGAFLMjHH: <RSCFDn_base> + 0507_H + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDEM	GAFLRTRM	—	GAFLIDM[28:16]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLIDM[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.30 RSCANnGAFLMj Register Contents

Bit	Bit Name	Function
31	GAFLIDEM	IDE Mask 0: The IDE bit is not compared. 1: The IDE bit is compared.
30	GAFLRTRM	RTR Mask 0: The RTR bit is not compared. 1: The RTR bit is compared
29	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
28 to 0	GAFLIDM [28:0]	ID Mask 0: The corresponding ID bit is not compared. 1: The corresponding ID bit is compared.

Modify the RSCANnGAFLMj register when the AFLDAE bit in the RSCANnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDEM Bit

When this bit is set to 1, filter processing is performed only for messages of the ID format specified by the GAFLIDE bit in the RSCANnGAFLIDj register.
When this bit is cleared to 0, the IDs of all the receive messages and the specified IDs are regarded as matched. To set the GAFLIDEM bit to 0, set the GAFLIDM[28:0] bits to all 0 at the same time.

GAFLRTRM Bit

This bit is used to mask the RTR bit of the receive rule.

GAFLIDM[28:0] Bits

These bits are used to mask the corresponding ID bit of the receive rule.

25.3.5.5 RSCANnGAFLP0_j — Receive Rule Pointer 0 Register (j = 0 to 15)

Access: RSCANnGAFLP0_j register can be read/written in 32-bit units
RSCANnGAFLP0_jL, RSCANnGAFLP0_jH registers can be read/written in 16-bit units
RSCANnGAFLP0_jLL, RSCANnGAFLP0_jLH, RSCANnGAFLP0_jHL, RSCANnGAFLP0_jHH registers can be read/written in 8-bit units

Address: RSCANnGAFLP0_j: <RSCFDn_base> + 0508_H + (10_H × j)
RSCANnGAFLP0_jL: <RSCFDn_base> + 0508_H + (10_H × j),
RSCANnGAFLP0_jH: <RSCFDn_base> + $050A_H$ + (10_H × j)
RSCANnGAFLP0_jLL: <RSCFDn_base> + 0508_H + (10_H × j),
RSCANnGAFLP0_jLH: <RSCFDn_base> + 0509_H + (10_H × j),
RSCANnGAFLP0_jHL: <RSCFDn_base> + $050A_H$ + (10_H × j),
RSCANnGAFLP0_jHH: <RSCFDn_base> + $050B_H$ + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLDLC[3:0]				GAFLPTR[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLRMV	GAFLRMDP[6:0]							—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Table 25.31 RSCANnGAFLP0_j Register Contents

Bit	Bit Name	Function
31 to 28	GAFLDLC [3:0]	Receive Rule DLC b31 b30 b29 b28 0 0 0 0: DLC check is disabled. 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	GAFLPTR [11:0]	Receive Rule Label Set the 12-bit label information.
15	GAFLRMV	Receive Buffer Enable 0: No receive buffer is used. 1: A receive buffer is used.
14 to 8	GAFLRMDP [6:0]	Receive Buffer Number Select Set the receive buffer number to store receive messages.
7 to 0	Reserved	These bits are read as the value after reset. The write value should be the value after reset.

Modify the RSCANnGAFLP0_j register when the AFLDAE bit in the RSCANnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLDLC[3:0] Bits

These bits are used to set the minimum data length necessary for receiving messages. If the data length of a message that is being filtered is equal to or larger than the value set by the GAFLDLC[3:0] bits, the message passes the DLC check. Setting these bits to 0000_B disables the DLC check function allowing messages with any data length to pass the DLC check.

GAFLPTR[11:0] Bits

These bits are used to set a 12-bit label to be attached to messages that have passed through the filter. A label is attached when a message is stored in the receive buffer or the FIFO buffer.

GAFLRMV Bit

When this bit is set to 1, receive messages that have passed through the filter are stored in the receive buffer selected by the GAFLRMDP[6:0] bits.

GAFLRMDP[6:0] Bits

These bits are used to select the number of the receive buffer that stores receive messages that have passed through the filter when the GAFLRMV bit is set to 1. Set these bits to a value smaller than the value set by the NRXMB[7:0] bits in the RSCANnRMNB register.

25.3.5.6 RSCANnGAFLP1_j — Receive Rule Pointer 0 Register (j = 0 to 15)

Access: RSCANnGAFLP1_j register can be read/written in 32-bit units
RSCANnGAFLP1_jL, RSCANnGAFLP1_jH registers can be read/written in 16-bit units
RSCANnGAFLP1_jLL, RSCANnGAFLP1_jLH, RSCANnGAFLP1_jHL, RSCANnGAFLP1_jHH registers can be read/written in 8-bit units

Address: RSCANnGAFLP1_j: <RSCFDn_base> + 050CH + (10_H × j)
RSCANnGAFLP1_jL: <RSCFDn_base> + 050CH + (10_H × j),
RSCANnGAFLP1_jH: <RSCFDn_base> + 050EH + (10_H × j)
RSCANnGAFLP1_jLL: <RSCFDn_base> + 050CH + (10_H × j),
RSCANnGAFLP1_jLH: <RSCFDn_base> + 050DH + (10_H × j),
RSCANnGAFLP1_jHL: <RSCFDn_base> + 050EH + (10_H × j),
RSCANnGAFLP1_jHH: <RSCFDn_base> + 050FH + (10_H × j)

Value after reset: 0000 0000_H

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	GAFLFDP[13:0]													
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.32 RSCANnGAFLP1_j Register Contents

Bit	Bit Name	Function
31 to 14	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
13 to 8	GAFLFDP [13:8]	Transmit/Receive FIFO Buffer k Select (Bit position –8 = target transmit/receive FIFO buffer number k) 0: Transmit/receive FIFO buffer is not selected. 1: Transmit/receive FIFO buffer is selected.
7 to 0	GAFLFDP [7:0]	Receive FIFO Buffer x Select (Bit position = target receive FIFO buffer number x) 0: Receive FIFO buffer is not selected. 1: Receive FIFO buffer is selected.

Modify the RSCANnGAFLP1_j register when the AFLDAE bit in the RSCANnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLFDP[13:0] Bits

These bits are used to specify FIFO buffers that store receive messages that have passed through the filter. Up to eight FIFO buffers are selectable. However, when the GAFLRMV bit in the RSCANnGAFLP0_j register is set to 1 (a message is stored in the receive buffer), up to seven FIFO buffers can be selected. Only receive FIFO buffers and the transmit/receive FIFO buffer for which the CFM[1:0] bits in the RSCANnCFCCk register are set to 00B (receive mode) or 10B (gateway mode) are selectable.

25.3.6 Details of Receive Buffer-related Registers

25.3.6.1 RSCANnRMNB — Receive Buffer Number Register

Access: RSCANnRMNB register can be read/written in 32-bit units
RSCANnRMNBL, RSCANnRMNBH registers can be read/written in 16-bit units
RSCANnRMNBLL, RSCANnRMNBLH, RSCANnRMNBHL, RSCANnRMNBHH registers can be read/written in 8-bit units

Address: RSCANnRMNB: <RSCFDn_base> + $00A4_H$
RSCANnRMNBL: <RSCFDn_base> + $00A4_H$, RSCANnRMNBH: <RSCFDn_base> + $00A6_H$
RSCANnRMNBLL: <RSCFDn_base> + $00A4_H$, RSCANnRMNBLH: <RSCFDn_base> + $00A5_H$,
RSCANnRMNBHL: <RSCFDn_base> + $00A6_H$, RSCANnRMNBHH: <RSCFDn_base> + $00A7_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	NRXMB[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.33 RSCANnRMNB Register Contents

Bit	Bit Name	Function
31 to 8	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
7 to 0	NRXMB[7:0]	Receive Buffer Number Configuration Set the number of receive buffers. Set a value of 0 to 32.

Modify the RSCANnRMNB register only in global reset mode.

NRXMB[7:0] Bits

These bits are used to set the total number of receive buffers of the RS-CANFD module. The maximum value is 16 × (number of channels).

Setting these bits all to 0 makes receive buffers unavailable.

25.3.6.2 RSCANnRMNDy — Receive Buffer New Data Register (y = 0)

Access: RSCANnRMNDy register can be read/written in 32-bit units
RSCANnRMNDyL, RSCANnRMNDyH registers can be read/written in 16-bit units
RSCANnRMNDyLL, RSCANnRMNDyLH, RSCANnRMNDyHL, RSCANnRMNDyHH registers can be read/written in 8-bit units

Address: RSCANnRMNDy: <RSCFDn_base> + $00A8_H$ + (04_H × y)
RSCANnRMNDyL: <RSCFDn_base> + $00A8_H$ + (04_H × y),
RSCANnRMNDyH: <RSCFDn_base> + $00AA_H$ + (04_H × y)
RSCANnRMNDyLL: <RSCFDn_base> + $00A8_H$ + (04_H × y),
RSCANnRMNDyLH: <RSCFDn_base> + $00A9_H$ + (04_H × y),
RSCANnRMNDyHL: <RSCFDn_base> + $00AA_H$ + (04_H × y),
RSCANnRMNDyHH: <RSCFDn_base> + $00AB_H$ + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMNSq (q = y × 32 + 31 to y × 32 + 16 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMNSq (q = y × 32 + 15 to y × 32 + 0 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.34 RSCANnRMNDy Register Contents

Bit	Bit Name	Function
31 to 16	RMNSq	Receive Buffer Receive Complete Flag q (q = y × 32 + 31 to y × 32 + 16) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.
15 to 0	RMNSq	Receive Buffer Receive Complete Flag q (q = y × 32 + 15 to y × 32 + 0) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.

Write 0 to the RSCANnRMNDy register in global operating mode or global test mode.

RMNSq Flags (q = 0 to 31)

Each RMNSq flag is set to 1 when the processing for storing a message in the corresponding receive buffer starts.
To clear a flag to 0, the program must write 0 to the flag. Use a store instruction to write “0” to the flag and “1” to other flags. These bits cannot be set to 0 while a message is being stored. It takes ten clock cycles of pclk to store a message.
These flags are cleared to 0 in global reset mode.

25.3.6.3 RSCANnRMIDq — Receive Buffer ID Register (q = 0 to 31)

Access: RSCANnRMIDq register can be read only in 32-bit units
RSCFDnRMIDqRSCANnRMIDqL, RSCFDnRMIDqRSCANnRMIDqH registers can be read only in 16-bit units
RSCFDnRMIDqRSCANnRMIDqLL, RSCFDnRMIDqRSCANnRMIDqLH, RSCFDnRMIDqRSCANnRMIDqHL, RSCFDnRMIDqRSCAN-nRMIDqHH registers can be read only in 8-bit units

Address: RSCANnRMIDq: <RSCFDn_base> + 0600_H + (10_H × q)
RSCANnRMIDqL: <RSCFDn_base> + 0600_H + (10_H × q),
RSCANnRMIDqH: <RSCFDn_base> + 0602_H + (10_H × q)
RSCANnRMIDqLL: <RSCFDn_base> + 0600_H + (10_H × q),
RSCANnRMIDqLH: <RSCFDn_base> + 0601_H + (10_H × q),
RSCANnRMIDqHL: <RSCFDn_base> + 0602_H + (10_H × q),
RSCANnRMIDqHH: <RSCFDn_base> + 0603_H + (10_H × q)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMIDE	RMRTR	—	RMID[28:16]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMID[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.35 RSCANnRMIDq Register Contents

Bit	Bit Name	Function
31	RMIDE	Receive Buffer IDE 0: Standard ID 1: Extended ID
30	RMRTR	Receive Buffer RTR 0: Data frame 1: Remote frame
29	Reserved	This bit is read as the value after reset.
28 to 0	RMID[28:0]	Receive Buffer ID Data These bits contain the standard ID or extended ID of the received message. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RMIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive buffer

RMRTR Bit

This bit indicates the frame format (data frame or remote frame) of the message stored in the receive buffer.

RMID[28:0] Bits

These bits contain the ID of the message stored in the receive buffer.

25.3.6.4 RSCANnRMPTRq — Receive Buffer Pointer Register (q = 0 to 31)

Access: RSCANnRMPTRq register can be read only in 32-bit units
RSCFDnRMPTRqRSCANnRMPTRqL, RSCFDnRMPTRqRSCANnRMPTRqH registers can be read only in 16-bit units
RSCFDnRMPTRqRSCANnRMPTRqLL, RSCFDnRMPTRqRSCANnRMPTRqLH, RSCFDnRMPTRqRSCANnRMPTRqHL, RSCFDnRMPTRqRSCANnRMPTRqHH registers can be read only in 8-bit units

Address: RSCANnRMPTRq: <RSCFDn_base> + 0604_H + (10_H × q)
RSCANnRMPTRqL: <RSCFDn_base> + 0604_H + (10_H × q),
RSCANnRMPTRqH: <RSCFDn_base> + 0606_H + (10_H × q)
RSCANnRMPTRqLL: <RSCFDn_base> + 0604_H + (10_H × q),
RSCANnRMPTRqLH: <RSCFDn_base> + 0605_H + (10_H × q),
RSCANnRMPTRqHL: <RSCFDn_base> + 0606_H + (10_H × q),
RSCANnRMPTRqHH: <RSCFDn_base> + 0607_H + (10_H × q)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDLC[3:0]				RMPTR[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMTS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.36 RSCANnRMPTRq Register Contents

Bit	Bit Name	Function
31 to 28	RMDLC[3:0]	Receive Buffer DLC Data b31 b30 b29 b28 0 0 0 0: No data byte 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	RMPTR[11:0]	Receive Buffer Label Data Label information of the received message.
15 to 0	RMTS[15:0]	Receive Buffer Timestamp Data Timestamp value of the received message.

RMDLC[3:0] Bits

These bits indicate the data length of the message stored in the receive buffer.

RMPTR[11:0] Bits

These bits indicate the label information of the message stored in the receive buffer.

RMTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the receive buffer.

25.3.6.5 RSCANnRMDF0_q — Receive Buffer Data Field 0 Register (q = 0 to 31)

Access: RSCANnRMDF0_q register can be read only in 32-bit units
RSCANnRMDF0_qL, RSCANnRMDF0_qH registers can be read only in 16-bit units
RSCANnRMDF0_qLL, RSCANnRMDF0_qLH, RSCANnRMDF0_qHL, RSCANnRMDF0_qHH registers can be read only in 8-bit units

Address:
RSCANnRMDF0_q: <RSCFDn_base> + 0608_H + (10_H × q)
RSCANnRMDF0_qL: <RSCFDn_base> + 0608_H + (10_H × q),
RSCANnRMDF0_qH: <RSCFDn_base> + $060A_H$ + (10_H × q)
RSCANnRMDF0_qLL: <RSCFDn_base> + 0608_H + (10_H × q),
RSCANnRMDF0_qLH: <RSCFDn_base> + 0609_H + (10_H × q),
RSCANnRMDF0_qHL: <RSCFDn_base> + $060A_H$ + (10_H × q),
RSCANnRMDF0_qHH: <RSCFDn_base> + $060B_H$ + (10_H × q)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB3[7:0]								RMDB2[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB1[7:0]								RMDB0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.37 RSCANnRMDF0_q Register Contents

Bit	Bit Name	Function
31 to 24	RMDB3[7:0]	Receive Buffer Data Byte 3
23 to 16	RMDB2[7:0]	Receive Buffer Data Byte 2
15 to 8	RMDB1[7:0]	Receive Buffer Data Byte 1
7 to 0	RMDB0[7:0]	Receive Buffer Data Byte 0 Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCANnRMPTRq register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

25.3.6.6 RSCANnRMDF1_q — Receive Buffer Data Field 1 Register (q = 0 to 31)

Access: RSCANnRMDF1_q register can be read only in 32-bit units
RSCANnRMDF1_qL, RSCANnRMDF1_qH registers can be read only in 16-bit units
RSCANnRMDF1_qLL, RSCANnRMDF1_qLH, RSCANnRMDF1_qHL, RSCANnRMDF1_qHH registers can be read only in 8-bit units

Address: RSCANnRMDF1_q: <RSCFDn_base> + $060C_H$ + (10_H × q)
RSCANnRMDF1_qL: <RSCFDn_base> + $060C_H$ + (10_H × q),
RSCANnRMDF1_qH: <RSCFDn_base> + $060E_H$ + (10_H × q)
RSCANnRMDF1_qLL: <RSCFDn_base> + $060C_H$ + (10_H × q),
RSCANnRMDF1_qLH: <RSCFDn_base> + $060D_H$ + (10_H × q),
RSCANnRMDF1_qHL: <RSCFDn_base> + $060E_H$ + (10_H × q),
RSCANnRMDF1_qHH: <RSCFDn_base> + $060F_H$ + (10_H × q)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB7[7:0]								RMDB6[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB5[7:0]								RMDB4[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.38 RSCANnRMDF1_q Register Contents

Bit	Bit Name	Function
31 to 24	RMDB7[7:0]	Receive Buffer Data Byte 7
23 to 16	RMDB6[7:0]	Receive Buffer Data Byte 6
15 to 8	RMDB5[7:0]	Receive Buffer Data Byte 5
7 to 0	RMDB4[7:0]	Receive Buffer Data Byte 4 Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCANnRMPTRq register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

25.3.7 Details of Receive FIFO Buffer-related Registers

25.3.7.1 RSCANnRFCCx — Receive FIFO Buffer Configuration and Control Register (x = 0 to 7)

Access: RSCANnRFCCxregister can be read/written in 32-bit units
RSCANnRFCCxL, RSCANnRFCCxH registers can be read/written in 16-bit units
RSCANnRFCCxLL, RSCANnRFCCxLH, RSCANnRFCCxHL, RSCANnRFCCxHH registers can be read/written in 8-bit units

Address: RSCANnRFCCx: <RSCFDn_base> + $00B8_H$ + (04_H × x)
RSCANnRFCCxL: <RSCFDn_base> + $00B8_H$ + (04_H × x),
RSCANnRFCCxH: <RSCFDn_base> + $00BA_H$ + (04_H × x)
RSCANnRFCCxLL: <RSCFDn_base> + $00B8_H$ + (04_H × x),
RSCANnRFCCxLH: <RSCFDn_base> + $00B9_H$ + (04_H × x),
RSCANnRFCCxHL: <RSCFDn_base> + $00BA_H$ + (04_H × x),
RSCANnRFCCxHH: <RSCFDn_base> + $00BB_H$ + (04_H × x)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFIGCV[2:0]			RFIM	—	RFDC[2:0]			—	—	—	—	—	—	RFIE	RFE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Table 25.39 RSCANnRFCCx Register Contents

Bit	Bit Name	Function
31 to 16	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
15 to 13	RFIGCV[2:0]	Receive FIFO Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	RFIM	Receive FIFO Interrupt Source Select 0: An interrupt occurs when the condition set by the RFIGCV[2:0] bits is met. 1: An interrupt occurs each time a message has been received.
11	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
10 to 8	RFDC[2:0]	Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7 to 2	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
1	RFIE	Receive FIFO Interrupt Enable 0: Receive FIFO interrupt is disabled. 1: Receive FIFO interrupt is enabled.

Bit	Bit Name	Function
0	RFE	Receive FIFO Buffer Enable 0: No receive FIFO buffer is used. 1: Receive FIFO buffers are used.

RFIGCV[2:0] Bits

These bits are used to specify the number of received messages for generating a receive FIFO interrupt request when the RFIM bit is set to 0 with a fraction for the total number of buffers (the setting of RFDC[2:0]).
When the RFDC[2:0] bits are set to 001_B (4 messages), set the RFIGCV[2:0] bits to 001_B, 011_B, 101_B, or 111_B. Modify these bits only in global reset mode.

RFIM Bit

This bit is used to select a FIFO interrupt source. Modify this bit only in global reset mode.

RFDC[2:0] Bits

These bits are used to select the number of messages that can be stored in a single receive FIFO buffer. When these bits are set to 000B, no receive FIFO buffer should be used. Modify these bits only in global reset mode.

RFIE Bit

Setting the RFIE bit to 1 enables receive FIFO interrupts. Modify this bit when the RFE bit set to 0 (no receive FIFO buffer is used).

RFE Bit

Setting the RFE bit to 1 makes receive FIFO buffers available. Clearing this bit to 0 sets the RFEMP flag in the RSCFDnRFSTSxRSCANnRFSTSx register to 1 (buffer empty). Modify this bit in global operating mode or global test mode.
Set this bit to 1 with another instruction after the settings to all bits in the RSCANnRFCCx register have been done.
This bit is cleared to 0 in global reset mode.

25.3.7.2 RSCANnRFSTSx — Receive FIFO Buffer Status Register (x = 0 to 7)

Access: RSCANnRFSTSx register can be read/written in 32-bit units
RSCANnRFSTSxL, RSCANnRFSTSxH registers can be read/written in 16-bit units
RSCANnRFSTSxLL, RSCANnRFSTSxLH, RSCANnRFSTSxHL, RSCANnRFSTSxHH registers can be read/written in 8-bit units

Address: RSCANnRFSTSx: <RSCFDn_base> + $00D8_H$ + (04_H × x)
RSCANnRFSTSxL: <RSCFDn_base> + $00D8_H$ + (04_H × x),
RSCANnRFSTSxH: <RSCFDn_base> + $00DA_H$ + (04_H × x)
RSCANnRFSTSxLL: <RSCFDn_base> + $00D8_H$ + (04_H × x),
RSCANnRFSTSxLH: <RSCFDn_base> + $00D9_H$ + (04_H × x),
RSCANnRFSTSxHL: <RSCFDn_base> + $00DA_H$ + (04_H × x),
RSCANnRFSTSxHH: <RSCFDn_base> + $00DB_H$ + (04_H × x)

Value after reset: $0000\ 0001_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMC[7:0]								—	—	—	—	RFIF	RFMLT	RFFLL	RFEMP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.40 RSCANnRFSTSx Register Contents

Bit	Bit Name	Function
31 to 16	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
15 to 8	RFMC[7:0]	Receive FIFO Unread Message Counter The number of unread messages stored in the receive FIFO buffer is displayed.
7 to 4	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
3	RFIF	Receive FIFO Interrupt Request Flag 0: No receive FIFO interrupt request is present. 1: A receive FIFO interrupt request is present.
2	RFMLT	Receive FIFO Message Lost Flag 0: No receive FIFO message is lost. 1: A receive FIFO message is lost.
1	RFFLL	Receive FIFO Buffer Full Status Flag 0: The receive FIFO buffer is not full. 1: The receive FIFO buffer is full.
0	RFEMP	Receive FIFO Buffer Empty Status Flag 0: The receive FIFO buffer contains unread message. 1: The receive FIFO buffer contains no unread message (buffer empty).

RFMC[7:0] Flag

This flag indicates the number of unread messages in the receive FIFO buffer. This flag becomes 00_H when the RFE bit in the RSCANnRFCCx register is set to 0.

RFIF Flag

This flag is set to 1 when the receive FIFO interrupt request generation conditions set by the RFIGCV[2:0] bits and the RFIM bit in the RSCANnRFCCx register are met. This flag is cleared to 0 in global reset mode or by writing 0 to this flag. Modify this bit in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0,

using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

RFMLT Flag

This flag is set to 1 when an attempt is made to store a new message while the receive FIFO buffer is full. In this case, the new message is discarded.
This flag is cleared to 0 in global reset mode or by writing 0 to this flag.
Modify this bit in global operating mode or global test mode.
To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

RFFLL Flag

This flag is set to 1 when the number of messages stored in the receive FIFO buffer matches the FIFO buffer depth set by the RFDC[2:0] bits in the RSCANnRFCCx register.
If the number of messages stored in the receive FIFO buffer becomes smaller than the FIFO buffer depth set by the RFDC[2:0] bits, this flag is cleared to 0. This flag is also cleared to 0 when the RFE bit in the RSCANnRFCCx register is set to 0 (no receive FIFO buffer is used) or in global reset mode.

RFEMP Flag

This flag is set to 1 when all messages in the receive FIFO buffer have been read. This flag is also set to 1 when the RFE bit in the RSCANnRFCCx register is 0 or in global reset mode.
This flag is cleared to 0 when even a single received message has been stored in the receive FIFO buffer.

Note: To clear the RFMLT or RFIF flag to 0, use a store instruction to write “0” to the given flag and “1” to the other flags.

25.3.7.3 RSCANnRFPCTRx — Receive FIFO Buffer Pointer Control Register (x = 0 to 7)

Access: RSCANnRFPCTRx register can only be written in 32-bit units
RSCANnRFPCTRxL, RSCANnRFPCTRxH registers can only be written in 16-bit units
RSCANnRFPCTRxLL, RSCANnRFPCTRxLH, RSCANnRFPCTRxHL, RSCANnRFPCTRxHH registers can only be written in 8-bit units

Address: RSCANnRFPCTRx: <RSCFDn_base> + $00F8_H$ + (04_H × x)
RSCANnRFPCTRxL: <RSCFDn_base> + $00F8_H$ + (04_H × x),
RSCANnRFPCTRxH: <RSCFDn_base> + $00FA_H$ + (04_H × x)
RSCANnRFPCTRxLL: <RSCFDn_base> + $00F8_H$ + (04_H × x),
RSCANnRFPCTRxLH: <RSCFDn_base> + $00F9_H$ + (04_H × x),
RSCANnRFPCTRxHL: <RSCFDn_base> + $00FA_H$ + (04_H × x),
RSCANnRFPCTRxHH: <RSCFDn_base> + $00FB_H$ + (04_H × x)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RFPC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 25.41 RSCANnRFPCTRx Register Contents

Bit	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	RFPC[7:0]	Receive FIFO Pointer Control When these bits are set to FF_H, the read pointer moves to the next unread message in the receive FIFO buffer.

RFPC[7:0] Bits

When the RFPC[7:0] bits are set to FF_H, the read pointer moves to the next unread message in the receive FIFO buffer. At this time, the RFMC[7:0] (receive FIFO unread message counter) value in the RSCANnRFSTSx register is decremented. Read the RSCANnRFIDx, RSCANnRFPTRx, RSCANnRFDF0_x, and RSCANnRFDF1_x registers to read messages in the receive FIFO buffer, and then write FF_H to the RFPC[7:0] bits.
When writing FF_H to these bits, make sure that the RFE bit in the RSCANnRFCCx register is set to 1 (receive FIFO buffers are used) and the RFEMP flag in the RSCANnRFSTSx register is 0 (the receive FIFO buffer contains unread messages).

25.3.7.4 RSCANnRFIDx — Receive FIFO Buffer Access ID Register (x = 0 to 7)

Access: RSCANnRFIDx register can be read only in 32-bit units
RSCANnRFIDxL, RSCANnRFIDxH registers can be read only in 16-bit units
RSCANnRFIDxLL, RSCANnRFIDxLH, RSCANnRFIDxHL, RSCANnRFIDxHH registers can be read only in 8-bit units

Address: RSCANnRFIDx: <RSCFDn_base> + $0E00_H$ + (10_H × x)
RSCANnRFIDxL: <RSCFDn_base> + $0E00_H$ + (10_H × x),
RSCANnRFIDxH: <RSCFDn_base> + $0E02_H$ + (10_H × x)
RSCANnRFIDxLL: <RSCFDn_base> + $0E00_H$ + (10_H × x),
RSCANnRFIDxLH: <RSCFDn_base> + $0E01_H$ + (10_H × x),
RSCANnRFIDxHL: <RSCFDn_base> + $0E02_H$ + (10_H × x),
RSCANnRFIDxHH: <RSCFDn_base> + $0E03_H$ + (10_H × x)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFIDE	RFRTR	—	RFID[28:16]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFID[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.42 RSCANnRFIDx Register Contents

Bit	Bit Name	Function
31	RFIDE	Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	RFRTR	Receive FIFO Buffer RTR 0: Data frame 1: Remote frame
29	Reserved	This bit is read as the value after reset.
28 to 0	RFID[28:0]	Receive FIFO Buffer ID Data The standard ID or extended ID of received message can be read. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive FIFO buffer.

RFRTR Bit

This bit indicates the frame format (data frame or remote frame) of the message stored in the receive FIFO buffer.

RFID[28:0] Bits

These bits indicate the ID of the message stored in the receive FIFO buffer.

25.3.7.5 RSCANnRFPTRx — Receive FIFO Buffer Access Pointer Register (x = 0 to 7)

Access: RSCANnRFPTRx register can be read only in 32-bit units
RSCANnRFPTRxL, RSCANnRFPTRxH registers can be read only in 16-bit units
RSCANnRFPTRxLL, RSCANnRFPTRxLH, RSCANnRFPTRxHL, RSCANnRFPTRxHH registers can be read only in 8-bit units

Address: RSCANnRFPTRx: <RSCFDn_base> + $0E04_H$ + (10_H × x)
RSCANnRFPTRxL: <RSCFDn_base> + $0E04_H$ + (10_H × x),
RSCANnRFPTRxH: <RSCFDn_base> + $0E06_H$ + (10_H × x)
RSCANnRFPTRxLL: <RSCFDn_base> + $0E04_H$ + (10_H × x),
RSCANnRFPTRxLH: <RSCFDn_base> + $0E05_H$ + (10_H × x),
RSCANnRFPTRxHL: <RSCFDn_base> + $0E06_H$ + (10_H × x),
RSCANnRFPTRxHH: <RSCFDn_base> + $0E07_H$ + (10_H × x)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDLC[3:0]				RFPTR[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFTS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.43 RSCANnRFPTRx Register Contents

Bit	Bit Name	Function
31 to 28	RFDLC[3:0]	Receive FIFO Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	RFPTR[11:0]	Receive FIFO Buffer Label Data Label information of the received message can be read.
15 to 0	RFTS[15:0]	Receive FIFO Buffer Timestamp Data Timestamp value of the received message can be read.

RFDLC[3:0] Bits

These bits contain the data length of the message stored in the receive FIFO buffer.

RFPTR[11:0] Bits

These bits contain the label information of the message stored in the receive FIFO buffer.

RFTS[15:0] Bits

These bits contain the timestamp value of the message stored in the receive FIFO buffer.

25.3.7.6 RSCANnRFDF0_x — Receive FIFO Buffer Access Data Field 0 Register (x = 0 to 7)

Access: RSCANnRFDF0_x register can be read only in 32-bit units
RSCANnRFDF0_xL, RSCANnRFDF0_xH registers can be read only in 16-bit units
RSCANnRFDF0_xLL, RSCANnRFDF0_xLH, RSCANnRFDF0_xHL, RSCANnRFDF0_xHH registers can be read only in 8-bit units

Address: RSCANnRFDF0_x: <RSCFDn_base> + $0E08_H$ + (10_H × x)
RSCANnRFDF0_xL: <RSCFDn_base> + $0E08_H$ + (10_H × x),
RSCANnRFDF0_xH: <RSCFDn_base> + $0E0A_H$ + (10_H × x)
RSCANnRFDF0_xLL: <RSCFDn_base> + $0E08_H$ + (10_H × x),
RSCANnRFDF0_xLH: <RSCFDn_base> + $0E09_H$ + (10_H × x),
RSCANnRFDF0_xHL: <RSCFDn_base> + $0E0A_H$ + (10_H × x),
RSCANnRFDF0_xHH: <RSCFDn_base> + $0E0B_H$ + (10_H × x)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB3[7:0]								RFDB2[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB1[7:0]								RFDB0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.44 RSCANnRFDF0_x Register Contents

Bit	Bit Name	Function
31 to 24	RFDB3[7:0]	Receive FIFO Buffer Data Byte 3
23 to 16	RFDB2[7:0]	Receive FIFO Buffer Data Byte 2
15 to 8	RFDB1[7:0]	Receive FIFO Buffer Data Byte 1
7 to 0	RFDB0[7:0]	Receive FIFO Buffer Data Byte 0 Data for a message stored in the receive FIFO buffer can be read.

When the RFDLC[3:0] value in the RSCANnRFPTRx register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

25.3.7.7 RSCANnRFDF1_x — Receive FIFO Buffer Access Data Field 1 Register (x = 0 to 7)

Access: RSCANnRFDF1_x register can be read only in 32-bit units
RSCANnRFDF1_xL, RSCANnRFDF1_xH registers can be read only in 16-bit units
RSCANnRFDF1_xLL, RSCANnRFDF1_xLH, RSCANnRFDF1_xHL, RSCANnRFDF1_xHH registers can be read only in 8-bit units

Address: RSCANnRFDF1_x: <RSCFDn_base> + $0E0C_H$ + (10_H × x)
RSCANnRFDF1_xL: <RSCFDn_base> + $0E0C_H$ + (10_H × x),
RSCANnRFDF1_xH: <RSCFDn_base> + $0E0E_H$ + (10_H × x)
RSCANnRFDF1_xLL: <RSCFDn_base> + $0E0C_H$ + (10_H × x),
RSCANnRFDF1_xLH: <RSCFDn_base> + $0E0D_H$ + (10_H × x),
RSCANnRFDF1_xHL: <RSCFDn_base> + $0E0E_H$ + (10_H× x),
RSCANnRFDF1_xHH: <RSCFDn_base> + $0E0F_H$ + (10_H × x)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB7[7:0]								RFDB6[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB5[7:0]								RFDB4[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.45 RSCANnRFDF1_x Register Contents

Bit	Bit Name	Function
31 to 24	RFDB7[7:0]	Receive FIFO Buffer Data Byte 7
23 to 16	RFDB6[7:0]	Receive FIFO Buffer Data Byte 6
15 to 8	RFDB5[7:0]	Receive FIFO Buffer Data Byte 5
7 to 0	RFDB4[7:0]	Receive FIFO Buffer Data Byte 4 Data for a message stored in the receive FIFO buffer can be read.

When the RFDLC[3:0] value in the RSCANnRFPTRx register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

25.3.8 Details of Transmit/Receive FIFO Buffer-related Registers

25.3.8.1 RSCANnCFCCk — Transmit/receive FIFO Buffer Configuration and Control Register k (k = 0 to 5)

Access: RSCANnCFCCk register can be read/written in 32-bit units
RSCANnCFCCkL, RSCANnCFCCkH registers can be read/written in 16-bit units
RSCANnCFCCkLL, RSCANnCFCCkLH, RSCANnCFCCkHL, RSCANnCFCCkHH registers can be read/written in 8-bit units

Address: RSCANnCFCCk: <RSCFDn_base> + 0118_H + (04_H × k)
RSCANnCFCCkL: <RSCFDn_base> + 0118_H + (04_H × k),
RSCANnCFCCkH: <RSCFDn_base> + $011A_H$ + (04_H × k)
RSCANnCFCCkLL: <RSCFDn_base> + 0118_H + (04_H × k),
RSCANnCFCCkLH: <RSCFDn_base> + 0119_H + (04_H × k),
RSCANnCFCCkHL: <RSCFDn_base> + $011A_H$ + (04_H × k),
RSCANnCFCCkHH: <RSCFDn_base> + $011B_H$ + (04_H × k)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFITT[7:0]								CFTML[3:0]				CFITR	CFITSS	CFM[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFIGCV[2:0]			CFIM	—	CFDC[2:0]			—	—	—	—	—	CFTXIE	CFRXIE	CFE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.46 RSCANnCFCCk Register Contents

Bit	Bit Name	Function
31 to 24	CFITT[7:0]	Set a message transmission interval. Set Value: 00_H to FF_H
23 to 20	CFTML[3:0]	Transmit Buffer Link Configuration Set the transmit buffer number to be linked to the transmit/receive FIFO buffer.
19	CFITR	Transmit/Receive FIFO Interval Timer Resolution 0: Clock dividing pclk by (ITRCP[15:0] bits) 1: Clock dividing pclk by (ITRCP[15:0] bits × 10)
18	CFITSS	Transmit/Receive FIFO Interval Timer Clock Source Select 0: Interval timer clock source selected by the CFITR bit 1: Interval timer clock source is the bit time clock for the channel to which the FIFO is linked.
17 to 16	CFM[1:0]	Transmit/Receive FIFO Mode Select b17 b16 0 0: Receive mode 0 1: Transmit mode 1 0: Gateway mode 1 1: Setting prohibited
15 to 13	CFIGCV[2:0]	Transmit/Receive FIFO Receive Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.

Bit	Bit Name	Function
12	CFIM	Transmit/Receive FIFO Interrupt Source Select 0: • Receive mode/gateway mode When the number of received messages has met the condition set by the CFIGCV[2:0] bits, a FIFO receive interrupt request is generated. • Transmit mode/gateway mode When the buffer becomes empty upon completion of message transmission, a FIFO transmit interrupt request is generated. 1: • Receive mode/gateway mode A FIFO receive interrupt request is generated each time a message has been received. • Transmit mode/gateway mode A FIFO transmit interrupt request is generated each time a message has been transmitted.
11	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
10 to 8	CFDC[2:0]	Transmit/Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7 to 3	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
2	CFTXIE	Transmit/Receive FIFO Transmit Interrupt Enable 0: Transmit/receive FIFO transmit interrupt is disabled. 1: Transmit/receive FIFO transmit interrupt is enabled.
1	CFRXIE	Transmit/Receive FIFO Receive Interrupt Enable 0: Transmit/receive FIFO receive interrupt is disabled. 1: Transmit/receive FIFO receive interrupt is enabled.
0	CFE	Transmit/Receive FIFO Buffer Enable 0: No transmit/receive FIFO buffer is used. 1: Transmit/receive FIFO buffers are used.

CFITT[7:0] Bits

These bits are used to set a message transmission interval when transmitting messages continuously from a transmit/receive FIFO buffer whose CFM[1:0] bits are set to 01_B (transmit mode) or 10_B (gateway mode).
Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITT[7:0] bits.

CFTML[3:0] Bits

These bits are used to set the number of transmit buffer on the channel which will be linked to transmit/receive FIFO buffer k when the CFM[1:0] bits are set to 01_B (transmit mode) or 10_B (gateway mode). There are three transmit/receive FIFO buffers per channel, so channel number m of FIFO buffer k is calculated as m = k/3 (integer division). The actual assigned transmit buffer number p linked to FIFO buffer k will be ((16 × m) + CFTML[3:0]).
See Table 25.11 and Table 25.12, as for the relationship between transmit/receive FIFO buffer k and transmit buffer p.
Setting the CFDC[2:0] bits to 001_B or more enables the setting of the CFTML[3:0] bits.
Do not link to any transmit buffer which is already allocated to a transmit queue on the identical channel or to another transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFITR Bit

This bit is enabled when the CFITSS bit is 0.
When this bit is 0, the interval timer clock source is the pclk/2 clock divided by the value of the ITRCP[15:0] bits in the RSCANnGCFG register.
When this bit is 1, the interval timer clock source is the pclk/2 clock divided by (the value of the ITRCP[15:0] bits in the RSCANnGCFG register × 10).

Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFITSS Bit

When this bit is 0, the clock selected by the CFITR bit is the count source of the interval timer.
When this bit is 1, the bit time clock of the channel to which the FIFO is linked is the count source of the interval timer.
Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFM[1:0] Bits

These bits are used to select transmit/receive FIFO mode. Modify these bits only in global reset mode.

CFIGCV[2:0] Bits

These bits are used to specify the number of received messages for generating a transmit/receive FIFO receive interrupt request when the CFM[1:0] bits are set to 00_B (receive mode) or 10B (gateway mode) and the CFIM bit is set to 0 with a fraction for the total number of buffers (the setting of CFDC[2:0]).
When the CFDC[2:0] bits are set to 001_B (4 messages), set the CFIGCV[2:0] bits to 001_B, 011_B, 101_B, or 111_B.
Modify these bits only in global reset mode.

CFIM Bit

This bit is used to select a transmit/receive FIFO interrupt source. Modify this bit only in global reset mode.

CFDC[2:0] Bits

These bits are used to set the number of messages that can be stored in a single transmit/receive FIFO buffer. When these bits are set to 000_B, do not use a transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFTXIE Bit

When this bit is set to 1 and the CFTXIF flag in the RSCANnCFSTSk register is set to 1, a transmit/receive FIFO transmit interrupt request is generated.
Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used).

CFRXIE Bit

When this bit is set to 1 and the CFRXIF flag in the RSCANnCFSTSk register is set to 1, a transmit/receive FIFO receive interrupt request is generated.
Modify this bit with the CFE bit set to 0.

CFE Bit

Setting this bit to 1 makes transmit/receive FIFO buffers available.
When this bit is set to 0 in transmit mode or gateway mode, if a message in the transmit/receive FIFO buffer is being transmitted or will be transmitted next, the transmit/receive FIFO buffer becomes empty after completion of transmission of that message, or upon detection of a CAN bus error, or arbitration-lost. In other cases or in receive mode, the transmit/receive FIFO buffer becomes empty immediately.
This bit is cleared to 0 when the following conditions are met.

- Receive mode: Global reset mode
- Transmit mode or gateway mode: Channel reset mode

Modify this bit in the following mode.

- Receive mode: Global operating mode or global test mode
- Transmit mode or gateway mode: Channel communication mode or channel halt mode

After all other bits in the RSCANnCFCCk register have been set, set this bit to 1 by using another instruction.

25.3.8.2 RSCANnCFSTSk — Transmit/receive FIFO Buffer Status Register (k = 0 to 5)

Access: RSCANnCFSTSk register can be read/written in 32-bit units
RSCANnCFSTSkL, RSCANnCFSTSkH registers can be read/written in 16-bit units
RSCANnCFSTSkLL, RSCANnCFSTSkLH, RSCANnCFSTSkHL, RSCANnCFSTSkHH registers can be read/written in 8-bit units

Address: RSCANnCFSTSk: <RSCFDn_base> + 0178_H + (04_H × k)
RSCANnCFSTSkL: <RSCFDn_base> + 0178_H + (04_H × k),
RSCANnCFSTSkH: <RSCFDn_base> + $017A_H$ + (04_H × k)
RSCANnCFSTSkLL: <RSCFDn_base> + 0178_H + (04_H × k),
RSCANnCFSTSkLH: <RSCFDn_base> + 0179_H + (04_H × k),
RSCANnCFSTSkHL: <RSCFDn_base> + $017A_H$ + (04_H × k),
RSCANnCFSTSkHH: <RSCFDn_base> + $017B_H$ + (04_H × k)

Value after reset: $0000\ 0001_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFMC[7:0]								—	—	—	CFTXIF	CFRXIF	CFMLT	CFFLL	CFEMP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.47 RSCANnCFSTSk Register Contents

Bit	Bit Name	Function
31 to 16	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
15 to 8	CFMC[7:0]	Transmit/Receive FIFO Message Counter The number of messages stored in the transmit/receive FIFO buffer.
7 to 5	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
4	CFTXIF	Transmit/Receive FIFO Transmit Interrupt Request Flag 0: No transmit/receive FIFO transmit interrupt request is present. 1: A transmit/receive FIFO transmit interrupt request is present.
3	CFRXIF	Transmit/Receive FIFO Receive Interrupt Request Flag 0: No transmit/receive FIFO receive interrupt request is present. 1: A transmit/receive FIFO receive interrupt request is present.
2	CFMLT	Transmit/Receive FIFO Message Lost Flag 0: No transmit/receive FIFO message is lost. 1: A transmit/receive FIFO message is lost.
1	CFFLL	Transmit/Receive FIFO Buffer Full Status Flag 0: The transmit/receive FIFO buffer is not full. 1: The transmit/receive FIFO buffer is full.
0	CFEMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: The transmit/receive FIFO buffer contains messages. 1: The transmit/receive FIFO buffer contains no message (buffer empty).

CFMC[7:0] Bits

The CFMC[7:0] bits indicate the following values that depend on the setting of the CFM[1:0] bits in the RSCANnCFCCk register.

- When CFM[1:0] value is 01_B (transmit mode): Number of untransmitted messages in the buffer
- When CFM[1:0] value is 00_B (receive mode): Number of unread received messages in the buffer
- When CFM[1:0] value is 10_B (gateway mode): Number of untransmitted received messages in the buffer

These bits are cleared to 0 when any of the following conditions is met.

- When CFM[1:0] value is 00_B: In global reset mode
- When CFM[1:0] value is 01_B or 10_B: In channel reset mode
- When the CFE bit in the RSCANnCFCCk register is cleared to 0.

CFTXIF Flag

The CFTXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 01_B or 10_B, and the factor selected by the CFIM bit in the RSCANnCFCCk register occurs

The CFTXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFTXIF flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.
To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFRXIF Flag

The CFRXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B or 10_B, and the factor selected by the CFIM bit in the RSCANnCFCCk register occurs

The CFRXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFRXIF flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.
To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.
When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFMLT Flag

The CFMLT flag is set to 1 when any of the following conditions is met.

- When an attempt is made to store a new message while the transmit/receive FIFO buffer is full. In this case, the new message is discarded.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFMLT flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode
To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFFLL Flag

The CFFLL flag is set to 1 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer matches the FIFO buffer depth set by the CFDC[2:0] bits in the RSCANnCFCCk register.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer becomes smaller than the FIFO buffer depth set by the CFDC[2:0] bits.
- When the CFE bit in the RSCANnCFCCk register is 0 (no transmit/receive FIFO buffer is used): When not in the transmit abort
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

CFEMP Flag

The CFEMP flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B: All messages have been read, or in global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: All messages have been transmitted, or in channel reset mode
- When the CFE bit is 0 (no transmit/receive FIFO buffer is used): Not in the transmit abort

The CFEMP flag is cleared to 0 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B or 10_B: At least one received message has been stored in the transmit/receive FIFO buffer.
- When the CFM[1:0] bits are set to 01_B: A value of FFH has been written to the RSCANnCFPCTRk register after data was written to the RSCANnCFIDk, RSCANnCFPTRk, RSCANnCFDF0_k, and RSCANnCFDF1_k registers.

Note: To clear CFTXIF, CFRXIF, or CFMLT flag to 0, the program must write 0. When writing, use a store instruction to write “0” to the given flag and “1” to other flags.

25.3.8.3 RSCANnCFPCTRk — Transmit/receive FIFO Buffer Pointer Control Register (k = 0 to 5)

Access: RSCANnCFPCTRk register can only be written in 32-bit units
RSCANnCFPCTRkL, RSCANnCFPCTRkH registers can only be written in 16-bit units
RSCANnCFPCTRkLL, RSCANnCFPCTRkLH, RSCANnCFPCTRkHL, RSCANnCFPCTRkHH registers can only be written in 8-bit units

Address: RSCANnCFPCTRk: <RSCFDn_base> + $01D8_H$ + (04_H × k)
RSCANnCFPCTRkL: <RSCFDn_base> + $01D8_H$ + (04_H × k),
RSCANnCFPCTRkH: <RSCFDn_base> + $01DA_H$ + (04_H × k)
RSCANnCFPCTRkLL: <RSCFDn_base> + $01D8_H$ + (04_H × k),
RSCANnCFPCTRkLH: <RSCFDn_base> + $01D9_H$ + (04_H × k),
RSCANnCFPCTRkHL: <RSCFDn_base> + $01DA_H$ + (04_H × k),
RSCANnCFPCTRkHH: <RSCFDn_base> + $01DB_H$ + (04_H × k)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CFPC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 25.48 RSCANnCFPCTRk Register Contents

Bit	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	CFPC[7:0]	Transmit/Receive FIFO Pointer Control • Receive mode: Writing FFH to these bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. • Transmit mode: Writing FFH to these bits moves the write pointer to the next stage of the transmit/receive FIFO buffer. • Gateway mode: Setting prohibited

CFPC[7:0] Bits

- Receive mode (CFM[1:0] value in the RSCANnCFCCk register is 00B):
  Writing FFH to the CFPC[7:0] bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value (transmit/receive FIFO message counter) in the RSCANnCFSTSk register is decremented. Read the RSCANnCFIDk, RSCANnCFPTRk, RSCANnCFDF0_k, and RSCANnCFDF1_k registers to read messages from the transmit/receive FIFO buffer, and then write FFH to the CFPC[7:0] bits.
  When writing FFH to these bits, make sure that the CFE bit in the RSCANnCFCCk register is set to 1 (transmit/receive FIFO buffers are used) and the CFEMP flag in the RSCANnCFSTSk register is 0 (the transmit/receive FIFO buffer contains messages).
- Transmit mode (CFM[1:0] value in the RSCANnCFCCk register is 01B):
  Writing FFH to the CFPC[7:0] bits stores the data written to the RSCANnCFIDk, RSCANnCFPTRk, RSCANnCFDF0_k, and RSCANnCFDF1_k registers in the transmit/receive FIFO buffer and moves the write pointer to the next stage of the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value is incremented. Write transmit messages to the RSCANnCFIDk, RSCANnCFPTRk, RSCANnCFDF0_k, and RSCANnCFDF1_k registers before writing FFH to the CFPC[7:0] bits.
  When writing FFH to these bits, make sure that the CFE bit in the RSCANnCFCCk register is set to 1 and the

CFFLL flag in the RSCANnCFSTSk register is 0 (the transmit/receive FIFO buffer is not full).

- Gateway mode (CFM[1:0] value in the RSCANnCFCCk register is 10_B):
  Setting prohibited

25.3.8.4 RSCANnCFIDk — Transmit/receive FIFO Buffer Access ID Register (k = 0 to 5)

Access: RSCANnCFIDk register can be read/written in 32-bit units
RSCANnCFIDkL, RSCANnCFIDkH registers can be read/written in 16-bit units
RSCANnCFIDkLL, RSCANnCFIDkLH, RSCANnCFIDkHL, RSCANnCFIDkHH registers can be read/written in 8-bit units

Address: RSCANnCFIDk: <RSCFDn_base> + $0E80_H$ + (10_H × k)
RSCANnCFIDkL: <RSCFDn_base> + $0E80_H$ + (10_H × k),
RSCANnCFIDkH: <RSCFDn_base> + $0E82_H$ + (10_H × k)
RSCANnCFIDkLL: <RSCFDn_base> + $0E80_H$ + (10_H × k),
RSCANnCFIDkLH: <RSCFDn_base> + $0E81_H$ + (10_H × k),
RSCANnCFIDkHL: <RSCFDn_base> + $0E82_H$ + (10_H × k),
RSCANnCFIDkHH: <RSCFDn_base> + $0E83_H$ + (10_H × k)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFIDE	CFRTR	THLEN	CFID[28:16]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFID[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.49 RSCANnCFIDk Register Contents

Bit	Bit Name	Function
31	CFIDE	Transmit/Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	CFRTR	Transmit/Receive FIFO Buffer RTR 0: Data frame 1: Remote frame
29	THLEN	Transmit History Data Store Enable This bit is valid only when the CFM[1:0] value is 01_B (transmit mode). 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	CFID[28:0]	Transmit/Receive FIFO Buffer ID Data • When CFM[1:0] value is 01_B (transmit mode): Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11. • When CFM[1:0] value is 00_B (receive mode): Standard ID or extended ID in the received message can be read. For standard ID, read bits 10 to 0. Bits 28 to 11 are read as 0.

This register is writable only when the CFM[1:0] value in the RSCANnCFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). This RSCFDnCFIDk register should not be read or written when the CFM[1:0] value is 10B (gateway mode).

CFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, these bits are used to set the ID format of the message to be transmitted from the transmit/receive FIFO buffer.

CFRTR Bit

This bit indicates the data format (data frame or remote frame) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, this bit is used to set the data format of the message to be transmitted from the transmit/receive FIFO buffer.

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, and buffer type) of transmit messages is stored in the transmit history buffer after transmission is completed.
This bit is enabled when the CFM[1:0] value is 01_B (transmit mode).

CFID[28:0] Bits

These bits contain the ID of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B.
When the CFM[1:0] value is 01_B, this bit is used to set the ID of the message to be transmitted from the transmit/receive FIFO buffer.

25.3.8.5 RSCANnCFPTRk — Transmit/receive FIFO Buffer Access Pointer Register (k = 0 to 5)

Access: RSCANnCFPTRk register can be read/written in 32-bit units
RSCANnCFPTRkL, RSCANnCFPTRkH registers can be read/written in 16-bit units
RSCANnCFPTRkLL, RSCANnCFPTRkLH, RSCANnCFPTRkHL, RSCANnCFPTRkHH registers can be read/written in 8-bit units

Address: RSCANnCFPTRk: <RSCFDn_base> + $0E84_H$ + (10_H × k)
RSCANnCFPTRkL: <RSCFDn_base> + $0E84_H$ + (10_H × k),
RSCANnCFPTRkH: <RSCFDn_base> + $0E86_H$ + (10_H × k)
RSCANnCFPTRkLL: <RSCFDn_base> + $0E84_H$ + (10_H × k),
RSCANnCFPTRkLH: <RSCFDn_base> + $0E85_H$ + (10_H × k),
RSCANnCFPTRkHL: <RSCFDn_base> + $0E86_H$ + (10_H × k),
RSCANnCFPTRkHH: <RSCFDn_base> + $0E87_H$ + (10_H × k)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDLC[3:0]				CFPTR[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFTS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.50 RSCANnCFPTRk Register Contents

Bit	Bit Name	Function
31 to 28	CFDLC[3:0]	Transmit/Receive FIFO Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	CFPTR[11:0]	Transmit/Receive FIFO Buffer Label Data • When CFM[1:0] value is 01_B (transmit mode): Set the label information to be stored in the transmit history buffer. Only bits CFPTR[7:0] are valid. • When CFM[1:0] value is 00_B (receive mode): The label information of the received message can be read.
15 to 0	CFTS[15:0]	Transmit/Receive FIFO Buffer Timestamp Data These bits are valid only when the CFM[1:0] value is 00_B (receive mode). The timestamp value of the received message can be read.

This register is writable only when the CFM[1:0] value in the RSCANnCFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

CFDLC[3:0] Bits

These bits indicate the data length of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, these bits are used to set the data length of the message to be transmitted from the transmit/receive FIFO buffer. If the data length is set to 1001_B or more, the actual transmit data defaults to 8 bytes.

CFPTR[11:0] Bits

These bits indicate the label information attached to the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, the CFPTR[7:0] value is stored in the transmit history buffer when message transmission has been completed.

CFTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the transmit/receive FIFO buffer.
These bits are valid when the CFM[1:0] value is 00_B.

25.3.8.6 RSCANnCFDF0_k — Transmit/receive FIFO Buffer Access Data Field 0 Register (k = 0 to 5)

Access: RSCANnCFDF0_k register can be read/written in 32-bit units
RSCANnCFDF0_kL, RSCANnCFDF0_kH registers can be read/written in 16-bit units
RSCANnCFDF0_kLL, RSCANnCFDF0_kLH, RSCANnCFDF0_kHL, RSCANnCFDF0_kHH registers can be read/written in 8-bit units

Address: RSCANnCFDF0_k: <RSCFDn_base> + $0E88_H$ + (10_H × k)
RSCANnCFDF0_kL: <RSCFDn_base> + $0E88_H$ + (10_H × k),
RSCANnCFDF0_kH: <RSCFDn_base> + $0E8A_H$ + (10_H × k)
RSCANnCFDF0_kLL: <RSCFDn_base> + $0E88_H$ + (10_H × k),
RSCANnCFDF0_kLH: <RSCFDn_base> + $0E89_H$ + (10_H × k),
RSCANnCFDF0_kHL: <RSCFDn_base> + $0E8A_H$ + (10_H × k),
RSCANnCFDF0_kHH: <RSCFDn_base> + $0E8B_H$ + (10_H × k)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB3[7:0]								CFDB2[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB1[7:0]								CFDB0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.51 RSCANnCFDF0_k Register Contents

Bit	Bit Name	Function
31 to 24	CFDB3[7:0]	Transmit/Receive FIFO Buffer Data Byte 3
23 to 16	CFDB2[7:0]	Transmit/Receive FIFO Buffer Data Byte 2
15 to 8	CFDB1[7:0]	Transmit/Receive FIFO Buffer Data Byte 1
7 to 0	CFDB0[7:0]	Transmit/Receive FIFO Buffer Data Byte 0 • When CFM[1:0] value is 01B (transmit mode): Set the transmit/receive FIFO buffer data. • When CFM[1:0] value is 00B (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.

This register is writable only when the CFM[1:0] value in the RSCANnCFCCk register is 01_B (transmit mode).
This register is readable only when the CFM[1:0] value is 00_B (receive mode). When the CFDLC[3:0] value in the RSCANnCFPTRk register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.
This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

25.3.8.7 RSCANnCFDF1_k — Transmit/receive FIFO Buffer Access Data Field 1 Register (k = 0 to 5)

Access: RSCANnCFDF1_k register can be read/written in 32-bit units
RSCANnCFDF1_kL, RSCANnCFDF1_kH registers can be read/written in 16-bit units
RSCANnCFDF1_kLL, RSCANnCFDF1_kLH, RSCANnCFDF1_kHL, RSCANnCFDF1_kHH registers can be read/written in 8-bit units

Address: RSCANnCFDF1_k: <RSCFDn_base> + $0E8C_H$ + (10_H × k)
RSCANnCFDF1_kL: <RSCFDn_base> + $0E8C_H$ + (10_H × k),
RSCANnCFDF1_kH: <RSCFDn_base> + $0E8E_H$ + (10_H × k)
RSCANnCFDF1_kLL: <RSCFDn_base> + $0E8C_H$ + (10_H × k),
RSCANnCFDF1_kLH: <RSCFDn_base> + $0E8D_H$ + (10_H × k),
RSCANnCFDF1_kHL: <RSCFDn_base> + $0E8E_H$ + (10_H × k),
RSCANnCFDF1_kHH: <RSCFDn_base> + $0E8F_H$ + (10_H × k)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB7[7:0]								CFDB6[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB5[7:0]								CFDB4[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.52 RSCANnCFDF1_k Register Contents

Bit	Bit Name	Function
31 to 24	CFDB7[7:0]	Transmit/Receive FIFO Buffer Data Byte 7
23 to 16	CFDB6[7:0]	Transmit/Receive FIFO Buffer Data Byte 6
15 to 8	CFDB5[7:0]	Transmit/Receive FIFO Buffer Data Byte 5
7 to 0	CFDB4[7:0]	Transmit/Receive FIFO Buffer Data Byte 4 • When CFM[1:0] value is 01B (transmit mode): Set the transmit/receive FIFO buffer data. • When CFM[1:0] value is 00B (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.

This register is writable only when the CFM[1:0] value in the RSCANnCFCCk register is 01_B (transmit mode).
This register is readable only when the CFM[1:0] value is 00_B (receive mode). When the CFDLC[3:0] value in the RSCANnCFPTRk register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.
This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

25.3.9 Details of FIFO Status-related Registers

25.3.9.1 RSCANnFESTS — FIFO Empty Status Register

Access: RSCANnFESTS register can be read only in 32-bit units
RSCANnFESTSL, RSCANnFESTSH registers can be read only in 16-bit units
RSCANnFESTSLL, RSCANnFESTSLH, RSCANnFESTSHL, RSCANnFESTSHH registers can be read only in 8-bit units

Address: RSCANnFESTS: <RSCFDn_base> + 0238_H
RSCANnFESTSL: <RSCFDn_base> + 0238_H, RSCANnFESTSH: <RSCFDn_base> + $023A_H$
RSCANnFESTSLL: <RSCFDn_base> + 0238_H, RSCANnFESTSLH: <RSCFDn_base> + 0239_H,
RSCANnFESTSHL: <RSCFDn_base> + $023A_H$, RSCANnFESTSHH: <RSCFDn_base> + $023B_H$

Value after reset: 03FF $FFFF_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CF5 EMP	CF4 EMP	CF3 EMP	CF2 EMP	CF1 EMP	CF0 EMP	RF7 EMP	RF6 EMP	RF5 EMP	RF4 EMP	RF3 EMP	RF2 EMP	RF1 EMP	RF0 EMP
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.53 RSCANnFESTS Register Contents

Bit	Bit Name	Function
31 to 14	Reserved	These bits are read as the value after reset.
13	CF5EMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: Transmit/receive FIFO buffer k contains a message. 1: Transmit/receive FIFO buffer k contains no message. (k = 0 to 5)
12	CF4EMP	
11	CF3EMP	
10	CF2EMP	
9	CF1EMP	
8	CF0EMP	
7	RF7EMP	Receive FIFO Buffer Empty Status Flag 0: Receive FIFO buffer x contains a unread messages. 1: Receive FIFO buffer x contains no unread message. (x = 0 to 7)
6	RF6EMP	
5	RF5EMP	
4	RF4EMP	
3	RF3EMP	
2	RF2EMP	
1	RF1EMP	
0	RF0EMP	

The RSCANnFESTS register is set to 03FF $FFFF_H$ in global reset mode.

CFkEMP Flag (k = 0 to 5)

The CFkEMP flag is set to 1 when the CFEMP flag in the RSCANnCFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)). When the CFEMP flag is cleared to 0 (the transmit/receive FIFO buffer contains messages), the CFkEMP flag is cleared to 0.

RFxEMP Flag (x = 0 to 7)

The RFxEMP flag is set to 1 when the RFEMP flag in the RSCANnRFSTSx register is set to 1 (the receive FIFO buffer

contains no unread message). When the RFEMP flag is cleared to 0 (the receive FIFO buffer contains unread messages), the RFxEMP flag is cleared to 0.

25.3.9.2 RSCANnFFSTS — FIFO Full Status Register

Access: RSCANnFFSTS register can be read only in 32-bit units
RSCANnFFSTSL, RSCANnFFSTSH registers can be read only in 16-bit units
RSCANnFFSTSLL, RSCANnFFSTSLH, RSCANnFFSTSHL, RSCANnFFSTSHH registers can be read only in 8-bit units

Address: RSCANnFFSTS: <RSCFDn_base> + $023C_H$
RSCANnFFSTSL: <RSCFDn_base> + $023C_H$, RSCANnFFSTSH: <RSCFDn_base> + $023E_H$
RSCANnFFSTSLL: <RSCFDn_base> + $023C_H$, RSCANnFFSTSLH: <RSCFDn_base> + $023D_H$,
RSCANnFFSTSHL: <RSCFDn_base> + $023E_H$, RSCANnFFSTSHH: <RSCFDn_base> + $023F_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CF5 FLL	CF4 FLL	CF3 FLL	CF2 FLL	CF1 FLL	CF0 FLL	RF7 FLL	RF6 FLL	RF5 FLL	RF4 FLL	RF3 FLL	RF2 FLL	RF1 FLL	RF0 FLL
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.54 RSCANnFFSTS Register Contents

Bit	Bit Name	Function
31 to 14	Reserved	These bits are read as the value after reset.
13	CF5FLL	Transmit/Receive FIFO Buffer Full Status Flag 0: Transmit/receive buffer k is not full. 1: Transmit/receive buffer k is full. (k = 0 to 5)
12	CF4FLL	
11	CF3FLL	
10	CF2FLL	
9	CF1FLL	
8	CF0FLL	
7	RF7FLL	Receive FIFO Buffer Full Status Flag 0: Receive FIFO buffer x is not full. 1: Receive FIFO buffer x is full. (x = 0 to 7)
6	RF6FLL	
5	RF5FLL	
4	RF4FLL	
3	RF3FLL	
2	RF2FLL	
1	RF1FLL	
0	RF0FLL	

The RSCANnFFSTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkFLL Flag (k = 0 to 5)

The CFkFLL flag is set to 1 when the CFFLL flag in the RSCANnCFSTSk register is set to 1 (the transmit/receive FIFO buffer is full).
When the CFFLL flag is cleared to 0 (the transmit/receive FIFO buffer is not full), the CFkFLL flag is cleared to 0.

RFxFLL Flag (x = 0 to 7)

The RFxFLL flag is set to 1 when the RFFLL flag in the RSCANnRFSTSx register is set to 1 (the receive FIFO buffer is full). When the RFFLL flag is cleared to 0 (the receive FIFO buffer is not full), the RFxFLL flag is cleared to 0.

25.3.9.3 RSCANnFMSTS — FIFO Message Lost Status Register

Access: RSCANnFMSTS register can be read only in 32-bit units
RSCANnFMSTSL, RSCANnFMSTSH registers can be read only in 16-bit units
RSCANnFMSTSLL, RSCANnFMSTSLH, RSCANnFMSTSHL, RSCANnFMSTSHH registers can be read only in 8-bit units

Address: RSCANnFMSTS: <RSCFDn_base> + 0240_H
RSCANnFMSTSL: <RSCFDn_base> + 0240_H, RSCANnFMSTSH: <RSCFDn_base> + 0242_H
RSCANnFMSTSLL: <RSCFDn_base> + 0240_H, RSCANnFMSTSLH: <RSCFDn_base> + 0241_H,
RSCANnFMSTSHL: <RSCFDn_base> + 0242_H, RSCANnFMSTSHH: <RSCFDn_base> + 0243_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CF5 MLT	CF4 MLT	CF3 MLT	CF2 MLT	CF1 MLT	CF0 MLT	RF7 MLT	RF6 MLT	RF5 MLT	RF4 MLT	RF3 MLT	RF2 MLT	RF1 MLT	RF0 MLT
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.55 RSCANnFMSTS Register Contents

Bit	Bit Name	Function
31 to 14	Reserved	These bits are read as the value after reset.
13	CF5MLT	Transmit/Receive FIFO Buffer Message Lost Status Flag 0: No transmit/receive FIFO buffer k message is lost. 1: A transmit/receive FIFO buffer k message is lost. (k = 0 to 5)
12	CF4MLT	
11	CF3MLT	
10	CF2MLT	
9	CF1MLT	
8	CF0MLT	
7	RF7MLT	Receive FIFO Buffer Message Lost Status Flag 0: No receive FIFO buffer x message is lost. 1: A receive FIFO buffer x message is lost. (x = 0 to 7)
6	RF6MLT	
5	RF5MLT	
4	RF4MLT	
3	RF3MLT	
2	RF2MLT	
1	RF1MLT	
0	RF0MLT	

The RSCANnFMSTS register is cleared to 0000 0000H in global reset mode.

CFkMLT Flag (k = 0 to 5)

The CFkMLT flag is set to 1 when the CFMLT flag in the RSCANnCFSTSk register is set to 1 (a transmit/receive FIFO message is lost). When the CFMLT flag is cleared to 0, the CFkMLT flag is cleared to 0.

RFxMLT Flag (x = 0 to 7)

The RFxMLT flag is set to 1 when the RFMLT flag in the RSCANnRFSTSx register is set to 1 (a receive FIFO message is lost). When the RFMLT flag is cleared to 0, the RFxMLT flag is cleared to 0.

25.3.9.4 RSCANnRFISTS — Receive FIFO Buffer Interrupt Flag Status Register

Access: RSCANnRFISTS register can be read only in 32-bit units
RSCANnRFISTSL, RSCANnRFISTSH registers can be read only in 16-bit units
RSCANnRFISTSLL, RSCANnRFISTSLH, RSCANnRFISTSHL, RSCANnRFISTSHH registers can be read only in 8-bit units

Address: RSCANnRFISTS: <RSCFDn_base> + 0244_H
RSCANnRFISTSL: <RSCFDn_base> + 0244_H, RSCANnRFISTSH: <RSCFDn_base> + 0246_H
RSCANnRFISTSLL: <RSCFDn_base> + 0244_H, RSCANnRFISTSLH: <RSCFDn_base> + 0245_H,
RSCANnRFISTSHL: <RSCFDn_base> + 0246_H, RSCANnRFISTSHH: <RSCFDn_base> + 0247_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RF7IF	RF6IF	RF5IF	RF4IF	RF3IF	RF2IF	RF1IF	RF0IF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.56 RSCANnRFISTS Register Contents

Bit	Bit Name	Function
31 to 8	Reserved	These bits are read as the value after reset.
7	RF7IF	Receive FIFO Buffer Interrupt Request Status Flag 0: No receive FIFO buffer x interrupt request is present. 1: A receive FIFO buffer x interrupt request is present. (x = 0 to 7)
6	RF6IF	
5	RF5IF	
4	RF4IF	
3	RF3IF	
2	RF2IF	
1	RF1IF	
0	RF0IF	

The RSCANnRFISTS register is cleared to $0000\ 0000_H$ in global reset mode.

RFxIF Flag (x = 0 to 7)

The RFxIF flag is set to 1 when the RFIF flag in the RSCANnRFSTSx register is set to 1 (a receive FIFO interrupt request is present). When the RFIF flag is cleared to 0, the RFxIF flag is cleared to 0.

25.3.9.5 RSCANnCFRISTS — Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register

Access: RSCANnCFRISTS register can be read only in 32-bit units
RSCANnCFRISTSL, RSCANnCFRISTSH registers can be read only in 16-bit units
RSCANnCFRISTSLL, RSCANnCFRISTSLH, RSCANnCFRISTSHL, RSCANnCFRISTSHH registers can be read only in 8-bit units

Address: RSCANnCFRISTS: <RSCFDn_base> + 0248_H
RSCANnCFRISTSL: <RSCFDn_base> + 0248_H, RSCANnCFRISTSH: <RSCFDn_base> + $024A_H$
RSCANnCFRISTSLL: <RSCFDn_base> + 0248_H, RSCANnCFRISTSLH: <RSCFDn_base> + 0249_H,
RSCANnCFRISTSHL: <RSCFDn_base> + $024A_H$, RSCANnCFRISTSHH: <RSCFDn_base> + $024B_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CF5 RXIF	CF4 RXIF	CF3 RXIF	CF2 RXIF	CF1 RXIF	CF0 RXIF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.57 RSCANnCFRISTS Register Contents

Bit	Bit Name	Function
31 to 6	Reserved	These bits are read as the value after reset.
5	CF5RXIF	Transmit/Receive FIFO Buffer Receive Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k receive interrupt request is present. 1: A transmit/receive FIFO buffer k receive interrupt request is present. (k = 0 to 5)
4	CF4RXIF	
3	CF3RXIF	
2	CF2RXIF	
1	CF1RXIF	
0	CF0RXIF	

The RSCANnCFRISTS register is cleared to 0000 0000H in global reset mode.

CFkRXIF Flag (k = 0 to 5)

The CFkRXIF flag is set to 1 when the CFRXIF flag in the RSCANnCFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). When the CFRXIF flag is cleared to 0, the CFkRXIF flag is cleared to 0.

25.3.9.6 RSCANnCFTISTS — Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register

Access: RSCANnCFTISTS register can be read only in 32-bit units
RSCANnCFTISTSL, RSCANnCFTISTSH registers can be read only in 16-bit units
RSCANnCFTISTSLL, RSCANnCFTISTSLH, RSCANnCFTISTSHL, RSCANnCFTISTSHH registers can be read only in 8-bit units

Address: RSCANnCFTISTS: <RSCFDn_base> + $024C_H$
RSCANnCFTISTSL: <RSCFDn_base> + $024C_H$, RSCANnCFTISTSH: <RSCFDn_base> + $024E_H$
RSCANnCFTISTSLL: <RSCFDn_base> + $024C_H$, RSCANnCFTISTSLH: <RSCFDn_base> + $024D_H$,
RSCANnCFTISTSHL: <RSCFDn_base> + $024E_H$, RSCANnCFTISTSHH: <RSCFDn_base> + $024F_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CF5 TXIF	CF4 TXIF	CF3 TXIF	CF2 TXIF	CF1 TXIF	CF0 TXIF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.58 RSCANnCFTISTS Register Contents

Bit	Bit Name	Function
31 to 6	Reserved	These bits are read as the value after reset.
5	CF5TXIF	Transmit/Receive FIFO Buffer Transmit Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k transmit interrupt request is present. 1: A transmit/receive FIFO buffer k transmit interrupt request is present. (k = 0 to 5)
4	CF4TXIF	
3	CF3TXIF	
2	CF2TXIF	
1	CF1TXIF	
0	CF0TXIF	

The RSCANnCFTISTS register is cleared to 0000 0000H in global reset mode.

CFkTXIF Flag (k = 0 to 5)

The CFkTXIF flag is set to 1 when the CFTXIF flag in the RSCANnCFSTSk register is set to 1 (a transmit/receive FIFO transmit interrupt request is present). When the CFTXIF flag is cleared to 0, the CFkTXIF flag is cleared to 0.

25.3.10 Details of Transmit Buffer-related Registers

25.3.10.1 RSCANnTMCp — Transmit Buffer Control Register (p = 0 to 31)

Access: RSCANnTMCp registers can be read/written in 8-bit units

Address: RSCANnTMCp: <RSCFDn_base> + 0250_H + (01_H × p)

Value after reset: 00_H

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	TMOM	TMTAR	TMTR
Value after reset:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W *1	R/W *1

Note 1. The only effective value for writing to this bit is 1, which sets the bit. Otherwise writing to the bit results in retention of its state.

Table 25.59 RSCANnTMCp Register Contents

Bit	Bit Name	Function
7 to 3	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
2	TMOM	One-Shot Transmission Enable 0: One-shot transmission is disabled. 1: One-shot transmission is enabled.
1	TMTAR	Transmit Abort Request 0: Transmit abort is not requested. 1: Transmit abort is requested.
0	TMTR	Transmit Request 0: Transmission is not requested. 1: Transmission is requested.

When the RSCANnTMCp register meets any of the following conditions, set it to 00_H.

- The RSCANnTMCp register corresponds to the transmit buffer number selected by the CFTML[3:0] bits in the RSCANnCFCCk register (p = m × 16 + the value of CFTML[3:0] bits).
- The RSCANnTMCp register corresponds to the transmit buffer allocated to the transmit queue by the TXQDC[3:0] bits in the RSCANnTXQCCm (m = 0, 1) register (p = (m × 16 + 15) to (m × 16 + 15 – the value of TXQDC[3:0] bits)).

Bits in the RSCANnTMCp register are all cleared to 0 in channel reset mode. Modify the RSCANnTMCp register in channel communication mode or channel halt mode.

TMOM Bit

Setting this bit to 1 enables one-shot transmission. When transmission fails, retransmission defined in the CAN protocol is not performed.
Modify the TMOM bit when the TMTRM flag in the RSCANnTMSTSp register is set to 0. Set the TMOM bit to 1 together with the TMTR bit.

TMTAR Bit

Setting this bit to 1 generates a transmit abort request for the message stored in the transmit buffer. However, a message that is being transmitted or one that will be transmitted next cannot be aborted.
The TMTAR bit can be set to 1 when TMTR bit is 1.
The TMTAR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed.
- An error or arbitration loss has been detected.

If this bit becomes 0 at the same time as the program writes 1 to this bit, this bit becomes 0.

TMTR Bit

Setting this bit to 1 transmits the message stored in the transmit buffer.
The TMTR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed after the TMTAR bit was set to 1.
- An error or arbitration-lost has been detected with the TMOM bit set to 1.

Set the TMTR bit to 1 when the value of TMTRF[1:0] in the RSCANnTMSTSp register is 00_B.

25.3.10.2 RSCANnTMSTSp — Transmit Buffer Status Register (p = 0 to 31)

Access: RSCANnTMSTSp registers can be read/written in 8-bit units

Address: RSCANnTMSTSp: <RSCFDn_base> + $02D0_H$ + (01_H × p)

Value after reset: 00_H

Bit:	7	6	5	4	3	2	1	0
	—	—	—	TMTARM	TMTRM	TMTRF[1:0]		TMTSTS
Value after reset:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R

Table 25.60 RSCANnTMSTSp Register Contents

Bit	Bit Name	Function
7 to 5	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
4	TMTARM	Transmit Buffer Transmit Abort Request Status Flag 0: No transmit abort request is present. 1: A transmit abort request is present.
3	TMTRM	Transmit Buffer Transmit Request Status Flag 0: No transmit request is present. 1: A transmit request is present.
2, 1	TMTRF[1:0]	Transmit Buffer Transmit Result Status Flag b2 b1 0 0: Transmission is in progress or no transmit request is present. 0 1: Transmit abort has been completed. 1 0: Transmission has been completed (without transmit abort request). 1 1: Transmission has been completed (with transmit abort request).
0	TMTSTS	Transmit Buffer Transmit Status Flag 0: Transmission is not in progress. 1: Transmission is in progress.

The RSCANnTMSTSp register is cleared to all 0 in channel reset mode.

TMTARM Flag

The TMTARM flag is set to 1 when the TMTAR bit in the RSCANnTMCp register is set to 1.
The TMTARM flag is set to 0 when the TMTAR bit in the RSCANnTMCp register is set to 0.

TMTRM Flag

The TMTRM flag is set to 1 when the TMTR bit in the RSCANnTMCp register is set to 1.
The TMTRM flag is set to 0 when the TMTR bit in the RSCANnTMCp register is set to 0.

TMTRF[1:0] Flag

This flag indicates the result of transmission from the transmit buffer.
00B: Transmission is in progress or no transmit request is present.
01B: Transmission from the transmit buffer was aborted.
10B: Transmission has been completed with the TMTAR bit in the RSCANnTMCp register set to 0 (transmit abort is not requested).
11B: Transmission has been completed with the TMTAR bit in the RSCANnTMCp register set to 1 (transmit abort is requested).
Write 00B to the TMTRF[1:0] flag in channel communication mode or channel halt mode. Do not write any value other than 00B to this flag.

TMTSTS Flag

This flag is set to 1 when transmission from the transmit buffer starts, and is cleared to 0 when transmission from the transmit buffer has been completed or terminated due to a bus error or arbitration lost.

25.3.10.3 RSCANnTMIDp — Transmit Buffer ID Register (p = 0 to 31)

Access: RSCANnTMIDp register can be read/written in 32-bit units
RSCANnTMIDpL, RSCANnTMIDpH registers can be read/written in 16-bit units
RSCANnTMIDpLL, RSCANnTMIDpLH, RSCANnTMIDpHL, RSCANnTMIDpHH registers can be read/written in 8-bit units

Address: RSCANnTMIDp: <RSCFDn_base> + 1000_H + (10_H × p)
RSCANnTMIDpL: <RSCFDn_base> + 1000_H + (10_H × p),
RSCANnTMIDpH: <RSCFDn_base> + 1002_H + (10_H × p)
RSCANnTMIDpLL: <RSCFDn_base> + 1000_H + (10_H × p),
RSCANnTMIDpLH: <RSCFDn_base> + 1001_H + (10_H × p),
RSCANnTMIDpHL: <RSCFDn_base> + 1002_H + (10_H × p),
RSCANnTMIDpHH: <RSCFDn_base> + 1003_H + (10_H × p)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIDE	TMRTR	THLEN	TMID[28:16]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMID[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.61 RSCANnTMIDp Register Contents

Bit	Bit Name	Function
31	TMIDE	Transmit Buffer IDE 0: Standard ID 1: Extended ID
30	TMRTR	Transmit Buffer RTR 0: Data frame 1: Remote frame
29	THLEN	Transmit History Data Store Enable 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	TMID[28:0]	Transmit Buffer ID Data Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11.

Modify this register when the TMTRM bit in the corresponding RSCANnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write data to this register. If this register is allocated to the transmit queue, only write data to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMIDE Bit

This bit is used to set the ID format of the message to be transmitted from the transmit buffer.

TMRTR Bit

This bit is used to set the data format of the message to be transmitted from the transmit buffer.

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, buffer type, and timestamp (timestamp is included if the TMTSCE bit in the RSCANnGCFG register is 1)) of transmit messages is stored in the

transmit history buffer after transmission is completed.

TMID[28:0] Bits

These bits are used to set the ID of the message to be transmitted from the transmit buffer.

25.3.10.4 RSCANnTMPTRp — Transmit Buffer Pointer Register (p = 0 to 31)

Access: RSCANnTMPTRp register can be read/written in 32-bit units
RSCANnTMPTRpL, RSCANnTMPTRpH registers can be read/written in 16-bit units
RSCANnTMPTRpLL, RSCANnTMPTRpLH, RSCANnTMPTRpHL, RSCANnTMPTRpHH registers can be read/written in 8-bit units

Address: RSCANnTMPTRp: <RSCFDn_base> + 1004_H + (10_H × p)
RSCANnTMPTRpL: <RSCFDn_base> + 1004_H + (10_H × p),
RSCANnTMPTRpH: <RSCFDn_base> + 1006_H + (10_H × p)
RSCANnTMPTRpLL: <RSCFDn_base> + 1004_H + (10_H × p),
RSCANnTMPTRpLH: <RSCFDn_base> + 1005_H + (10_H × p),
RSCANnTMPTRpHL: <RSCFDn_base> + 1006_H + (10_H × p),
RSCANnTMPTRpHH: <RSCFDn_base> + 1007_H + (10_H × p)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDLC[3:0]				—	—	—	—	TMPTR[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.62 RSCANnTMPTRp Register Contents

Bit	Bit Name	Function
31 to 28	TMDLC[3:0]	Transmit Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 24	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
23 to 16	TMPTR[7:0]	Transmit Buffer Label Data Set the label information to be stored in the transmit history buffer.
15 to 0	Reserved	These bits are read as the value after reset. The write value should be the value after reset.

Modify this register when the TMTRM bit in the corresponding RSCANnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMDLC[3:0] Bits

These bits are used to set the data length of the message to be transmitted from the transmit buffer when the TMRTR bit in the RSCANnTMIDp register is set to 0 (data frame). If the data length is set to 1001B or more, the transmit data is 8 bytes long.
When the TMRTR bit is set to 1 (remote frame), set the data length of messages to be requested.

TMPTR[7:0] Bits

When message transmission has been completed, the TMPTR[7:0] value is stored in the transmit history buffer.

25.3.10.5 RSCANnTMDF0_p — Transmit Buffer Data Field 0 Register (p = 0 to 31)

Access: RSCANnTMDF0_p register can be read/written in 32-bit units
RSCANnTMDF0_pL, RSCANnTMDF0_pH registers can be read/written in 16-bit units
RSCANnTMDF0_pLL, RSCANnTMDF0_pLH, RSCANnTMDF0_pHL, RSCANnTMDF0_pHH registers can be read/written in 8-bit units

Address: RSCANnTMDF0_p: <RSCFDn_base> + 1008_H + (10_H × p)
RSCANnTMDF0_pL: <RSCFDn_base> + 1008_H + (10_H × p),
RSCANnTMDF0_pH: <RSCFDn_base> + $100A_H$ + (10_H × p)
RSCANnTMDF0_pLL: <RSCFDn_base> + 1008_H + (10_H × p),
RSCANnTMDF0_pLH: <RSCFDn_base> + 1009_H + (10_H × p),
RSCANnTMDF0_pHL: <RSCFDn_base> + $100A_H$ + (10_H × p),
RSCANnTMDF0_pHH: <RSCFDn_base> + $100B_H$ + (10_H × p)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB3[7:0]								TMDB2[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB1[7:0]								TMDB0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.63 RSCANnTMDF0_p Register Contents

Bit	Bit Name	Function
31 to 24	TMDB3[7:0]	Transmit Buffer Data Byte 3
23 to 16	TMDB2[7:0]	Transmit Buffer Data Byte 2
15 to 8	TMDB1[7:0]	Transmit Buffer Data Byte 1
7 to 0	TMDB0[7:0]	Transmit Buffer Data Byte 0 Set the transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCANnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

25.3.10.6 RSCANnTMDF1_p — Transmit Buffer Data Field 1 Register (p = 0 to 31)

Access: RSCANnTMDF1_p register can be read/written in 32-bit units
RSCANnTMDF1_pL, RSCANnTMDF1_pH registers can be read/written in 16-bit units
RSCANnTMDF1_pLL, RSCANnTMDF1_pLH, RSCANnTMDF1_pHL, RSCANnTMDF1_pHH registers can be read/written in 8-bit units

Address: RSCANnTMDF1_p: <RSCFDn_base> + $100C_H$ + (10_H × p)
RSCANnTMDF1_pL: <RSCFDn_base> + $100C_H$ + (10_H × p),
RSCANnTMDF1_pH: <RSCFDn_base> + $100E_H$ + (10_H × p)
RSCANnTMDF1_pLL: <RSCFDn_base> + $100C_H$ + (10_H × p),
RSCANnTMDF1_pLH: <RSCFDn_base> + $100D_H$ + (10_H × p),
RSCANnTMDF1_pHL: <RSCFDn_base> + $100E_H$ + (10_H × p),
RSCANnTMDF1_pHH: <RSCFDn_base> + $100F_H$ + (10_H × p)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB7[7:0]								TMDB6[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB5[7:0]								TMDB4[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.64 RSCANnTMDF1_p Register Contents

Bit	Bit Name	Function
31 to 24	TMDB7[7:0]	Transmit Buffer Data Byte 7
23 to 16	TMDB6[7:0]	Transmit Buffer Data Byte 6
15 to 8	TMDB5[7:0]	Transmit Buffer Data Byte 5
7 to 0	TMDB4[7:0]	Transmit Buffer Data Byte 4 Set the transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCANnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

25.3.10.7 RSCANnTMIECy — Transmit Buffer Interrupt Enable Configuration Register (y = 0)

Access: RSCANnTMIECy register can be read/written in 32-bit units
RSCANnTMIECyL, RSCANnTMIECyH registers can be read/written in 16-bit units
RSCANnTMIECyLL, RSCANnTMIECyLH, RSCANnTMIECyHL, RSCANnTMIECyHH registers can be read/written in 8-bit units

Address: RSCANnTMIECy: <RSCFDn_base> + 0390_H + (04_H × y)
RSCANnTMIECyL: <RSCFDn_base> + 0390_H + (04_H × y),
RSCANnTMIECyH: <RSCFDn_base> + 0392_H + (04_H × y)
RSCANnTMIECyLL: <RSCFDn_base> + 0390_H + (04_H × y),
RSCANnTMIECyLH: <RSCFDn_base> + 0391_H + (04_H × y),
RSCANnTMIECyHL: <RSCFDn_base> + 0392_H + (04_H × y),
RSCANnTMIECyHH: <RSCFDn_base> + 0393_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIEp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMIEp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.65 RSCANnTMIECy Register Contents

Bit	Bit Name	Function
31 to 16	TMIEp	Transmit Buffer Interrupt Enable p (p = y × 32 + 31 to y × 32 + 16) 0: Transmit buffer interrupt is disabled 1: Transmit buffer interrupt is enabled
15 to 0	TMIEp	Transmit Buffer Interrupt Enable p (p = y × 32 + 15 to y × 32 + 0) 0: Transmit buffer interrupt is disabled. 1: Transmit buffer interrupt is enabled.

TMIEp Bits (p = 0 to 31)

When any of these bits is set to 1 and the corresponding transmission has been completed, a transmit buffer interrupt request is generated.

Modify these bits when the TMTRM flag in the corresponding RSCANnTMSTSp register is 0 (no transmit request is present).

Write 0 to bits corresponding to transmit buffers linked to transmit/receive FIFO buffers or transmit buffers allocated to the transmit queue.

Table 25.66 shows the bit assignment.

Table 25.66 TMIEp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

25.3.11 Details of Transmit Buffer Status-related Registers

25.3.11.1 RSCANnTMTRSTSy — Transmit Buffer Transmit Request Status Register (y = 0)

Access: RSCANnTMTRSTSy register can be read only in 32-bit units
RSCANnTMTRSTSyL, RSCANnTMTRSTSyH registers can be read only in 16-bit units
RSCANnTMTRSTSyLL, RSCANnTMTRSTSyLH, RSCANnTMTRSTSyHL, RSCANnTMTRSTSyHH registers can be read only in 8-bit units

Address: RSCANnTMTRSTSy: <RSCFDn_base> + 0350_H + (04_H × y)
RSCANnTMTRSTSyL: <RSCFDn_base> + 0350_H + (04_H × y),
RSCANnTMTRSTSyH: <RSCFDn_base> + 0352_H + (04_H × y)
RSCANnTMTRSTSyLL: <RSCFDn_base> + 0350_H + (04_H × y),
RSCANnTMTRSTSyLH: <RSCFDn_base> + 0351_H + (04_H × y),
RSCANnTMTRSTSyHL: <RSCFDn_base> + 0352_H + (04_H × y),
RSCANnTMTRSTSyHH: <RSCFDn_base> + 0353_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTRSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTRSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.67 RSCANnTMTRSTSy Register Contents

Bit	Bit Name	Function
31 to 16	TMTRSTSp	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit request is present. 1: A transmit request is present.
15 to 0	TMTRSTSp	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit request is present. 1: A transmit request is present.

TMTRSTSp Flags (p = 0 to 31)

These flags indicate the status of the TMTR bit in the RSCANnTMCp register.

When the TMTR bit is set to 1 (transmission is requested), the corresponding TMTRSTSp flag is set to 1.

The corresponding TMTRSTSp flag is cleared to 0 when the TMTR bit is set to 0 (transmission is not requested) or in channel reset mode.

Table 25.68 shows the bit assignment.

Table 25.68 TMTRSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

25.3.11.2 RSCANnTMTARSTSy — Transmit Buffer Transmit Abort Request Status Register (y = 0)

Access: RSCANnTMTARSTSy register can be read only in 32-bit units
RSCANnTMTARSTSyL, RSCANnTMTARSTSyH registers can be read only in 16-bit units
RSCANnTMTARSTSyLL, RSCANnTMTARSTSyLH, RSCANnTMTARSTSyHL, RSCANnTMTARSTSyHH registers can be read only in 8-bit units

Address: RSCANnTMTARSTSy: <RSCFDn_base> + 0360_H + (04_H × y)
RSCANnTMTARSTSyL: <RSCFDn_base> + 0360_H + (04_H × y),
RSCANnTMTARSTSyH: <RSCFDn_base> + 0362_H + (04_H × y)
RSCANnTMTARSTSyLL: <RSCFDn_base> + 0360_H + (04_H × y),
RSCANnTMTARSTSyLH: <RSCFDn_base> + 0361_H + (04_H × y),
RSCANnTMTARSTSyHL: <RSCFDn_base> + 0362_H + (04_H × y),
RSCANnTMTARSTSyHH: <RSCFDn_base> + 0363_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTARSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTARSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.69 RSCANnTMTRSTSy Register Contents

Bit	Bit Name	Function
31 to 16	TMTARSTSp	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit abort request is present. 1: A transmit abort request is present.
15 to 0	TMTARSTSp	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit abort request is present. 1: A transmit abort request is present.

TMTARSTSp Flags (p = 0 to 31)

These flags indicate the status of the TMTAR bit in the RSCANnTMCp register.

When the TMTAR bit is set to 1 (transmit abort is requested), the corresponding TMTARSTSp flag is set to 1.

The corresponding TMTARSTSp flag is cleared to 0 when the TMTAR bit is set to 0 (transmit abort is not requested) or in channel reset mode.

Table 25.70 shows the bit assignment.

Table 25.70 TMTARSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

25.3.11.3 RSCANnTMTCSTSy — Transmit Buffer Transmit Complete Status Register (y = 0)

Access: RSCFDnTMTCSTSyRSCANnTMTCSTSy register can be read only in 32-bit units
RSCANnTMTCSTSyL, RSCANnTMTCSTSyH registers can be read only in 16-bit units
RSCANnTMTCSTSyLL, RSCANnTMTCSTSyLH, RSCANnTMTCSTSyHL, RSCANnTMTCSTSyHH registers can be read only in 8-bit units

Address: RSCANnTMTCSTSy: <RSCFDn_base> + 0370_H + (04_H × y)
RSCANnTMTCSTSyL: <RSCFDn_base> + 0370_H + (04_H × y),
RSCANnTMTCSTSyH: <RSCFDn_base> + 0372_H + (04_H × y)
RSCANnTMTCSTSyLL: <RSCFDn_base> + 0370_H + (04_H × y),
RSCANnTMTCSTSyLH: <RSCFDn_base> + 0371_H + (04_H × y),
RSCANnTMTCSTSyHL: <RSCFDn_base> + 0372_H + (04_H × y),
RSCANnTMTCSTSyHH: <RSCFDn_base> + 0373_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTCSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTCSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.71 RSCANnTMTCSTSy Register Contents

Bit	Bit Name	Function
31 to 16	TMTCSTSp	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission has not been completed. 1: Transmission has been completed.
15 to 0	TMTCSTSp	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission has not been completed. 1: Transmission has been completed.

TMTCSTSp Flags (p = 0 to 31)

When the TMTRF[1:0] flag in the RSCANnTMSTSp register is set to 10_B (transmission has been completed (without transmit abort request)) or 11_B (transmission has been completed (with transmit abort request)), the corresponding TMTCSTSp flag is set to 1.

A TMTCSTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00_B or in channel reset mode.

Table 25.72 shows the bit assignment.

Table 25.72 TMTCSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

25.3.11.4 RSCANnTMTASTSy — Transmit Buffer Transmit Abort Status Register (y = 0)

Access: RSCANnTMTASTSy register can be read only in 32-bit units
RSCANnTMTASTSyL, RSCANnTMTASTSyH registers can be read only in 16-bit units
RSCANnTMTASTSyLL, RSCANnTMTASTSyLH, RSCANnTMTASTSyHL, RSCANnTMTASTSyHH registers can be read only in 8-bit units

Address: RSCANnTMTASTSy: <RSCFDn_base> + 0380_H + (04_H × y)
RSCANnTMTASTSyL: <RSCFDn_base> + 0380_H + (04_H × y),
RSCANnTMTASTSyH: <RSCFDn_base> + 0382_H + (04_H × y)
RSCANnTMTASTSyLL: <RSCFDn_base> + 0380_H + (04_H × y),
RSCANnTMTASTSyLH: <RSCFDn_base> + 0381_H + (04_H × y),
RSCANnTMTASTSyHL: <RSCFDn_base> + 0382_H + (04_H × y),
RSCANnTMTASTSyHH: <RSCFDn_base> + 0383_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTASTSp (p = y × 32 + 31 to y x 32 + 16 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTASTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.73 RSCANnTMTASTSy Register Contents

Bit	Bit Name	Function
31 to 16	TMTASTSp	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission is not aborted 1: Transmission is aborted
15 to 0	TMTASTSp	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission is not aborted. 1: Transmission is aborted.

TMTASTSp Flags (p = 0 to 31)

When the TMTRF[1:0] flag in the RSCANnTMSTSp register is set to 01_B (transmit abort has been completed), the corresponding TMTASTSp flag is set to 1.

A TMTASTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00_B or in channel reset mode.

Table 25.74 shows the bit assignment.

Table 25.74 TMTASTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

25.3.12 Details of Transmit Queue-related Registers

25.3.12.1 RSCANnTXQCCm — Transmit Queue Configuration and Control Register (m = 0, 1)

Access: RSCANnTXQCCm register can be read/written in 32-bit units
RSCANnTXQCCmL, RSCANnTXQCCmH registers can be read/written in 16-bit units
RSCANnTXQCCmLL, RSCANnTXQCCmLH, RSCANnTXQCCmHL, RSCANnTXQCCmHH registers can be read/written in 8-bit units

Address: RSCANnTXQCCm: <RSCFDn_base> + $03A0_H$ + (04_H × m)
RSCANnTXQCCmL: <RSCFDn_base> + $03A0_H$ + (04_H × m),
RSCANnTXQCCmH: <RSCFDn_base> + $03A2_H$ + (04_H × m)
RSCANnTXQCCmLL: <RSCFDn_base> + $03A0_H$ + (04_H × m),
RSCANnTXQCCmLH: <RSCFDn_base> + $03A1_H$ + (04_H × m),
RSCANnTXQCCmHL: <RSCFDn_base> + $03A2_H$ + (04_H × m),
RSCANnTXQCCmHH: <RSCFDn_base> + $03A3_H$ + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQIM	TXQIE	TXQDC[3:0]				—	—	—	—	—	—	—	TXQE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 25.75 RSCANnTXQCCm Register Contents

Bit	Bit Name	Function
31 to 14	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
13	TXQIM	Transmit Queue Interrupt Source Select 0: When the transmit queue becomes empty upon completion of message transmission, a transmit queue interrupt request is generated. 1: A transmit queue interrupt request is generated each time a message has been transmitted.
12	TXQIE	Transmit Queue Interrupt Enable 0: Transmit queue interrupt is disabled. 1: Transmit queue interrupt is enabled.
11 to 8	TXQDC[3:0]	Transmit Queue Depth Configuration Setting these bits to g (g = 2 to 15) makes the (g + 1)-buffer transmit queue available. Setting these bits to 0 disables the transmit queue. Setting these bits to 1 is prohibited.
7 to 1	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
0	TXQE	Transmit Queue Enable 0: The transmit queue is not used. 1: The transmit queue is used.

TXQIM Bit

This bit is used to select a transmit queue interrupt source. Modify this bit in channel reset mode.

TXQIE Bit

When the TXQIE bit is set to 1 and the source selected by the TXQIM bit occurs, an interrupt request is generated.
Set the TXQE bit to 0 before modifying the TXQIE bit.

TXQDC[3:0] Bits

These bits are used to specify the number of transmit buffers to be allocated to the transmit queues. Transmit buffers are allocated to transmit queues in descending order of buffer number, that is, from (m × 16 + 15) to (m × 16 + 0). For examples of how buffer allocation is done, see Figure 25.9. Modify these bits only in channel reset mode.

TXQE Bit

Setting this bit to 1 makes the transmit queue available. Modify this bit in channel communication mode or channel halt mode. This bit is cleared to 0 in channel reset mode.
Before setting the TXQE bit to 1, set the TXQDC[3:0] bits to 0010_B or more.

25.3.12.2 RSCANnTXQSTSm — Transmit Queue Status Register (m = 0, 1)

Access: RSCANnTXQSTSm register can be read/written in 32-bit units
RSCANnTXQSTSmL, RSCANnTXQSTSmH registers can be read/written in 16-bit units
RSCANnTXQSTSmLL, RSCANnTXQSTSmLH, RSCANnTXQSTSmHL, RSCANnTXQSTSmHH registers can be read/written in 8-bit units

Address: RSCANnTXQSTSm: <RSCFDn_base> + $03C0_H$ + (04_H × m)
RSCANnTXQSTSmL: <RSCFDn_base> + $03C0_H$ + (04_H × m),
RSCANnTXQSTSmH: <RSCFDn_base> + $03C2_H$ + (04_H × m)
RSCANnTXQSTSmLL: <RSCFDn_base> + $03C0_H$ + (04_H × m),
RSCANnTXQSTSmLH: <RSCFDn_base> + $03C1_H$ + (04_H × m),
RSCANnTXQSTSmHL: <RSCFDn_base> + $03C2_H$ + (04_H × m),
RSCANnTXQSTSmHH: <RSCFDn_base> + $03C3_H$ + (04_H × m)

Value after reset: $0000\ 0001_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQIF	TXQFLL	TXQEMP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.76 RSCANnTXQSTSm Register Contents

Bit	Bit Name	Function
31 to 13	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
12 to 8	Reserved	When read, an undefined value is returned.The write value should be the value after reset.
7 to 3	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
2	TXQIF	Transmit Queue Interrupt Request Flag 0: No transmit queue interrupt request is present. 1: A transmit queue interrupt request is present.
1	TXQFLL	Transmit Queue Full Status Flag 0: The transmit queue is not full. 1: The transmit queue is full.
0	TXQEMP	Transmit Queue Empty Status Flag 0: The transmit queue contains messages. 1: The transmit queue contains no message (transmit queue empty).

TXQIF Flag

The TXQIF flag is set to 1 when the event specified by the TXQIM bit in the RSCANnTXQCCm register has occurred. The TXQIF flag is cleared to 0 in channel reset mode or by writing 0 to this flag. This flag is not cleared to 0 by setting the TXQE bit in the RSCANnTXQCCm register to 0 (the transmit queue is not used).

TXQFLL Flag

The TXQFLL flag is set to 1 when the number of messages set for the transmit queue matches the transmit queue depth set by the TXQDC[3:0] bits in the RSCANnTXQCCm register.
This flag is cleared to 0 in any of the following cases.

- The number of messages set for the transmit queue is smaller than the transmit queue depth set by the TXQDC[3:0] bits.
- In channel reset mode

TXQEMP Flag

The TXQEMP flag is cleared to 0 when even a single message is set for the transmit queue.
This flag is set to 1 in any of the following cases.

- The TXQE bit is set to 0 (the transmit queue is not used).
- The transmit queue becomes empty.
- In channel reset mode

25.3.12.3 RSCANnTXQPCTRm — Transmit Queue Pointer Control Register (m = 0, 1)

Access: RSCANnTXQPCTRm register can only be written in 32-bit units
RSCANnTXQPCTRmL, RSCANnTXQPCTRmH registers can only be written in 16-bit units
RSCANnTXQPCTRmLL, RSCANnTXQPCTRmLH, RSCANnTXQPCTRmHL, RSCANnTXQPCTRmHH registers can only be written in 8-bit units

Address: RSCANnTXQPCTRm: <RSCFDn_base> + $03E0_H$ + (04_H × m)
RSCANnTXQPCTRmL: <RSCFDn_base> + $03E0_H$ + (04_H × m),
RSCANnTXQPCTRmH: <RSCFDn_base> + $03E2_H$ + (04_H × m)
RSCANnTXQPCTRmLL: <RSCFDn_base> + $03E0_H$ + (04_H × m),
RSCANnTXQPCTRmLH: <RSCFDn_base> + $03E1_H$ + (04_H × m),
RSCANnTXQPCTRmHL: <RSCFDn_base> + $03E2_H$ + (04_H × m),
RSCANnTXQPCTRmHH: <RSCFDn_base> + $03E3_H$ + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 25.77 RSCANnTXQPCTRm Register Contents

Bit	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	TXQPC[7:0]	Transmit Queue Pointer Control Writing FFH to these bits moves the write pointer of the transmit queue to the next queue buffer.

TXQPC[7:0] Bits

Writing FFH to the TXQPC[7:0] bits moves the write pointer to the next transmit queue buffer and generates a transmit request of the message. Write transmit messages to the RSCANnTMIDp, RSCANnTMPTRp, RSCANnTMDF0_p, and RSCANnTMDF1_p registers (p = 15, 31) before writing FFH to the TXQPC[7:0] bits.
When writing FFH to these bits, make sure that the TXQE bit in the RSCANnTXQCCm register is set to 1 (the transmit queue is used) and the TXQFLL flag in the RSCANnTXQSTSm register is 0 (the transmit queue is not full).

25.3.13 Details of Transmit history-related Registers

25.3.13.1 RSCANnTHLCCm — Transmit History Configuration and Control Register (m = 0, 1)

Access: RSCANnTHLCCm register can be read/written in 32-bit units
RSCANnTHLCCmL, RSCANnTHLCCmH registers can be read/written in 16-bit units
RSCANnTHLCCmLL, RSCANnTHLCCmLH, RSCANnTHLCCmHL, RSCANnTHLCCmHH registers can be read/written in 8-bit units

Address: RSCANnTHLCCm: <RSCFDn_base> + 0400_H + (04_H × m)
RSCANnTHLCCmL: <RSCFDn_base> + 0400_H + (04_H × m),
RSCANnTHLCCmH: <RSCFDn_base> + 0402_H + (04_H × m)
RSCANnTHLCCmLL: <RSCFDn_base> + 0400_H + (04_H × m),
RSCANnTHLCCmLH: <RSCFDn_base> + 0401_H + (04_H × m),
RSCANnTHLCCmHL: <RSCFDn_base> + 0402_H + (04_H × m),
RSCANnTHLCCmHH: <RSCFDn_base> + 0403_H + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLDTE	THLIM	THLIE	—	—	—	—	—	—	—	THLE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 25.78 RSCANnTHLCCm Register Contents

Bit	Bit Name	Function
31 to 11	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
10	THLDTE	Transmit History Target Buffer Select 0: Entry from transmit/receive FIFO buffers and transmit queue 1: Entry from transmit buffers, transmit/receive FIFO buffers, and transmit queue
9	THLIM	Transmit History Interrupt Source Select 0: When 12 sets of data have been stored in the transmit history buffer 1: When a single set of transmit history data has been stored
8	THLIE	Transmit History Interrupt Enable 0: Transmit history interrupt is disabled. 1: Transmit history interrupt is enabled.
7 to 1	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
0	THLE	Transmit History Buffer Enable 0: Transmit history buffer is not used. 1: Transmit history buffer is used.

THLDTE Bit

When this bit is set to 0, the transmit history data of messages transmitted from transmit/receive FIFO buffers and the transmit queue is stored in the transmit history buffer. When this bit is set to 1, the transmit history data of messages transmitted from transmit buffers, transmit/receive FIFO buffers, and the transmit queue is stored in the transmit history buffer.

Modify this bit only in channel reset mode.

THLIM Bit

This bit is used to select a transmit history interrupt source.
Modify this bit only in channel reset mode.

THLIE Bit

When the THLIE bit is set to 1 and the source selected by the THLIM bit has occurred, a transmit history interrupt request is generated. Modify the THLIE bit only when the THLE bit set to 0.

THLE Bit

Setting this bit to 1 makes the transmit history buffer available. When data transmission from the buffer selected by the THLDTE bit has been completed, the transmit history data of transmit messages is stored in the transmit history buffer. Modify this bit in channel communication mode or channel halt mode.
This bit is cleared to 0 in channel reset mode.

25.3.13.2 RSCANnTHLSTSm — Transmit History Status Register (m = 0, 1)

Access: RSCANnTHLSTSm register can be read/written in 32-bit units
RSCANnTHLSTSmL, RSCANnTHLSTSmH registers can be read/written in 16-bit units
RSCANnTHLSTSmLL, RSCANnTHLSTSmLH, RSCANnTHLSTSmHL, RSCANnTHLSTSmHH registers can be read/written in 8-bit units

Address: RSCANnTHLSTSm: <RSCFDn_base> + 0420_H + (04_H × m)
RSCANnTHLSTSmL: <RSCFDn_base> + 0420_H + (04_H × m),
RSCANnTHLSTSmH: <RSCFDn_base> + 0422_H + (04_H × m)
RSCANnTHLSTSmLL: <RSCFDn_base> + 0420_H + (04_H × m),
RSCANnTHLSTSmLH: <RSCFDn_base> + 0421_H + (04_H × m),
RSCANnTHLSTSmHL: <RSCFDn_base> + 0422_H + (04_H × m),
RSCANnTHLSTSmHH: <RSCFDn_base> + 0423_H + (04_H × m)

Value after reset: $0000\ 0001_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THLMC[4:0]					—	—	—	—	THLIF	THLELT	THLFLL	THLEMP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.79 RSCANnTHLSTSm Register Contents

Bit	Bit Name	Function
31 to 13	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
12 to 8	THLMC[4:0]	Transmit History Buffer Unread Data Counter These bits indicate the number of unread data sets stored in the transmit history buffer.
7 to 4	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
3	THLIF	Transmit History Interrupt Request Flag 0: No transmit history interrupt request is present. 1: A transmit history interrupt request is present.
2	THLELT	Transmit History Buffer Overflow Flag 0: Transmit history buffer overflow has not occurred. 1: Transmit history buffer overflow has occurred.
1	THLFLL	Transmit history Buffer Full Status Flag 0: Transmit history buffer is not full. 1: Transmit history buffer is full.
0	THLEMP	Transmit History Buffer Empty Status Flag 0: Transmit history buffer contains unread data. 1: Transmit history buffer contains no unread data (buffer empty).

HLMC[4:0] Bits

These bits indicate the number of unread data sets stored in the transmit history buffer.

These bits are cleared to 0 in channel reset mode.

HLIF Flag

The THLIF flag is set to 1 when the interrupt source specified with the THLIM bit in the RSCANnTHLCCm register occurs.

This flag is cleared to 0 in channel reset mode or by the program writing 0 to this flag.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.
When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

THLELT Flag

The THLELT flag is set to 1 when an attempt is made to store new transmit history data while the transmit history buffer is full. In this case, the new data is discarded. This flag becomes 0 in channel reset mode or by the program writing 0 to this flag.
To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.
When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

THLFLL Flag

The THLFLL flag is set to 1 when 16 data sets have been stored in the transmit history buffer, and is cleared to 0 when the number of data sets stored in the transmit history buffer has decreased to less than 16. This bit is also cleared to 0 in channel reset mode or when the THLE bit in the RSCANnTHLCCm register is set to 0 (transmit history buffer is not used).

THLEMP Flag

The THLEMP flag is cleared to 0 when even a single set of transmit history data has been stored in the transmit history buffer.
This flag is set to 1 when all the data in the transmit history buffer has been read. This flag is also set to 1 in channel reset mode or when the THLE bit in the RSCANnTHLCCm register is set to 0 (transmit history buffer is not used).

Note: To clear THLIF or THLELT flag to 0, the program must write 0. When writing, use a store instruction to write “0” to the given flag and “1” to other flags.

25.3.13.3 RSCANnTHLPCTRm — Transmit History Pointer Control Register (m = 0, 1)

Access: RSCANnTHLPCTRm register can only be written in 32-bit units
RSCANnTHLPCTRmL, RSCANnTHLPCTRmH registers can only be written in 16-bit units
RSCANnTHLPCTRmLL, RSCANnTHLPCTRmLH, RSCANnTHLPCTRmHL, RSCANnTHLPCTRmHH registers can only be written in 8-bit units

Address: RSCANnTHLPCTRm: <RSCFDn_base> + 0440_H + (04_H × m)
RSCANnTHLPCTRmL: <RSCFDn_base> + 0440_H + (04_H × m),
RSCANnTHLPCTRmH: <RSCFDn_base> + 0442_H + (04_H × m)
RSCANnTHLPCTRmLL: <RSCFDn_base> + 0440_H + (04_H × m),
RSCANnTHLPCTRmLH: <RSCFDn_base> + 0441_H + (04_H × m),
RSCANnTHLPCTRmHL: <RSCFDn_base> + 0442_H + (04_H × m),
RSCANnTHLPCTRmHH: <RSCFDn_base> + 0443_H + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	THLPC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 25.80 RSCANnTHLPCTRm Register Contents

Bit	Bit Name	Function
31 to 8	Reserved	When writing to these bits, write the value after reset.
7 to 0	THLPC[7:0]	Transmit History List Pointer Control Writing FFH to these bits moves the read pointer to the next unread data in the transmit history buffer.

HLPC[7:0] Bits

When the THLPC[7:0] bits are set to FFH, the read pointer moves to the next data in the transmit history buffer. At this time, the THLMC[4:0] (transmit history buffer unread data counter) value in the RSCANnTHLSTSm register is decremented. Write FFH to the THLPC[7:0] bits after reading from the RSCANnTHLACCm register.
When writing FFH to these bits, make sure that the THLE bit in the RSCANnTHLCCm register is set to 1 (transmit history buffer is used) and the THLEMP flag in the RSCANnTHLSTSm register is 0.

25.3.13.4 RSCANnTHLACCm — Transmit History Access Register (m = 0, 1)

Access: RSCANnTHLACCm register can be read only in 32-bit units
RSCANnTHLACCmL, RSCANnTHLACCmH registers can be read only in 16-bit units
RSCANnTHLACCmLL, RSCANnTHLACCmLH, RSCANnTHLACCmHL, RSCANnTHLACCmHH registers can be read only in 8-bit units

Address: RSCANnTHLACCm: <RSCFDn_base> + 1800_H + (04_H × m)
RSCANnTHLACCmL: <RSCFDn_base> + 1800_H + (04_H × m),
RSCANnTHLACCmH: <RSCFDn_base> + 1802_H + (04_H × m)
RSCANnTHLACCmLL: <RSCFDn_base> + 1800_H + (04_H × m),
RSCANnTHLACCmLH: <RSCFDn_base> + 1801_H + (04_H × m),
RSCANnTHLACCmHL: <RSCFDn_base> + 1802_H + (04_H × m),
RSCANnTHLACCmHH: <RSCFDn_base> + 1803_H + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TID[7:0]								—	BN[3:0]				BT[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.81 RSCANnTHLACCm Register Contents

Bit	Bit Name	Function
31 to 16	TMTS[15:0]	Timestamp Data The timestamp data of stored data can be read.
15 to 8	TID[7:0]	Label Data The label information of stored data can be read.
7	Reserved	This bit is read as the value after reset.
6 to 3	BN[3:0]	Buffer Number Data The buffer number of transmit source (transmit buffer, transmit/receive FIFO or transmit queue) can be read.
2 to 0	BT[2:0]	Buffer Type Data b2 b1 b0 0 0 1: Transmit buffer 0 1 0: Transmit/receive FIFO buffer 1 0 0: Transmit queue

TMTS[15:0] Bits

When the TMTSCE bit in the RSCANnGCFG register is 1, timestamp values in transmit history data stored in the transmit history buffer are displayed. When the TMTSCE bit is 0, these bits are always read as 0.

TID[7:0] Bits

These bits indicate the label information of transmit history data stored in the transmit history buffer.

BN[3:0] Bits

These bits indicate the transmit source buffer number in the transmit history data stored in the transmit history buffer.

BT[2:0] Bits

These bits indicate the type of the transmit source buffer in the transmit history data stored in the transmit history buffer.

25.3.14 Details of Test-related Registers

25.3.14.1 RSCANnGTSTCFG — Global Test Configuration Register

Access: RSCANnGTSTCFG register can be read/written in 32-bit units
RSCANnGTSTCFGL, RSCANnGTSTCFGH registers can be read/written in 16-bit units
RSCANnGTSTCFGLL, RSCANnGTSTCFGLH, RSCANnGTSTCFGHL, RSCANnGTSTCFGHH registers can be read/written in 8-bit units

Address: RSCANnGTSTCFG: <RSCFDn_base> + 0468_H
RSCANnGTSTCFGL: <RSCFDn_base> + 0468_H, RSCANnGTSTCFGH: <RSCFDn_base> + $046A_H$
RSCANnGTSTCFGLL: <RSCFDn_base> + 0468_H, RSCANnGTSTCFGLH: <RSCFDn_base> + 0469_H,
RSCANnGTSTCFGHL: <RSCFDn_base> + $046A_H$, RSCANnGTSTCFGHH: <RSCFDn_base> + $046B_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	RTMPS[6:0]						
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	C1ICBCE	C0ICBCE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.82 RSCANnGTSTCFG Register Contents

Bit	Bit Name	Function
31 to 23	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
22 to 16	RTMPS[6:0]	RAM Test Page Configuration Set a value within a range of page 0 (00_H) to page 19 (13_H).
15 to 2	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
1	C1ICBCE	CAN1 Inter-Channel Communication Test Enable 0: CAN1 inter-channel communication test is disabled. 1: CAN1 inter-channel communication test is enabled.
0	C0ICBCE	CAN0 Inter-Channel Communication Test Enable 0: CAN0 inter-channel communication test is disabled. 1: CAN0 inter-channel communication test is enabled.

Modify the RSCANnGTSTCFG register only in global test mode.

RTMPS[6:0] Bits

These bits are used to set the RAM test target page number for RAM test. Set a value in the range of 00_H to 13_H, inclusive.

C1ICBCE Bit

Setting this bit to 1 enables the channel 1 inter-channel communication test.
This bit is cleared to 0 in global reset mode.

C0ICBCE Bit

Setting this bit to 1 enables the channel 0 inter-channel communication test.
This bit is cleared to 0 in global reset mode.

25.3.14.2 RSCANnGTSTCTR — Global Test Control Register

Access: RSCANnGTSTCTR register can be read/written in 32-bit units
RSCANnGTSTCTRL, RSCANnGTSTCTRH registers can be read/written in 16-bit units
RSCANnGTSTCTRLL, RSCANnGTSTCTRLH, RSCANnGTSTCTRHL, RSCANnGTSTCTRHH registers can be read/written in 8-bit units

Address: RSCANnGTSTCTR: <RSCFDn_base> + $046C_H$
RSCANnGTSTCTRL: <RSCFDn_base> + $046C_H$, RSCANnGTSTCTRH: <RSCFDn_base> + $046E_H$
RSCANnGTSTCTRLL: <RSCFDn_base> + $046C_H$, RSCANnGTSTCTRLH: <RSCFDn_base> + $046D_H$,
RSCANnGTSTCTRHL: <RSCFDn_base> + $046E_H$, RSCANnGTSTCTRHH: <RSCFDn_base> + $046F_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RTME	—	ICBCTME
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W

Table 25.83 RSCANnGTSTCTR Register Contents

Bit	Bit Name	Function
31 to 3	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
2	RTME	RAM Test Enable 0: RAM test is disabled. 1: RAM test is enabled.
1	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
0	ICBCTME	Communication Test between Channels Enable 0: Communication test between channels disabled 1: Communication test between channels enabled

RTME Bit

Setting this bit to 1 enables the RAM test. Modify this bit only in global test mode.
This bit is cleared to 0 in global reset mode.

1. Set the GMDC[1:0] bits in the RSCANnGCTR register to 10B (Global test mode).
2. Set the RTME bit to 1.
3. Check that the RTME bit is set to 1.

ICBCTME Bit

When this bit is set to 1, a communication test is enabled between the channels for which the CmICBCE bit (m = 0, 1) in the RSCANnGTSTCFG register has been set to 1. Modify the ICBCTME bit only in global test mode.
This bit is cleared to 0 in global reset mode.

25.3.14.3 RSCANnGLOCKK — Global Lock Key Register

Access: RSCANnGLOCKK register can only be written in 32-bit units
RSCANnGLOCKKL, RSCANnGLOCKKH registers can only be written in 16-bit units

Address: RSCANnGLOCKK: <RSCFDn_base> + $047C_H$
RSCANnGLOCKKL: <RSCFDn_base> + $047C_H$, RSCANnGLOCKKH: <RSCFDn_base> + $047E_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCK[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1

Note 1. Writing to these bits is effective only when the RS-CANFD module is in global test mode.

Table 25.84 RSCANnGLOCKK Register Contents

Bit	Bit Name	Function
31 to 16	Reserved	When writing these bits, write the value after reset.
15 to 0	LOCK[15:0]	Lock Key These bits are key bits to release protection of test mode.

The RSCANnGLOCKK register releases protection of special test bits and is write only.
For the protection release data, see section 25.11.4.2, Procedure for Releasing the Protection.

LOCK[15:0] Bits

Writing the protection release data to the LOCK[15:0] bits in succession enables writing 1 to the RTME bit in the RSCANnGTSTCTR register.
After the protection has been released, writing to the I/O register area (<RSCFDn_base> + 0000H to <RSCFDn_base> + 04FFH) of the CAN (except the RAM) enables the protection again.
Reading from the I/O register area of the CAN or reading from/writing to other areas does not enable the protection.

25.3.14.4 RSCANnRPGACCr — RAM Test Page Access Register (r = 0 to 63)

Access: RSCANnRPGACCr register can be read/written in 32-bit units
RSCANnRPGACCrL, RSCANnRPGACCrH registers can be read/written in 16-bit units
RSCANnRPGACCrLL, RSCANnRPGACCrLH, RSCANnRPGACCrHL, RSCANnRPGACCrHH registers can be read/written in 8-bit units

Address: RSCANnRPGACCr: <RSCFDn_base> + 1900_H + (04_H × r)
RSCANnRPGACCrL: <RSCFDn_base> + 1900_H + (04_H × r),
RSCANnRPGACCrH: <RSCFDn_base> + 1902_H + (04_H × r)
RSCANnRPGACCrLL: <RSCFDn_base> + 1900_H + (04_H × r),
RSCANnRPGACCrLH: <RSCFDn_base> + 1901_H + (04_H × r),
RSCANnRPGACCrHL: <RSCFDn_base> + 1902_H + (04_H × r),
RSCANnRPGACCrHH: <RSCFDn_base> + 1903_H + (04_H × r)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDTA[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDTA[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.85 RSCANnRPGACCr Register Contents

Bit	Bit Name	Function
31 to 0	RDTA[31:0]	RAM Data Test Access Data can be read and written in RSCAN RAM.

Modify the RSCANnRPGACCr register in global test mode with the RTME bit in the RSCANnGTSTCTR register set to 1 (RAM test is enabled).
The RSCANnRPGACCr register is readable and writable when the RTME bit is set to 1.

25.4 Registers (CANFD Mode)

This section describes all registers to be used when the RS-CANFD is used in CANFD mode.

25.4.1 List of Registers

The following tables list RS-CANFD registers to be used in CANFD mode.
For details about <RSCFDn_base>, see section 25.1.2, Register Base Address.

- Table 25.176, Registers Initialized in Global Reset Mode or Channel Reset Mode
- Table 25.177, Registers Initialized Only in Global Reset Mode

Table 25.86 Registers

Module	Register	Symbol	Address
Interface mode-related register			
RSCFDn	Global interface mode select register	RSCFDnCFDGRMCFG	<RSCFDn_base> + $04FC_H$
Channel-related registers			
RSCFDn	Channel m nominal bit rate configuration register	RSCFDnCFDCmNCFG	<RSCFDn_base> + 0000_H + (10_H × m)
RSCFDn	Channel m control register	RSCFDnCFDCmCTR	<RSCFDn_base> + 0004_H + (10_H × m)
RSCFDn	Channel m status register	RSCFDnCFDCmSTS	<RSCFDn_base> + 0008_H + (10_H × m)
RSCFDn	Channel m error flag register	RSCFDnCFDCmERFL	<RSCFDn_base> + $000C_H$ + (10_H× m)
RSCFDn	Channel m data bit rate configuration register	RSCFDnCFDCmDCFG	<RSCFDn_base> + 0500_H + (20_H × m)
RSCFDn	Channel m CANFD configuration register	RSCFDnCFDCmFDCFG	<RSCFDn_base> + 0504_H + (20_H × m)
RSCFDn	Channel m CANFD control register	RSCFDnCFDCmFDCTR	<RSCFDn_base> + 0508_H + (20_H × m)
RSCFDn	Channel m CANFD status register	RSCFDnCFDCmFDSTS	<RSCFDn_base> + $050C_H$ + (20_H × m)
RSCFDn	Channel m CANFD CRC register	RSCFDnCFDCmFDCRC	<RSCFDn_base> + 0510_H + (20_H × m)
Global-related registers			
RSCFDn	Global configuration register	RSCFDnCFDGCFG	<RSCFDn_base> + 0084_H
RSCFDn	Global control register	RSCFDnCFDGCTR	<RSCFDn_base> + 0088_H
RSCFDn	Global status register	RSCFDnCFDGSTS	<RSCFDn_base> + $008C_H$
RSCFDn	Global error flag register	RSCFDnCFDGERFL	<RSCFDn_base> + 0090_H
RSCFDn	Global timestamp counter register	RSCFDnCFDGTSC	<RSCFDn_base> + 0094_H
RSCFDn	Global TX Interrupt Status Register 0	RSCFDnCFDGTINTSTS0	<RSCFDn_base> + 0460_H
RSCFDn	Global FD configuration register	RSCFDnCFDGFDCFG	<RSCFDn_base> + 0474_H
Receive rule-related registers			
RSCFDn	Receive Rule Entry Control Register	RSCFDnCFDGAFLECTR	<RSCFDn_base> + 0098_H
RSCFDn	Receive Rule Configuration Register 0	RSCFDnCFDGAFLCFG0	<RSCFDn_base> + $009C_H$
RSCFDn	Receive Rule ID Register j	RSCFDnCFDGAFLIDj	<RSCFDn_base> + 1000_H + (10_H × j)
RSCFDn	Receive Rule Mask Register j	RSCFDnCFDGAFLMj	<RSCFDn_base> + 1004_H + (10_H × j)
RSCFDn	Receive Rule Pointer 0 Register j	RSCFDnCFDGAFLP0_j	<RSCFDn_base> + 1008_H + (10_H × j)
RSCFDn	Receive Rule Pointer 1 Register j	RSCFDnCFDGAFLP1_j	<RSCFDn_base> + $100C_H$ + (10_H × j)
Receive buffer-related registers			
RSCFDn	Receive Buffer Number Register	RSCFDnCFDRMNB	<RSCFDn_base> + $00A4_H$
RSCFDn	Receive Buffer New Data Register y	RSCFDnCFDRMNDy	<RSCFDn_base> + $00A8_H$ + (04_H × y)
RSCFDn	Receive Buffer ID Register q	RSCFDnCFDRMIDq	<RSCFDn_base> + 2000_H + (20_H × q)
RSCFDn	Receive Buffer Pointer Register q	RSCFDnCFDRMPTRq	<RSCFDn_base> + 2004_H + (20_H × q)
RSCFDn	Receive Buffer CANFD status register q	RSCFDnCFDRMFDSTSq	<RSCFDn_base> + 2008_H + (20_H × q)
RSCFDn	Receive Buffer Data Field b Register q	RSCFDnCFDRMDFb_q	<RSCFDn_base> + $200C_H$ + (04_H × b) + (20_H × q)

Module	Register	Symbol	Address
Receive FIFO buffer-related registers			
RSCFDn	Receive FIFO Buffer Configuration and Control Register x	RSCFDnCFDRFCCx	<RSCFDn_base> + $00B8_H$ + (04_H × x)
RSCFDn	Receive FIFO Buffer Status Register x	RSCFDnCFDRFSTSx	<RSCFDn_base> + $00D8_H$ + (04_H × x)
RSCFDn	Receive FIFO Buffer Pointer Control Register x	RSCFDnCFDRFPCTRx	<RSCFDn_base> + $00F8_H$ + (04_H × x)
RSCFDn	Receive FIFO Buffer Access ID Register x	RSCFDnCFDRFIDx	<RSCFDn_base> + 3000_H + (80_H × x)
RSCFDn	Receive FIFO Buffer Access Pointer Register x	RSCFDnCFDRFPTRx	<RSCFDn_base> + 3004_H + (80_H × x)
RSCFDn	Receive FIFO CANFD status register x	RSCFDnCFDRFFDSTSx	<RSCFDn_base> + 3008_H + (80_H × x)
RSCFDn	Receive FIFO Buffer Access Data Field d Register x	RSCFDnCFDRFDFd_x	<RSCFDn_base> + $300C_H$ + (04_H × d) + (80_H × x)
Transmit/Receive FIFO buffer related registers			
RSCFDn	Transmit/receive FIFO Buffer Configuration and Control Register k	RSCFDnCFDCFCCk	<RSCFDn_base> + 0118_H + (04_H × k)
RSCFDn	Transmit/receive FIFO Buffer Status Register k	RSCFDnCFDCFSTSk	<RSCFDn_base> + 0178_H + (04_H × k)
RSCFDn	Transmit/receive FIFO Buffer Pointer Control Register k	RSCFDnCFDCFPCTRk	<RSCFDn_base> + $01D8_H$ + (04_H × k)
RSCFDn	Transmit/receive FIFO Buffer Access ID Register k	RSCFDnCFDCFIDk	<RSCFDn_base> + 3400_H + (80_H × k)
RSCFDn	Transmit/receive FIFO Buffer Access Pointer Register k	RSCFDnCFDCFPTRk	<RSCFDn_base> + 3404_H + (80_H × k)
RSCFDn	Transmit/receive FIFO CANFD configuration/status register k	RSCFDnCFDCFFDCSTSk	<RSCFDn_base> + 3408_H + (80_H × k)
RSCFDn	Transmit/receive FIFO Buffer Access Data Field d Register k	RSCFDnCFDCFDFd_k	<RSCFDn_base> + $340C_H$ + (04_H × d) + (80_H × k)
FIFO status-related registers			
RSCFDn	FIFO Empty Status Register	RSCFDnCFDFESTS	<RSCFDn_base> + 0238_H
RSCFDn	FIFO Full Status Register	RSCFDnCFDFFSTS	<RSCFDn_base> + $023C_H$
RSCFDn	FIFO Message Lost Status Register	RSCFDnCFDFMSTS	<RSCFDn_base> + 0240_H
RSCFDn	Receive FIFO Buffer Interrupt Flag Status Register	RSCFDnCFDRFISTS	<RSCFDn_base> + 0244_H
RSCFDn	Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register	RSCFDnCFDCFRISTS	<RSCFDn_base> + 0248_H
RSCFDn	Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register	RSCFDnCFDCFTISTS	<RSCFDn_base> + $024C_H$
FIFO DMA-related registers			
RSCFDn	DMA enable register	RSCFDnCFDCDTCT	<RSCFDn_base> + 0490_H
RSCFDn	DMA status register	RSCFDnCFDCDTSTS	<RSCFDn_base> + 0494_H
Transmit buffer-related registers			
RSCFDn	Transmit Buffer Control Register p	RSCFDnCFDTMCp	<RSCFDn_base> + 0250_H + (01_H × p)
RSCFDn	Transmit Buffer Status Register p	RSCFDnCFDTMSTSp	<RSCFDn_base> + $02D0_H$ + (01_H × p)
RSCFDn	Transmit Buffer ID Register p	RSCFDnCFDTMIDp	<RSCFDn_base> + 4000_H + (20_H × p)
RSCFDn	Transmit Buffer Pointer Register p	RSCFDnCFDTMPTRp	<RSCFDn_base> + 4004_H + (20_H × p)
RSCFDn	Transmit Buffer CANFD configuration register p	RSCFDnCFDTMFDCTRp	<RSCFDn_base> + 4008_H + (20_H × p)
RSCFDn	Transmit Buffer Data Field b Register p	RSCFDnCFDTMDFb_p	<RSCFDn_base> + $400C_H$ + (04_H × b) + (20H × p)
RSCFDn	Transmit Buffer Interrupt Enable Configuration Register y	RSCFDnCFDTMIECy	<RSCFDn_base> + 0390_H + (04_H × y)
Transmit buffer status-related registers			
RSCFDn	Transmit Buffer Transmit Request Status Register y	RSCFDnCFDTMTRSTSy	<RSCFDn_base> + 0350_H + (04_H × y)
RSCFDn	Transmit Buffer Transmit Abort Request Status Register y	RSCFDnCFDTMTARSTSy	<RSCFDn_base> + 0360_H + (04_H × y)
RSCFDn	Transmit Buffer Transmit Complete Status Register y	RSCFDnCFDTMTCSTSy	<RSCFDn_base> + 0370_H + (04_H × y)

Module	Register	Symbol	Address
RSCFDn	Transmit Buffer Transmit Abort Status Register y	RSCFDnCFDTMTASTSy	<RSCFDn_base> + 0380_H + (04_H × y)
Transmit queue-related registers			
RSCFDn	Transmit Queue Configuration and Control Register m	RSCFDnCFDTXQCCm	<RSCFDn_base> + $03A0_H$ + (04_H × m)
RSCFDn	Transmit Queue Status Register m	RSCFDnCFDTXQSTSm	<RSCFDn_base> + $03C0_H$ + (04_H × m)
RSCFDn	Transmit Queue Pointer Control Register m	RSCFDnCFDTXQPCTRm	<RSCFDn_base> + $03E0_H$ + (04_H × m)
Transmit history-related registers			
RSCFDn	Transmit History Configuration and Control Register m	RSCFDnCFDTHLCCm	<RSCFDn_base> + 0400_H + (04_H × m)
RSCFDn	Transmit History Status Register m	RSCFDnCFDTHLSTSm	<RSCFDn_base> + 0420_H + (04_H × m)
RSCFDn	Transmit History Pointer Control Register m	RSCFDnCFDTHLPCTRm	<RSCFDn_base> + 0440_H + (04_H × m)
RSCFDn	Transmit History Access Register m	RSCFDnCFDTHLACCm	<RSCFDn_base> + 6000_H + (04_H × m)
Test-related registers			
RSCFDn	Global Test Configuration Register	RSCFDnCFDGTSTCFG	<RSCFDn_base> + 0468_H
RSCFDn	Global Test Control Register	RSCFDnCFDGTSTCTR	<RSCFDn_base> + $046C_H$
RSCFDn	Global Lock Key Register	RSCFDnCFDGLOCKK	<RSCFDn_base> + $047C_H$
RSCFDn	RAM Test Page Access Register r	RSCFDnCFDRPGACCr	<RSCFDn_base> + 6400_H + (04_H × r)

Table 25.87 Transmit Buffer p Allocated to Each Channel

	CANm
Transmit buffer p	Transmit buffer 16 × m + 0
	Transmit buffer 16 × m + 1
	Transmit buffer 16 × m + 2
	Transmit buffer 16 × m + 3
	Transmit buffer 16 × m + 4
	Transmit buffer 16 × m + 5
	Transmit buffer 16 × m + 6
	Transmit buffer 16 × m + 7
	Transmit buffer 16 × m + 8
	Transmit buffer 16 × m + 9
	Transmit buffer 16 × m + 10
	Transmit buffer 16 × m + 11
	Transmit buffer 16 × m + 12
	Transmit buffer 16 × m + 13
	Transmit buffer 16 × m + 14
	Transmit buffer 16 × m + 15

Table 25.88 Transmit/Receive FIFO Buffer k Allocated to Each Channel

	CANm
Transmit buffer k	Transmit/receive FIFO buffer 3 × m + 0
	Transmit/receive FIFO buffer 3 × m + 1
	Transmit/receive FIFO buffer 3 × m + 2

Table 25.89 Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer by the Setting of Bits CFTML[3:0]

Setting of Bits CFTML[3:0]	Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer
0000_B	Transmit buffer 16 × m + 0
0001_B	Transmit buffer 16 × m + 1
0010_B	Transmit buffer 16 × m + 2
0011_B	Transmit buffer 16 × m + 3
0100_B	Transmit buffer 16 × m + 4
0101_B	Transmit buffer 16 × m + 5
0110_B	Transmit buffer 16 × m + 6
0111_B	Transmit buffer 16 × m + 7
1000_B	Transmit buffer 16 × m + 8
1001_B	Transmit buffer 16 × m + 9
1010_B	Transmit buffer 16 × m + 10
1011_B	Transmit buffer 16 × m + 11
1100_B	Transmit buffer 16 × m + 12
1101_B	Transmit buffer 16 × m + 13
1110_B	Transmit buffer 16 × m + 14
1111_B	Transmit buffer 16 × m + 15

Table 25.90 Transmit Buffer p Allocated to the Transmit Queue of Each Channel

Setting of Bits TXQDC[3:0]	Transmit Buffer p Allocated to the Transmit Queue
0000_B	Setting prohibited
0001_B	Setting prohibited
0010_B	Transmit buffer 16 × m + 15 to 16 × m + 13
0011_B	Transmit buffer 16 × m + 15 to 16 × m + 12
0100_B	Transmit buffer 16 × m + 15 to 16 × m + 11
0101_B	Transmit buffer 16 × m + 15 to 16 × m + 10
0110_B	Transmit buffer 16 × m + 15 to 16 × m + 9
0111_B	Transmit buffer 16 × m + 15 to 16 × m + 8
1000_B	Transmit buffer 16 × m + 15 to 16 × m + 7
1001_B	Transmit buffer 16 × m + 15 to 16 × m + 6
1010_B	Transmit buffer 16 × m + 15 to 16 × m + 5
1011_B	Transmit buffer 16 × m + 15 to 16 × m + 4
1100_B	Transmit buffer 16 × m + 15 to 16 × m + 3
1101_B	Transmit buffer 16 × m + 15 to 16 × m + 2
1110_B	Transmit buffer 16 × m + 15 to 16 × m + 1
1111_B	Transmit buffer 16 × m + 15 to 16 × m + 0

25.4.2 Details of Interface Mode-related Registers

25.4.2.1 RSCFDnCFDGRMCFG — Global Interface Mode Select Register

Access: RSCFDnCFDGRMCFG register can be read/written in 32-bit units
RSCFDnCFDGRMCFGL, RSCFDnCFDGRMCFGH registers can be read/written in 16-bit units
RSCFDnCFDGRMCFGLL, RSCFDnCFDGRMCFGLH, RSCFDnCFDGRMCFGHL, RSCFDnCFDGRMCFGHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGRMCFG: <RSCFDn_base> + $04FC_H$
RSCFDnCFDGRMCFGL: <RSCFDn_base> + $04FC_H$,
RSCFDnCFDGRMCFGH: <RSCFDn_base> + $04FE_H$
RSCFDnCFDGRMCFGLL: <RSCFDn_base> + $04FC_H$,
RSCFDnCFDGRMCFGLH: <RSCFDn_base> + $04FD_H$,
RSCFDnCFDGRMCFGHL: <RSCFDn_base> + $04FE_H$,
RSCFDnCFDGRMCFGHH: <RSCFDn_base> + $04FF_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RCMC
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 25.91 RSCFDnCFDGRMCFG Register Contents

Bit	Bit Name	Function
31 to 1	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
0	RCMC	Interface Mode Select 0: Classical CAN mode 1: CANFD mode

Note 1. The RSCANnGRMCFG register and the RSCFDnCFDGRMCFG register are the same register. Therefore, set either one of the registers.

Modify the RSCFDnCFDGRMCFG register only in global reset mode. Before setting other RS-CANFD registers, set this register.

RCMC Bit

Setting this bit to 0 places the RS-CANFD module in classical CAN mode. Setting this bit to 1 places the RS-CANFD module in CANFD mode. To switch classical CAN mode to CANFD mode, set the value after reset in all registers and bits allocated to the register map of classical CAN mode and then modify the RSCFDnCFDGRMCFG register.

25.4.3 Details of Channel-related Registers

25.4.3.1 RSCFDnCFDCmNCFG — Channel Nominal Bit Rate Configuration Register (m = 0, 1)

Access: RSCFDnCFDCmNCFG register can be read/written in 32-bit units
RSCFDnCFDCmNCFGL, RSCFDnCFDCmNCFGH registers can be read/written in 16-bit units
RSCFDnCFDCmNCFGLL, RSCFDnCFDCmNCFGLH, RSCFDnCFDCmNCFGHL, RSCFDnCFDCmNCFGHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmNCFG: <RSCFDn_base> + 0000_H + (10_H × m)
RSCFDnCFDCmNCFGL: <RSCFDn_base> + 0000_H + (10_H × m),
RSCFDnCFDCmNCFGH: <RSCFDn_base> + 0002_H + (10_H × m)
RSCFDnCFDCmNCFGLL: <RSCFDn_base> + 0000_H + (10_H × m),
RSCFDnCFDCmNCFGLH: <RSCFDn_base> + 0001_H + (10_H × m),
RSCFDnCFDCmNCFGHL: <RSCFDn_base> + 0002_H + (10_H × m),
RSCFDnCFDCmNCFGHH: <RSCFDn_base> + 0003_H + (10_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	NTSEG2[4:0]					—	NTSEG1[6:0]						
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NSJW[4:0]					—	NBRP[9:0]									
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.92 RSCFDnCFDCmNCFG Register Contents

Bit	Bit Name	Function
31 to 29	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
28 to 24	NTSEG2[4:0]	Nominal Bit Rate Time Segment 2 Control b28 b27 b26 b25 b24 0 0 0 0 0: Setting prohibited 0 0 0 0 1: 2 Tq : : 1 1 1 1 0: 31 Tq 1 1 1 1 1: 32 Tq
23	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
22 to 16	NTSEG1[6:0]	Nominal Bit Rate Time Segment 1 Control b22 b21 b20 b19 b18 b17 b16 0 0 0 0 0 0 0: Setting prohibited 0 0 0 0 0 0 1: Setting prohibited 0 0 0 0 0 1 0: Setting prohibited 0 0 0 0 0 1 1: 4 Tq : : 1 1 1 1 1 1 0: 127 Tq 1 1 1 1 1 1 1: 128 Tq
15 to 11	NSJW[4:0]	Nominal Bit Rate Resynchronization Jump Width Control b15 b14 b13 b12 b11 0 0 0 0 0: 1 Tq 0 0 0 0 1: 2 Tq 0 0 0 1 0: 3 Tq : : 1 1 1 1 0: 31 Tq 1 1 1 1 1: 32 Tq
10	Reserved	These bits are read as the value after reset. The write value should be the value after reset.

Bit	Bit Name	Function
9 to 0	NBRP[9:0]	Nominal Bit Rate Prescaler Division Ratio Setting When the set value = P (0 to 1023), the nominal bit rate prescaler divides fCAN by (P + 1).

Modify the RSCFDnCFDCmNCFG register in channel reset mode or channel halt mode. Set this register in channel reset mode, and then transition to channel communication mode or channel halt mode. For the description and settings of bit timing parameters, see section 25.11.1, Initial Settings.

NTSEG2[4:0] Bits

These bits are used to specify a Tq value for the length of phase segment 2 (PHASE_SEG2) of nominal bit rate.
Allowed values are 2 Tq to 32 Tq, inclusive.
Set a value smaller than the value of the NTSEG1[6:0] bits.

NTSEG1[6:0] Bits

These bits specify the total length of propagation segment (PROP_SEG) and phase segment 1 (PHASE_SEG1) of nominal bit rate as a Tq value.
A value of 4 to 128 Tq is settable.

NSJW[4:0] Bits

These bits specify the resynchronization jump width of nominal bit rate as a Tq value. A value of 1 to 32 Tq is settable.
Specify a value equal to or smaller than the NTSEG2[4:0] value.

NBRP[9:0] Bits

The clock obtained by dividing the CAN clock (fCAN) by the nominal bit rate prescaler ((NBRP[9:0]) + 1) becomes CANmTq(N) clock (fCANTQ(N)m). One clock of the CANmTq(N) clock becomes one Time Quantum (Tq).

25.4.3.2 RSCFDnCFDCmCTR — Channel Control Register (m = 0, 1)

Access: RSCFDnCFDCmCTR register can be read/written in 32-bit units
RSCFDnCFDCmCTRL, RSCFDnCFDCmCTRH registers can be read/written in 16-bit units
RSCFDnCFDCmCTRLL, RSCFDnCFDCmCTRLH, RSCFDnCFDCmCTRHL, RSCFDnCFDCmCTRHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmCTR: <RSCFDn_base> + 0004_H + (10_H × m)
RSCFDnCFDCmCTRL: <RSCFDn_base> + 0004_H + (10_H × m),
RSCFDnCFDCmCTRH: <RSCFDn_base> + 0006_H + (10_H × m)
RSCFDnCFDCmCTRLL: <RSCFDn_base> + 0004_H + (10_H × m),
RSCFDnCFDCmCTRLH: <RSCFDn_base> + 0005_H + (10_H × m),
RSCFDnCFDCmCTRHL: <RSCFDn_base> + 0006_H + (10_H × m),
RSCFDnCFDCmCTRHH: <RSCFDn_base> + 0007_H + (10_H × m)

Value after reset: $0000\ 0005_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ROM	CRCT	—	—	—	CTMS[1:0]		CTME	ERRD	BOM[1:0]		—	TDCVFIE	SOCOIE	EOCOIE	TAIE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE	—	—	—	—	RTBO	CSLPR	CHMDC[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 25.93 RSCFDnCFDCmCTR Register Contents

Bit	Bit Name	Function
31	ROM	Restricted Operation Mode Enable 0: Restricted operation mode is disabled. 1: Restricted operation mode is enabled.
30	CRCT	CRC Error Test Enable 0: The first bit of the reception ID field is not inverted. 1: The first bit of the reception ID field is inverted.
29 to 27	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
26, 25	CTMS[1:0]	Communication Test Mode Select b26 b25 0 0: Standard test mode 0 1: Listen-only mode 1 0: Self-test mode 0 (external loopback mode) 1 1: Self-test mode 1 (internal loopback mode)
24	CTME	Communication Test Mode Enable 0: Communication test mode is disabled. 1: Communication test mode is enabled.
23	ERRD	Error Display Mode Select 0: Error flags are displayed only for the first error information after bits 14 to 8 in the RSCFDnCFDCmERFL register are all cleared. 1: Error flags for all error information are displayed.
22, 21	BOM[1:0]	Bus Off Recovery Mode Select b22 b21 0 0: ISO11898-1 compliant 0 1: Entry to channel halt mode automatically at bus-off entry 1 0: Entry to channel halt mode automatically at bus-off end 1 1: Entry to channel halt mode (in bus-off state) by program request
20	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
19	TDCVFIE	Transmitter Delay Compensation Violation Interrupt Enable 0: A transmitter delay compensation violation interrupt is disabled. 1: A transmitter delay compensation violation interrupt is enabled.

Bit	Bit Name	Function
18	SOCOIE	Successful Occurrence Counter Overflow Interrupt Enable 0: A successful occurrence counter overflow interrupt is disabled. 1: A successful occurrence counter overflow interrupt is enabled.
17	EOCOIE	Error Occurrence Counter Overflow Interrupt Enable 0: An error occurrence counter overflow interrupt is disabled. 1: An error occurrence counter overflow interrupt is enabled.
16	TAIE	Transmit Abort Interrupt Enable 0: Transmit abort interrupt is disabled. 1: Transmit abort interrupt is enabled.
15	ALIE	Arbitration Lost Interrupt Enable 0: Arbitration lost interrupt is disabled. 1: Arbitration lost interrupt is enabled.
14	BLIE	Bus Lock Interrupt Enable 0: Bus lock interrupt is disabled. 1: Bus lock interrupt is enabled.
13	OLIE	Overload Frame Transmit Interrupt Enable 0: Overload frame transmit interrupt is disabled. 1: Overload frame transmit interrupt is enabled.
12	BORIE	Bus Off Recovery Interrupt Enable 0: Bus off recovery interrupt is disabled. 1: Bus off recovery interrupt is enabled.
11	BOEIE	Bus Off Entry Interrupt Enable 0: Bus off entry interrupt is disabled. 1: Bus off entry interrupt is enabled.
10	EPIE	Error Passive Interrupt Enable 0: Error passive interrupt is disabled. 1: Error passive interrupt is enabled.
9	EWIE	Error Warning Interrupt Enable 0: Error warning interrupt is disabled. 1: Error warning interrupt is enabled.
8	BEIE	Bus Error Interrupt Enable 0: Bus error interrupt is disabled. 1: Bus error interrupt is enabled.
7 to 4	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
3	RTBO	Forcible Return from Bus-off When this bit is set to 1, forcible return from the bus off state is made. This bit is always read as 0.
2	CSLPR	Channel Stop Mode 0: Other than channel stop mode 1: Channel stop mode
1, 0	CHMDC[1:0]	Mode Select b1 b0 0 0: Channel communication mode 0 1: Channel reset mode 1 0: Channel halt mode 1 1: Setting prohibited

ROM Bit

When the ROM bit and the CTME bit in the RSCFDnCFDCmCTR register are set to 1, restricted operation mode is enabled. Use the restricted operation mode only when the CTMS[1:0] value in the RSCFDnCFDCmCTR register is 00_B (standard test mode). Modify this bit only in channel halt mode. This bit is always 0 in channel reset mode.

CRCT Bit

This bit is used to test the CRC generation circuit in the RS-CANFD module. Setting this bit to 1 inverts the first bit of the ID field when a message is received. With this inversion of bit, the CRC calculation result does not match the normal CRC value of the received frame, which can detect a CRC error (the CERR bit in the RSCFDnCFDCmERFL register is 1). When using this function, note the following.

- This function is available while the CTME bit in the RSCFDnCFDCmCTR register is 1 (communication test mode enabled).
- This function cannot communicate with other CAN nodes. Use this function for inter-channel communication test (the CmICBCE bit in the RSCFDnCFDGTSTCFG register is 1).
- Bit inversion in the ID field may cause bit stuffing rule violation. In that case, no CRC error is detected but a stuff error is detected.

Modify this bit only in channel halt mode. This bit is always 0 in channel reset mode.

CTMS[1:0] Bits

These bits are used to select a communication test mode. Modify these bits in channel halt mode only. These bits are set to 0 in channel reset mode.

CTME Bit

Setting this bit to 1 enables communication test mode. Modify these bits in channel halt mode. This bit is set to 0 in channel reset mode.

ERRD Bit

This bit is used to control the display mode of bits 14 to 8 in the RSCFDnCFDCmERFL register.
When this bit is clear to 0, if any error is detected while the flags of bits 14-8 in the RSCFDnCFDCmERFL register are all 0, only the flags of the first error event are set to 1. If two or more errors occur in the first error event, all the flags of the detected errors are set to 1.
When this bit is set to 1, all the flags of errors that have occurred are set to 1 regardless of the error occurrence order.
Modify this bit only in channel reset mode or channel halt mode.

BOM[1:0] Bits

These bits are used to select the bus off recovery mode of the RS-CANFD module.
When the BOM[1:0] bits are set to 00_B, return from the bus off state to the error active state is compliant with the CAN specifications. That is, the RS-CANFD module reenters the CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. A bus off recovery interrupt request is generated at the time of return from the bus off state. Even if the CHMDC[1:0] bits are set to 10_B (channel halt mode) before recessive bits are detected 128 times, the RS-CANFD module does not transition to channel halt mode until recessive bits are detected 128 times.
When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to 01_B, the CHMDC[1:0] bits in the RSCFDnCFDCmCTR register (m = 0, 1) are set to 10_B and the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated and the TEC[7:0] and REC[7:0] bits in the RSCFDnCFDCmSTS register are cleared to 00_H.
When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to 10_B, the CHMDC[1:0] bits are set to 10_B and the RS-CANFD module transitions to channel halt mode after return from the bus off state (11 consecutive recessive bits are detected 128 times). A bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00_H.
When the BOM[1:0] bits are set to 11_B and the CHMDC[1:0] bits are set to 10_B while the RS-CANFD module is in the bus off state, the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00_H. However, if 11 consecutive recessive bits are detected 128 times and the RS-CANFD module has recovered to the error active state from the bus off state before the CHMDC[1:0] bits are set to 10_B, a bus off recovery interrupt request is generated.
If a program writes to the CHMDC[1:0] bit at the same time as the RS-CANFD module transition to channel halt mode (at bus off entry when the BOM[1:0] bits are 01B or at bus off end when the BOM[1:0] bits are 10_B), the program's writing takes precedence. Modify the BOM[1:0] bits only in channel reset mode.

TDCVFIE Bit

When the TDCVF flag in the RSCFDnCFDCmFDSTS register is set to 1 after the TDCVFIE bit is set to 1, an interrupt request occurs. Modify this bit only in channel reset mode.

SOCOIE Bit

When the SOCO flag in the RSCFDnCFDCmFDSTS register is set to 1 after the SOCOIE bit is set to 1, an interrupt request occurs. Modify this bit only in channel reset mode.

EOCOIE Bit

When the EOCO flag in the RSCFDnCFDCmFDSTS register is set to 1 after the EOCOIE bit is set to 1, an interrupt request occurs. Modify this bit only in channel reset mode.

TAIE Bit

When transmit abort of the transmit buffer is completed with the TAIE bit set to 1, an interrupt request is generated. Modify this bit only in channel reset mode.

ALIE Bit

When the ALF flag in the RSCFDnCFDCmERFL register is set to 1 with the ALIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BLIE Bit

When the BLF flag in the RSCFDnCFDCmERFL register is set to 1 with the BLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

OLIE Bit

When the OVLF flag in the RSCFDnCFDCmERFL register is set to 1 with the OLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BORIE Bit

When the BORF flag in the RSCFDnCFDCmERFL register is set to 1 with the BORIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BOEIE Bit

When the BOEF flag in the RSCFDnCFDCmERFL register is set to 1 with the BOEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EPIE Bit

When the EPF flag in the RSCFDnCFDCmERFL register is set to 1 with the EPIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EWIE Bit

When the EWF flag in the RSCFDnCFDCmERFL register is set to 1 with the EWIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BEIE Bit

When the BEF flag in the RSCFDnCFDCmERFL register is set to 1 with the BEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

RTBO Bit

Setting this bit to 1 in the bus off state forcibly returns the state from the bus off state to the error active state. This bit is automatically cleared to 0. Setting this bit to 1 clears the TEC[7:0] and REC[7:0] bits in the RSCFDnCFDCmSTS register to 00_H and also clears the BOSTS flag in the RSCFDnCFDCmSTS register to 0 (not in bus off state). The other registers remain unchanged. No bus off recovery interrupt request is generated upon return from the bus off state in this case. Use this bit only when the BOM[1:0] bits in the RSCFDnCFDCmCTR register are 00_B (ISO11898-1 compliant). A delay of up to 1 CAN bit time occurs after the RTBO bit is set to 1 until the RSCAN module transitions to the error active state. Set this bit to 1 in channel communication mode.

CSLPR Bit

Setting this bit to 1 places the channel into channel stop mode.
Clearing this bit to 0 makes the channel exit channel stop mode.
This bit should not be modified in channel communication mode or channel halt mode.

CHMDC[1:0] Bits

These bits are used to select a channel mode (channel communication mode, channel reset mode, or channel halt mode). For details, see Figure 25.5. Setting the CSLPR bit to 1 in channel reset mode allows transition to channel stop mode. Do not set the CHMDC[1:0] bits to 11_B. When the CAN module has automatically transitioned to channel halt mode based on the setting of the BOM[1:0] bits, the CHMDC[1:0] bits automatically become 10_B.

25.4.3.3 RSCFDnCFDCmSTS — Channel Status Register (m = 0, 1)

Access: RSCFDnCFDCmSTS register can be read/written in 32-bit units
RSCFDnCFDCmSTSL, RSCFDnCFDCmSTSH registers can be read/written in 16-bit units
RSCFDnCFDCmSTSLL, RSCFDnCFDCmSTSLH, RSCFDnCFDCmSTSHL, RSCFDnCFDCmSTSHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmSTS: <RSCFDn_base> + 0008_H + (10_H × m)
RSCFDnCFDCmSTSL: <RSCFDn_base> + 0008_H + (10_H × m),
RSCFDnCFDCmSTSH: <RSCFDn_base> + $000A_H$ + (10_H × m)
RSCFDnCFDCmSTSLL: <RSCFDn_base> + 0008_H + (10_H × m),
RSCFDnCFDCmSTSLH: <RSCFDn_base> + 0009_H + (10_H × m),
RSCFDnCFDCmSTSHL: <RSCFDn_base> + $000A_H$ + (10_H × m),
RSCFDnCFDCmSTSHH: <RSCFDn_base> + $000B_H$ + (10_H × m)

Value after reset: $0000\ 0005_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEC[7:0]								REC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ESIF	COMSTS	RECSTS	TRMSTS	BOSTS	EPSTS	CSLPSTS	CHLTSTS	CRSTSTS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W:	R	R	R	R	R	R	R	R/W *1	R	R	R	R	R	R	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.94 RSCFDnCFDCmSTS Register Contents

Bit	Bit Name	Function
31 to 24	TEC[7:0]	The transmit error counter (TEC) can be read.
23 to 16	REC[7:0]	The receive error counter (REC) can be read.
15 to 9	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
8	ESIF	Error State Indication Flag 0: No CANFD message whose ESI bit is recessive has been received. 1: At least one CANFD message whose ESI bit is recessive has been received.
7	COMSTS	Communication Status Flag 0: Communication is not ready. 1: Communication is ready.
6	RECSTS	Receive Status Flag 0: Bus idle, in transmission or bus off state 1: In reception
5	TRMSTS	Transmit Status Flag 0: Bus idle or in reception 1: In transmission or bus off state
4	BOSTS	Bus Off Status Flag 0: Not in bus off state 1: In bus off state
3	EPSTS	Error Passive Status Flag 0: Not in error passive state 1: In error passive state
2	CSLPSTS	Channel Stop Status Flag 0: Not in channel stop mode 1: In channel stop mode
1	CHLTSTS	Channel Halt Status Flag 0: Not in channel halt mode 1: In channel halt mode

Bit	Bit Name	Function
0	CRSTSTS	Channel Reset Status Flag 0: Not in channel reset mode 1: In channel reset mode

TEC[7:0] Bits

These bits contain the transmit error counter value. For transmit error counter increment/decrement conditions, see the CAN specification (ISO11898-1).
These bits are cleared to 0 in channel reset mode.

REC[7:0] Bits

These bits contain the receive error counter value. For receive error counter increment/decrement conditions, see the CAN specifications (ISO11898-1).
These bits are cleared to 0 in channel reset mode.

ESIF Flag

When the recessive ESI bit is detected in a successfully received message, this flag is set to 1. In loopback mode or mirror mode, the own transmission message is regarded as a received message. To clear this flag to 0, write 0 to this bit by the program. This bit cannot be set to 1 by the program. If the flag setting (to 1) timing matches the writing 0 (by the program) timing, this flag is set to 1.
This flag is 0 in channel reset mode.

COMSTS Flag

This bit indicates that communication is ready.
This flag becomes 1 when the CAN module has detected 11 consecutive recessive bits after it has transitioned from channel reset mode or channel halt mode to channel communication mode. This flag is cleared to 0 in channel reset mode or channel halt mode.

RECSTS Flag

This flag is set to 1 when reception has started, and is cleared to 0 when the bus has become idle or transmission has started.

TRMSTS Flag

This flag is set to 1 when transmission has started, and is cleared to 0 when the bus has become idle or reception has started. This flag remains 1 in the bus off state.

BOSTS Flag

This flag is set to 1 when the bus off state (TEC[7:0] > 255) is entered. It is cleared to 0 when the CAN module has exited the bus off state.

EPSTS Flag

This flag is set to 1 when the RS-CANFD module has entered the error passive state ((128 ≤ TEC[7:0] ≤ 255) or (128 ≤ REC[7:0])), It is cleared to 0 when the RS-CANFD module has exited the error passive state or has entered channel reset mode.

CSLPSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel stop mode, and is cleared to 0 when the CAN module has returned from channel stop mode.

CHLTSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel halt mode, and is cleared to 0 when the CAN module has returned from channel halt mode.

CRSTSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel reset mode, and is cleared to 0 when the CAN module has transitioned to channel communication mode or channel halt mode. This flag remains 1 when the CAN module transitions from channel reset mode to channel stop mode.

25.4.3.4 RSCFDnCFDCmERFL — Channel Error Flag Register (m = 0, 1)

Access: RSCFDnCFDCmERFL register can be read/written in 32-bit units
RSCFDnCFDCmERFLL, RSCFDnCFDCmERFLH registers can be read/written in 16-bit units
RSCFDnCFDCmERFLLL, RSCFDnCFDCmERFLLH, RSCFDnCFDCmERFLHL, RSCFDnCFDCmERFLHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmERFL: <RSCFDn_base> + $000C_H$ + (10_H × m)
RSCFDnCFDCmERFLL: <RSCFDn_base> + $000C_H$ + (10_H × m),
RSCFDnCFDCmERFLH: <RSCFDn_base> + $000E_H$ + (10_H × m)
RSCFDnCFDCmERFLLL: <RSCFDn_base> + $000C_H$ + (10_H × m),
RSCFDnCFDCmERFLLH: <RSCFDn_base> + $000D_H$ + (10_H × m),
RSCFDnCFDCmERFLHL: <RSCFDn_base> + $000E_H$ + (10_H × m),
RSCFDnCFDCmERFLHH: <RSCFDn_base> + $000F_H$ + (10_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CRCREG[14:0]														
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR	ALF	BLF	OVLF	BORF	BOEF	EPF	EWF	BEF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.95 RSCFDnCFDCmERFL Register Contents

Bit	Bit Name	Function
31	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
30 to 16	CRCREG [14:0]	CRC Calculation Data (CRC length:15 bits) A CRC value calculated based on the transmit message or receive message is indicated.
15	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
14	ADERR	ACK Delimiter Error Flag 0: No ACK delimiter error is detected. 1: ACK delimiter error is detected.
13	B0ERR	Dominant Bit Error Flag 0: No dominant bit error is detected. 1: Dominant bit error is detected.
12	B1ERR	Recessive Bit Error Flag 0: No recessive bit error is detected. 1: Recessive bit error is detected.
11	CERR	CRC Error Flag 0: No CRC error is detected. 1: CRC error is detected.
10	AERR	ACK Error Flag 0: No ACK error is detected. 1: ACK error is detected.
9	FERR	Form Error Flag 0: No form error is detected. 1: Form error is detected.
8	SERR	Stuff Error Flag 0: No stuff error is detected. 1: Stuff error is detected.
7	ALF	Arbitration-lost Flag 0: No arbitration-lost is detected. 1: Arbitration-lost is detected.

Bit	Bit Name	Function
6	BLF	Bus Lock Flag 0: No channel bus is detected. 1: Channel bus is detected.
5	OVLF	Overload Flag 0: No overload is detected. 1: Overload is detected.
4	BORF	Bus Off Recovery Flag 0: No bus off recovery is detected. 1: Bus off recovery is detected.
3	BOEF	Bus Off Entry Flag 0: No bus off entry is detected. 1: Bus off entry is detected.
2	EPF	Error Passive Flag 0: No error passive is detected. 1: Error passive is detected.
1	EWF	Error Warning Flag 0: No error warning is detected. 1: Error warning is detected.
0	BEF	Bus Error Flag 0: No channel bus error is detected. 1: Channel bus error is detected.

See the CAN specification (ISO11898-1) for a description of error occurrence conditions. To clear each flag of this register, the program must write a 0 to the corresponding bit. These flags cannot be set to 1 by the program. If any of these flags is set to 0 at the same time that the program writes 0 to the flag, the flag is still set to 1. The channel reset mode transition clears all of these flags to 0.
If the ERRD bit in the RSCFDnCFDCmCTR register is set to 0 (ie, only the flags for the first error event are displayed) and an error related to bits 14 to 8 of RSCFDnCFDCmERFL is detected, the flag bits are only set by the error event if bits 14 to 8 were all 0 at the when time the error occurred.

CRCREG[14:0] Flag

When the CTME bit in the RSCFDnCFDCmCTR register is set to 1 (communication test mode is enabled), if transmit or receive message is a classical CAN frame (CRC length = 15 bits), this flag is updated and the CRC value calculated based on the message can be read. When a CANFD frame is sent or received, the value of CRCREG[20:0] bits in the RSCFDnCFDCmFDCRC register is updated. When the CTME bit is set to 0 (communication test mode is disabled), these bits are always read as 0.

ADERR Flag

This flag is set to 1 when a form error has been detected in the ACK delimiter during transmission.

B0ERR Flag

This flag is set to 1 when a recessive bit has been detected though a dominant bit was transmitted.

B1ERR Flag

This flag is set to 1 when a dominant bit has been detected though a recessive bit was transmitted.

CERR Flag

This flag is set to 1 when a CRC error has been detected.

AERR Flag

This flag is set to 1 when an ACK error has been detected.

FERR Flag

This flag is set to 1 when a form error has been detected.

SERR Flag

This flag is set to 1 when a stuff error has been detected.

ALF Flag

This flag is set to 1 when an arbitration-lost has been detected.

BLF Flag

This flag is set to 1 when 32 consecutive dominant bits have been detected on the CAN bus in channel communication mode. After that, detection of a dominant lock is restarted when either of the following conditions is met.

- A recessive bit is detected after the BLF bit has been cleared from 1 to 0.
- The CAN module transitions to channel reset mode and returns to channel communication mode after the BLF bit has been cleared from 1 to 0.

OVLF Flag

This flag is set to 1 when the overload frame transmit condition has been detected when performing reception or transmission.

BORF Flag

This flag is set to 1 when 11 consecutive recessive bits have been detected 128 times and the CAN module returns from the bus off state. However, this flag is not set to 1 if the CAN module returns from the bus off state in any of the following ways before 11 consecutive recessive bits are detected 128 times.

- The CHMDC[1:0] bits in the RSCFDnCFDCmCTR register are set to 01_B (channel reset mode).
- The RTBO bit in the RSCFDnCFDCmCTR register is set to 1 (forcible return from the bus off state is made).
- The BOM[1:0] bits in the RSCFDnCFDCmCTR register are set to 01_B (transition to channel halt mode at bus off entry).
- The CHMDC[1:0] bits in the RSCFDnCFDCmCTR register are set to 10_B (channel halt mode) before 11 consecutive recessive bits are detected 128 times with the BOM[1:0] bits set to 11_B (transition to channel halt mode upon a request from the program during bus off).

BOEF Flag

This flag is set to 1 when the bus off state is reached (TEC[7:0] value > 255). This flag is also set to 1 if the bus off state is reached when the BOM[1:0] bits in the RSCFDnCFDCmCTR register (m = 0, 1) set to 01_B (transition to channel halt mode at bus off entry).

EPF Flag

This flag becomes 1 when the error passive state is reached (REC[7:0] or TEC[7:0] value > 127).
This flag becomes 1 only when the REC[7:0] or TEC[7:0] value first exceeds 127. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 127, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 127 or less and then the REC[7:0] or TEC[7:0] value exceeds 127 again.

EWF Flag

This flag is set to 1 only when the REC[7:0] or TEC[7:0] value first exceeds 95. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 95, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 95 or less and then the REC[7:0] or TEC[7:0] value exceeds 95 again.

BEF Flag

This flag is set to 1 when any one of the ADERR, B0ERR, B1ERR, CERR, AERR, FERR, and SERR flags in the RSCFDnCFDCmERFL register is set to 1.

Note: To clear the flag of this register to 0, use a store instruction to write “0” to the given flag and “1” to the other flags.

25.4.3.5 RSCFDnCFDCmDCFG — Channel Data Bit Rate Configuration Register (m = 0, 1)

Access: RSCFDnCFDCmDCFG register can be read/written in 32-bit units
RSCFDnCFDCmDCFGL, RSCFDnCFDCmDCFGH registers can be read/written in 16-bit units
RSCFDnCFDCmDCFGLL, RSCFDnCFDCmDCFGLH, RSCFDnCFDCmDCFGHL, RSCFDnCFDCmDCFGHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmDCFG: <RSCFDn_base> + 0500_H + (20_H × m)
RSCFDnCFDCmDCFGL: <RSCFDn_base> + 0500_H + (20_H × m),
RSCFDnCFDCmDCFGH: <RSCFDn_base> + 0502_H + (20_H × m)
RSCFDnCFDCmDCFGLL: <RSCFDn_base> + 0500_H + (20_H × m),
RSCFDnCFDCmDCFGLH: <RSCFDn_base> + 0501_H + (20_H × m),
RSCFDnCFDCmDCFGHL: <RSCFDn_base> + 0502_H + (20_H × m),
RSCFDnCFDCmDCFGHH: <RSCFDn_base> + 0503_H + (20_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	DSJW[2:0]			—	DTSEG2[2:0]			DTSEG1[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DBRP[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.96 RSCFDnCFDCmDCFG Register Contents

Bit	Bit Name	Function
31 to 27	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
26 to 24	DSJW[2:0]	Data Bit Rate Resynchronization Jump Width Control b26 b25 b24 0 0 0: 1 Tq 0 0 1: 2 Tq 0 1 0: 3 Tq 0 1 1: 4 Tq 1 0 0: 5 Tq 1 0 1: 6 Tq 1 1 0: 7 Tq 1 1 1: 8 Tq
23	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
22 to 20	DTSEG2[2:0]	Data Bit Rate Time Segment 2 Control b22 b21 b20 0 0 0: Setting prohibited 0 0 1: 2 Tq 0 1 0: 3 Tq 0 1 1: 4 Tq 1 0 0: 5 Tq 1 0 1: 6 Tq 1 1 0: 7 Tq 1 1 1: 8 Tq

Bit	Bit Name	Function
19 to 16	DTSEG1[3:0]	Data Bit Rate Time Segment 1 Control b19 b18 b17 b16 0 0 0 0: Setting prohibited 0 0 0 1: 2 Tq 0 0 1 0: 3 Tq 0 0 1 1: 4 Tq 0 1 0 0: 5 Tq 0 1 0 1: 6 Tq 0 1 1 0: 7 Tq 0 1 1 1: 8 Tq 1 0 0 0: 9 Tq 1 0 0 1: 10 Tq 1 0 1 0: 11 Tq 1 0 1 1: 12 Tq 1 1 0 0: 13 Tq 1 1 0 1: 14 Tq 1 1 1 0: 15 Tq 1 1 1 1: 16 Tq
15 to 8	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
7 to 0	DBRP[7:0]	Data Bit Rate Prescaler Division Ratio Setting When the set value = P (0 to 255), the data bit rate prescaler divides fCAN by (P + 1).

Modify the RSCFDnCFDCmDCFG register in channel reset mode or channel halt mode. Set this register in channel reset mode, and then transition to channel communication mode or channel halt mode. When only classical CAN frames are used in CANFD mode, set the RSCFDnCFDCmDCFG register to the value equal to the set RSCFDnCFDCmNCFG register value. For the description and settings of bit timing parameters, see section 25.11.1, Initial Settings.

DSJW[2:0] Bits

These bits specify the resynchronization jump width of data bit rate as a Tq value. A value of 1 to 8 Tq is settable. Specify a value equal to or smaller than the DTSEG2[2:0] bits value.

DTSEG2[2:0] Bits

These bits are used to specify a Tq value for the length of phase segment 2 (PHASE_SEG2) of nominal bit rate. Allowed values are 2 Tq to 8 Tq, inclusive.
Set a value smaller than the value of the DTSEG1[3:0] bits.

DTSEG1[3:0] Bits

These bits specify the total length of propagation segment (PROP_SEG) and phase segment 1 (PHASE_SEG1) of data bit rate as a Tq value.
A value of 2 to 16 Tq is settable.

DBRP[7:0] Bits

The clock obtained by dividing the CAN clock (fCAN) by the data bit rate prescaler ((DBRP[7:0]) + 1) becomes CANmTq(D) clock (fCANTQ(D)m). One clock of the CANmTq(D) clock becomes one Time Quantum (Tq).
Make sure to set the same value in the NBRP[9:0] bits and the DBRP[7:0] bits. To set the nominal bit rate and data bit rate of different values, change the NTSEG 1 bit and NTSEG 2 bit in the RSCFDnCFDCmNCFG register and the DTSEG 1 bit and DTSEG 2 bit in the RSCFDnCFDCmDCFG register to the desired bit rate values respectively. When the TDCE bit of the RSCFDnCFDCmFDCFG register is 1 (transmission delay correction enabled), set the same value of 1 or less to the NBRP [9:0] and DBRP [7:0] bits.

25.4.3.6 RSCFDnCFDCmFDCFG — Channel CANFD Configuration Register (m = 0, 1)

Access: RSCFDnCFDCmFDCFG register can be read/written in 32-bit units
RSCFDnCFDCmFDCFGL, RSCFDnCFDCmFDCFGH registers can be read/written in 16-bit units
RSCFDnCFDCmFDCFGLL, RSCFDnCFDCmFDCFGLH, RSCFDnCFDCmFDCFGHL, RSCFDnCFDCmFDCFGHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmFDCFG: <RSCFDn_base> + 0504_H + (20_H × m)
RSCFDnCFDCmFDCFGL: <RSCFDn_base> + 0504_H + (20_H × m),
RSCFDnCFDCmFDCFGH: <RSCFDn_base> + 0506_H + (20_H × m)
RSCFDnCFDCmFDCFGLL: <RSCFDn_base> + 0504_H + (20_H × m),
RSCFDnCFDCmFDCFGLH: <RSCFDn_base> + 0505_H + (20_H × m),
RSCFDnCFDCmFDCFGHL: <RSCFDn_base> + 0506_H + (20_H × m),
RSCFDnCFDCmFDCFGHH: <RSCFDn_base> + 0507_H + (20_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	REFE	FDOE	TMME	GWBRS	GWFDF	GWEN	—	TDCO[6:0]						
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ESIC	TDCE	TDCOC	—	—	—	—	—	EOCCFG[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.97 RSCFDnCFDCmFDCFG Register Contents

Bit	Bit Name	Function
31, 30	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
29	REFE	Reception data edge filter enable bit 0: Reception data edge filter disabled 1: Reception data edge filter enabled
28	FDOE	FD-only mode enable bit 0: FD-only mode disabled 1: FD-only mode enabled
27	TMME	Transmit Buffer Merge Mode Enable 0: Transmit buffer merge mode is disabled. 1: Transmit buffer merge mode is enabled.
26	GWBRS	Gateway BRS 0: A frame is transmitted with the BRS bit in the received frame set to 0. 1: A frame is transmitted with the BRS bit in the received frame set to 1.
25	GWFDF	Gateway FDF 0: A frame is transmitted regarding the received frame as a classical CAN frame. 1: A frame is transmitted regarding the received frame as a CANFD frame.
24	GWEN	CAN-CANFD Gateway Enable 0: The CAN-CANFD gateway is disabled. 1: The CAN-CANFD gateway is enabled.
23	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
22 to 16	TDCO[6:0]	Transmitter Delay Compensation Offset These bits are set to the transmitter delay compensation offset value.
15 to 11	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
10	ESIC	Error State Display Mode Select 0: Always displays the node error state. 1: When the node is not in the error passive state: Displays the message buffer error state. When the node is in the error passive state: Displays the node error state.

Bit	Bit Name	Function
9	TDCE	Transmitter Delay Compensation Enable 0: Transmitter delay compensation is disabled. 1: Transmitter delay compensation is enabled.
8	TDCOC	Transmitter Delay Compensation Measurement Select 0: Measurement and offset 1: Only offset
7 to 3	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
2 to 0	EOCCFG [2:0]	Error Occurrence Counting Method Select b2 b1 b0 0 0 0: All transmit messages and receive messages 0 0 1: All transmit messages 0 1 0: All receive messages 0 1 1: Setting prohibited 1 0 0: Only data phase of transmitted or received CANFD message 1 0 1: Only data phase of transmitted CANFD message 1 1 0: Only data phase of received CANFD message 1 1 1: Setting prohibited

REFE bit

Setting this bit to 1 enables reception data edge filtering when the idle condition is detected, and a dominant level with less than 2 time quanta is ignored. A dominant level with more than or equal to 2 time quanta is detected as an edge. Modify this bit only in channel reset mode.

FDOE bit

Setting this bit to 1 enables FD-only mode. When data is transmitted, a CANFD frame will be sent regardless of the settings to the CFFDF bit in the RSCFDnCFDCFFDCSTSk register or the TMFDF bit in the RSCFDnCFDTMFDCTRp register. When a Classical CAN frame is received, a form error is detected. Modify this bit only in channel reset mode.

TMME Bit

Setting this bit to 1 enables transmit buffer merge mode. Modify this bit only in channel reset mode or channel halt mode.

GWBRS Bit

When the GWEN bit is 1, the BRS bit in a CANFD frame to be transmitted by the gateway function is set. Write 0 to this bit to clear the GWBRS bit to 0. Modify this bit only in channel reset mode.

GWFDF Bit

When the GWEN bit is 1, the FDF bit in a CANFD frame to be transmitted by the gateway function is set. Modify this bit only in channel reset mode.

GWEN Bit

This bit is used to control the operation of the transmit/receive FIFO buffer with the CFM[1:0] bits in the RSCFDnCFDCFCCk register set to 10_B (gateway mode).
Setting this bit to 1 enables the CAN-CANFD gateway, enabling transmission in a format different from that of frames received by the gateway function. Received frames are replaced in accordance with the settings of the GWFDF bit and the GWBRS bit. When the DLC value in the received classical CAN frame is 1001_B or more and the GWFDF bit is 1 (CANFD frame), the DLC value is replaced with 1000_B.
While this bit is 1, do not perform routing the following frames by using the gateway function.

- CANFD frames with a payload length of more than 8 bytes
- Remote frames

While this bit is 1, the following frame should be transmitted in the channel by setting of GWFDF.

- When GWFDF bit is set to 0, only classical CAN frames should be transmitted.

- When GWFDF bit is set to 1, only CANFD frames should be transmitted.

Table 25.98 shows the settings and formats of transmit frame and receive frame while the CAN-CANFD gateway is enabled.

Table 25.98 Operation when the CAN-CANFD Gateway is Enabled

Receive Frame				Transmit Frame		
Format	BRS Bit	Received DLC Value	GWFDF Bit	Format	BRS Bit	DLC Value to be Transmitted
Classical CAN	None	DLC ≤ 1000_B	0	Classical CAN	None	Not replaced
		DLC > 1000_B				
CANFD	Arbitrary	DLC ≤ 1000_B				
Classical CAN	None	DLC ≤ 1000_B	1	CANFD	According to GWBRS bit setting	Not replaced
		DLC > 1000_B				Replaced with 1000_B
CANFD	Arbitrary	DLC ≤ 1000_B				Not replaced

TDCO[6:0] Bits

These bits are set to the SSP offset value. How to use this value depends on the TDCOC bit in the RSCFDnCFDCmFDCFG register.

When the TDCOC bit is 0, the transmitter delay compensation result equals to the total value of the measured delay value and the TDCO[6:0] value (rounded to the nearest integer Tq).

When the TDCOC bit is 1, the transmitter delay compensation result equals to the TDCO[6:0] value.

The SSP offset value = (set value of TDCO[6:0] bits + 1).

Modify these bits only in channel reset mode or channel halt mode.

ESIC Bit

When the ESIC bit is set to 1, if the channel is in the error active state, the ESI bit value (CFESI bit in the RSCFDnCFDCFFDCSTSk register or TMESI bit in the RSCFDnCFDTMFDCTRp register) set in the transmit/receive FIFO buffer or transmit buffer is transmitted as an ESI bit value of the transmit message. When the channel is in the error passive state or the ESIC bit is 0, the channel status is transmitted as an ESI bit value. Modify this bit only in channel reset mode or channel halt mode.

Table 25.99 **ESI Value to Be Transmitted**

ESIC Bit	Channel Status	ESI Value to be Transmitted
0	Error active	0 (error active node)
	Error passive	1 (error passive node)
1	Error active	ESI value set in the transmit/receive FIFO buffer or transmit buffer CFESI bit in the RSCFDnCFDCFFDCSTSk register or TMESI bit in the RSCFDnCFDTMFDCTRp register)
	Error passive	1 (error passive node)

TDCE Bit

Setting this bit to 1 enables transmitter delay compensation. Modify this bit only in channel reset mode or channel halt mode.

TDCOC Bit

When this bit is 0, the SSP position is defined by the total of the measured delay value and the SSP offset value (fixed value).
When this bit is 1, the SSP position is defined only by the SSP offset value.
Modify this bit only in channel reset mode or channel halt mode.

EOCCFG[2:0] Bits

These bits are used to select a frame format and a transmission/reception direction when the error occurrence counter counts CAN bus errors.
Modify these bits only in channel reset mode or channel halt mode.

25.4.3.7 RSCFDnCFDCmFDCTR — Channel CANFD Control Register (m = 0, 1)

Access: RSCFDnCFDCmFDCTR register can be read/written in 32-bit units
RSCFDnCFDCmFDCTRL, RSCFDnCFDCmFDCTRH registers can be read/written in 16-bit units
RSCFDnCFDCmFDCTRLL, RSCFDnCFDCmFDCTRLH, RSCFDnCFDCmFDCTRHL, RSCFDnCFDCmFDCTRHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmFDCTR: <RSCFDn_base> + 0508_H + (20_H × m)
RSCFDnCFDCmFDCTRL: <RSCFDn_base> + 0508_H + (20_H × m),
RSCFDnCFDCmFDCTRH: <RSCFDn_base> + $050A_H$ + (20_H × m)
RSCFDnCFDCmFDCTRLL: <RSCFDn_base> + 0508_H + (20_H × m),
RSCFDnCFDCmFDCTRLH: <RSCFDn_base> + 0509_H + (20_H × m),
RSCFDnCFDCmFDCTRHL: <RSCFDn_base> + $050A_H$ + (20_H × m),
RSCFDnCFDCmFDCTRHH: <RSCFDn_base> + $050B_H$ + (20_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOCCLR	EOCCLR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 25.100 RSCFDnCFDCmFDCTR Register Contents

Bit	Bit Name	Function
31 to 2	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
1	SOCCLR	Successful Occurrence Counter Clear Setting the SOCCLR bit to 1 clears the successful occurrence counter. This bit is always read as 0.
0	EOCCLR	Error Occurrence Counter Clear Setting the EOCCLR bit to 1 clears the error occurrence counter. This bit is always read as 0.

SOCCLR Bit

Setting this bit to 1 clears the successful occurrence counter (SOC[7:0] bits in the RSCFDnCFDCmFDSTS register). This bit is automatically cleared to 0.

EOCCLR Bit

Setting this bit to 1 clears the error occurrence counter (EOC[7:0] bits in the RSCFDnCFDCmFDSTS register). This bit is automatically cleared to 0.

25.4.3.8 RSCFDnCFDCmFDSTS — Channel CANFD Status Register (m = 0, 1)

Access: RSCFDnCFDCmFDSTS register can be read/written in 32-bit units
RSCFDnCFDCmFDSTSL, RSCFDnCFDCmFDSTSH registers can be read/written in 16-bit units
RSCFDnCFDCmFDSTSLL, RSCFDnCFDCmFDSTSLH, RSCFDnCFDCmFDSTSHL, RSCFDnCFDCmFDSTSHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmFDSTS: <RSCFDn_base> + $050C_H$ + (20_H × m)
RSCFDnCFDCmFDSTSL: <RSCFDn_base> + $050C_H$ + (20_H × m),
RSCFDnCFDCmFDSTSH: <RSCFDn_base> + $050E_H$ + (20_H × m)
RSCFDnCFDCmFDSTSLL: <RSCFDn_base> + $050C_H$ + (20_H × m),
RSCFDnCFDCmFDSTSLH: <RSCFDn_base> + $050D_H$ + (20_H × m),
RSCFDnCFDCmFDSTSHL: <RSCFDn_base> + $050E_H$ + (20_H × m),
RSCFDnCFDCmFDSTSHH: <RSCFDn_base> + $050F_H$ + (20_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SOC[7:0]								EOC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SOCO	EOCO	TDCVF	TDCR[6:0]						
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W *1	R/W *1	R/W *1	R	R	R	R	R	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.101 RSCFDnCFDCmFDSTS Register Contents

Bit	Bit Name	Function
31 to 24	SOC[7:0]	Successful Occurrence Counter The successful occurrence counter value can be read.
23 to 16	EOC[7:0]	Error Occurrence Counter The error occurrence counter value can be read.
15 to 10	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
9	SOCO	Successful Occurrence Counter Overflow Flag 0: The successful occurrence counter does not overflow. 1: The successful occurrence counter has overflowed.
8	EOCO	Error Occurrence Counter Overflow Flag 0: The error occurrence counter does not overflow. 1: The error occurrence counter has overflowed.
7	TDCVF	Transmitter Delay Compensation Violation Flag 0: No transmitter delay compensation violation is present. 1: A transmitter delay compensation violation is present.
6 to 0	TDCR[6:0]	Transmitter Delay Compensation Result Status The transmitter delay compensation result can be read.

SOC[7:0] Bits

These bits show the successful occurrence counter value. The successful occurrence counter is incremented upon completion of message reception or transmission without an error. This counter stops counting when it reaches FFH. In loopback mode, this counter is incremented twice.

These bits are cleared to 0 by writing 1 to the SOCCLR bit in the RSCFDnCFDCmCTR register. These bits are 0 in channel reset mode.

EOC[7:0] Bits

These bits show the error occurrence counter value. The error occurrence counter is incremented each time an error occurs according to the condition specified by the EOCCFG[2:0] bits in the RSCFDnCFDCmFDCFG register. This counter stops counting when it reaches FFH.
These bits are cleared to 0 by writing 1 to the EOCCLR bit in the RSCFDnCFDCmCTR register. These bits are 0 in channel reset mode.

SOCO Flag

This bit indicates that successful occurrence counter overflow has occurred.
This flag is set to 1 when message reception or transmission is completed while the SOC[7:0] value has reached FF_H.
This flag is 0 in channel reset mode.

EOCO Flag

This bit indicates that error occurrence counter overflow has occurred.
This flag is set to 1 when a CAN bus error is detected under the condition specified by the EOCCFG[2:0] bits in the RSCFDnCFDCmFDCFG register when the EOC[7:0] value has reached FFH. This flag is 0 in channel reset mode.

TDCVF Flag

This bit indicates violation of transmitter delay compensation.
The transmit data is compared with the reception CAN bus level delayed due to the transceiver's loop delay. This delay changes due to physical factors such as temperature. Because the TDCR[6:0] flags are updated for each message, temporary maximum delay cannot be confirmed.
This bit is set to 1 when the transmitter delay compensation exceeds the maximum compensation 3 CANm bit times - 2 Tq (CANm bit time and Tq are the values of data bit rate).
This flag is 0 in channel reset mode.

TDCR[6:0] Flags

These bits indicate the transmitter delay compensation result as a multiple of CAN clock frequency (fCAN).
This result depends on the settings of the TDCOC bit and TDCO[6:0] bits in the RSCFDnCFDCmFDCFG register.
This flag is updated at a falling edge between the FDF bit and res bit when the TDCE bit (transmitter delay compensation enable) in the RSCFDnCFDCmFDCFG register is set to 1 and also the TDCOC bit (transmitter delay compensation measurement select) in the RSCFDnCFDCmFDCFG register is set to 0.
This flag is 0 in channel reset mode.

25.4.3.9 RSCFDnCFDCmFDCRC — Channel CANFD CRC Register (m = 0, 1)

Access: RSCFDnCFDCmFDCRC register can be read only in 32-bit units
RSCFDnCFDCmFDCRCL, RSCFDnCFDCmFDCRCH registers can be read only in 16-bit units
RSCFDnCFDCmFDCRCLL, RSCFDnCFDCmFDCRCLH, RSCFDnCFDCmFDCRCHL, RSCFDnCFDCmFDCRCHH registers can be read only in 8-bit units

Address: RSCFDnCFDCmFDCRC: <RSCFDn_base> + 0510_H + (20_H × m)
RSCFDnCFDCmFDCRCL: <RSCFDn_base> + 0510_H + (20_H × m),
RSCFDnCFDCmFDCRCH: <RSCFDn_base> + 0512_H + (20_H × m)
RSCFDnCFDCmFDCRCLL: <RSCFDn_base> + 0510_H + (20_H × m),
RSCFDnCFDCmFDCRCLH: <RSCFDn_base> + 0511_H + (20_H × m),
RSCFDnCFDCmFDCRCHL: <RSCFDn_base> + 0512_H + (20_H × m),
RSCFDnCFDCmFDCRCHH: <RSCFDn_base> + 0513_H + (20_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	SCNT[3:0]				—	—	—	CRCREG[20:16]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRCREG[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.102 RSCFDnCFDCmFDCRC Register Contents

Bit	Bit Name	Function
31 to 28	Reserved	These bits are read as the value after reset.
27 to 24	SCNT[3:0]	Stuff count Indicate a value of the stuff count in a CANFD frame. Bits 25 to 27 indicate the Gray-coded value of the stuff bit count modulo 8 in the transmitted/received frames. Bit 24 indicates an even parity value of bits 25 to 27.
23 to 21	Reserved	These bits are read as the value after reset.
20 to 0	CRCREG [20:0]	CRC Calculation Data (CRC Length:17 Bits or 21 Bits) These bits show the CRC value calculated based on the transmit message or receive message. When the CRC length is 17 bits, bits b20 to b17 are read as 0.

SCNT[3:0] Flag

When the CTME bit in the RSCFDnCFDCmCTR register is set to 1 (communication test mode enabled), a stuff count bit value of the CANFD frame can be read if a message transmitted/received is a CANFD frame. When the CTME bit is 0 (communication test mode disabled), this flag is always read as 0. This flag is updated at the first bit in the CRC field of the CANFD frame. These bits are cleared to 0 in channel reset mode.

CRCREG[20:0] Flag

When the CTME bit in the RSCFDnCFDCmCTR register is 1 (communication test mode enabled), if transmit or receive message is a CANFD frame (CRC length = 17 or 21 bits), this flag is updated and the CRC value calculated based on the message can be read. When the CRC length of the message is 17 bits, bits b20 to b17 are always read as 0. When a classical CAN frame is transmitted or received, the CRCREG[14:0] value in the RSCFDnCFDCmERFL register is updated. When the CTME bit is 0 (communication test mode disabled), these bits are always read as 0.

25.4.4 Details of Global-related Registers

25.4.4.1 SCFDnCFDGCFG — Global Configuration Register

Access: RSCFDnCFDGCFG register can be read/written in 32-bit units
RSCFDnCFDGCFGL, RSCFDnCFDGCFGH registers can be read/written in 16-bit units
RSCFDnCFDGCFGLL, RSCFDnCFDGCFGLH, RSCFDnCFDGCFGHL, RSCFDnCFDGCFGHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGCFG: <RSCFDn_base> + 0084_H
RSCFDnCFDGCFGL: <RSCFDn_base> + 0084_H, RSCFDnCFDGCFGH: <RSCFDn_base> + 0086_H
RSCFDnCFDGCFGLL: <RSCFDn_base> + 0084_H, RSCFDnCFDGCFGLH: <RSCFDn_base> + 0085_H,
RSCFDnCFDGCFGHL: <RSCFDn_base> + 0086_H, RSCFDnCFDGCFGHH: <RSCFDn_base> + 0087_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ITRCP[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSBTCS[2:0]			TSSS	TSP[3:0]				—	—	CMPOC	DCS	MME	DRE	DCE	TPRI
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.103 RSCFDnCFDGCFG Register Contents

Bit	Bit Name	Function
31 to 16	ITRCP[15:0]	Interval Timer Prescaler Set When these bits are set to M, the pclk is divided by M. Setting 0000_H is prohibited when the interval timer is in use.
15 to 13	TSBTCS[2:0]	Timestamp Clock Source Select b15 b14 b13 0 0 0: Channel 0 nominal bit time clock 0 0 1: Channel 1 nominal bit time clock 0 1 0: Channel 2 nominal bit time clock 0 1 1: Channel 3 nominal bit time clock 1 0 0: Channel 4 nominal bit time clock 1 0 1: Channel 5 nominal bit time clock 1 1 0: Setting prohibited 1 1 1: Setting prohibited
12	TSSS	Timestamp Source Select 0: pclk/2*1 1: Nominal bit time clock
11 to 8	TSP[3:0]	Timestamp Clock Source Division b11 b10 b9 b8 0 0 0 0: Not divided 0 0 0 1: Divided by 2 0 0 1 0: Divided by 4 0 0 1 1: Divided by 8 0 1 0 0: Divided by 16 0 1 0 1: Divided by 32 0 1 1 0: Divided by 64 0 1 1 1: Divided by 128 1 0 0 0: Divided by 256 1 0 0 1: Divided by 512 1 0 1 0: Divided by 1024 1 0 1 1: Divided by 2048 1 1 0 0: Divided by 4096 1 1 0 1: Divided by 8192 1 1 1 0: Divided by 16384 1 1 1 1: Divided by 32768

Bit	Bit Name	Function
7, 6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5	CMPOC	Payload Overflow Mode Select 0: No message is stored. 1: Messages are stored and payloads exceeding the buffer size are discarded.
4	DCS	CAN Clock Source Select*2 0: clkc 1: clk_xincan
3	MME	Mirror Function Enable 0: Mirror function is disabled. 1: Mirror function is enabled.
2	DRE	DLC Replacement Enable 0: DLC replacement is disabled. 1: DLC replacement is enabled.
1	DCE	DLC Check Enable 0: DLC check is disabled. 1: DLC check is enabled.
0	TPRI	Transmit Priority Select 0: ID priority 1: Transmit buffer number priority

Note 1. When specifying pclk/2 as the timestamp counter count source, set bits TSBTCS[2:0] to 000_B.
Note 2. For the CAN clock frequency settings, see section 25, Clock Supply.

Modify the RSCFDnCFDGCFG register only in global reset mode.

ITRCP[15:0] Bits

These bits are used to set a clock source division value of the interval timer for FIFO buffers. See section 25.8.3.1, Interval Transmission Function.

TSBTCS[2:0] Bits

When the TSSS bit is 1, these bits are used to select the channel of the nominal bit time clock that will be the clock source of the timestamp counter. However, do not select the channel that handles the CANFD frames.

TSSS Bit

This bit is used to select a clock source of the timestamp counter. Select pclk if there is no channel that handles only classical CAN frames.

TSP[3:0] Bits

A clock obtained by dividing the clock source selected with the TSBTCS[2:0] bits and TSSS bit according to the TSP[3:0] bits is used as the timestamp counter count source.

CMPOC Bit

This bit is used to select operation in case the payload length of received message exceeds the payload storage size of the storage buffer.
When this bit is 0, the received message in which the payload overflows is not stored in the buffer.
When this bit is 1, the received message in which the payload overflows is stored in the buffer. At this time, payloads exceeding the buffer's payload storage size are discarded.
The buffer’s payload storage size is set by the following bits.

- Receive buffer: RMPLS[1:0] bits in the RSCFDnCFDRMNB register
- Receive FIFO buffer: RFPLS[2:0] bits in the RSCFDnCFDRFCCx register
- Transmit/receive FIFO buffer: CFPLS[2:0] bits in the RSCFDnCFDCFCCk register

DCS Bit

When this bit is set to 0, clkc is used as the clock source of the CAN clock (fCAN).
When this bit is set to 1, clk_xincan is used as the clock source of the CAN clock (fCAN).
For the CAN clock frequency settings, see Table 25.6, Range of Operating Frequency Depending on the Transfer Rate and the Number of Channels in Use in this LSI.

MME Bit

Setting this bit to 1 makes the mirror function available.

DRE Bit

When the DRE bit is set to 1, the DLC value of the receive rule is stored in the buffer instead of the DLC value of the received message after the DLC value has passed through the DLC filter. In this case, a value of 00_H is stored in each data byte beyond the DLC value of the receive rule.
The DLC replacement function is only available when the DCE bit is set to 1 (DLC check is enabled).

DCE Bit

Setting this bit to 1 makes the DLC check function available. When disabling the DLC check function, set the GAFLDLC[3:0] bits in the RSCFDnCFDGAFLP0_j register to 0000_B before clearing the DCE bit in the RSCFDnCFDGCFG register to 0.

TPRI Bit

This bit is used to set the transmit priority.
When this bit is set to 0, ID priority is selected and the transmit priority complies with the CAN bus arbitration rule (ISO11898-1 specifications). When this bit is set to 1, transmit buffer number priority is selected and the lowest transmit buffer number of those has the highest priority.
While the transmit queue is in use, this bit should be set to 0.

25.4.4.2 RSCFDnCFDGCTR — Global Control Register

Access: RSCFDnCFDGCTR register can be read/written in 32-bit units
RSCFDnCFDGCTRL, RSCFDnCFDGCTRH registers can be read/written in 16-bit units
RSCFDnCFDGCTRLL, RSCFDnCFDGCTRLH, RSCFDnCFDGCTRHL, RSCFDnCFDGCTRHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGCTR: <RSCFDn_base> + 0088_H
RSCFDnCFDGCTRL: <RSCFDn_base> + 0088_H, RSCFDnCFDGCTRH: <RSCFDn_base> + $008A_H$
RSCFDnCFDGCTRLL: <RSCFDn_base> + 0088_H, RSCFDnCFDGCTRLH: <RSCFDn_base> + 0089_H,
RSCFDnCFDGCTRHL: <RSCFDn_base> + $008A_H$, RSCFDnCFDGCTRHH: <RSCFDn_base> + $008B_H$

Value after reset: $0000\ 0005_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSRST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CMPOFIE	THLEIE	MEIE	DEIE	—	—	—	—	—	GSLPR	GMDC[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W:	0	0	0	0	R/W	R/W	R/W	R/W	0	0	0	0	0	R/W	R/W	R/W

Table 25.104 RSCFDnCFDGCTR Register Contents

Bit	Bit Name	Function
31 to 17	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
16	TSRST	Timestamp Counter Reset Setting the TSRST bit to 1 resets the timestamp counter. This bit is always read as 0.
15 to 12	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
11	CMPOFIE	Payload Overflow Interrupt Enable 0: A payload overflow interrupt is disabled. 1: A payload overflow interrupt is enabled.
10	THLEIE	Transmit History Buffer Overflow Interrupt Enable 0: Transmit history buffer overflow interrupt is disabled. 1: Transmit history buffer overflow interrupt is enabled.
9	MEIE	FIFO Message Lost Interrupt Enable 0: FIFO message lost interrupt is disabled. 1: FIFO message lost interrupt is enabled.
8	DEIE	DLC Error Interrupt Enable 0: DLC error interrupt is disabled. 1: DLC error interrupt is enabled.
7 to 3	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
2	GSLPR	Global Stop Mode 0: Other than global stop mode 1: Global stop mode
1, 0	GMDC[1:0]	Global Mode Select b1 b0 0 0: Global operating mode 0 1: Global reset mode 1 0: Global test mode 1 1: Setting prohibited

TSRST Bit

This bit is used to reset the timestamp counter. When this bit is set to 1, the RSCFDnCFDGTSC register is cleared to 0000_H.

CMPOFIE Bit

When the CMPOF flag in the RSCFDnCFDGERFL register is set to 1 after the CMPOFIE bit is set to 1, an interrupt request occurs. Modify this bit only in global reset mode.

THLEIE Bit

When the THLEIE bit is set to 1 and the THLES flag in the RSCFDnCFDGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

MEIE Bit

When the MEIE bit is set to 1 and the MES flag in the RSCFDnCFDGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

DEIE Bit

When the DEIE bit is set to 1 and the DEF flag in the RSCFDnCFDGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

GSLPR Bit

Setting this bit to 1 places the RSCAN module into global stop mode.
Clearing this bit to 0 makes the RSCAN module leave from global stop mode.
This bit should not be modified in global operating mode or global test mode.

GMDC[1:0] Bits

These bits are used to select the mode of entire RS-CANFD module (global operating mode, global reset mode, or global test mode). For details, see section 25.6.1, Global Modes. Setting the GSLPR bit to 1 when in global reset mode places the RS-CANFD module into global stop mode.

25.4.4.3 RSCFDnCFDGSTS — Global Status Register

Access: RSCFDnCFDGSTS register can be read only in 32-bit units
RSCFDnCFDGSTSL, RSCFDnCFDGSTSH registers can be read only in 16-bit units
RSCFDnCFDGSTSLL, RSCFDnCFDGSTSLH, RSCFDnCFDGSTSHL, RSCFDnCFDGSTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDGSTS: <RSCFDn_base> + $008C_H$
RSCFDnCFDGSTSL: <RSCFDn_base> + $008C_H$, RSCFDnCFDGSTSH: <RSCFDn_base> + $008E_H$
RSCFDnCFDGSTSLL: <RSCFDn_base> + $008C_H$, RSCFDnCFDGSTSLH: <RSCFDn_base> + $008D_H$,
RSCFDnCFDGSTSHL: <RSCFDn_base> + $008E_H$, RSCFDnCFDGSTSHH: <RSCFDn_base> + $008F_H$

Value after reset: $0000\ 000D_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	GRAMINIT	GSLPSTS	GHLTSTS	GRSTSTS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.105 RSCFDnCFDGSTS Register Contents

Bit	Bit Name	Function
31 to 4	Reserved	These bits are read as the value after reset.
3	GRAMINIT	CAN RAM Initialization Status Flag 0: CAN RAM initialization is completed. 1: CAN RAM initialization is ongoing.
2	GSLPSTS	Global Stop Status Flag 0: Not in global stop mode 1: In global stop mode
1	GHLTSTS	Global Test Status Flag 0: Not in global test mode 1: In global test mode
0	GRSTSTS	Global Reset Status Flag 0: Not in global reset mode 1: In global reset mode

GRAMINIT Flag

This flag indicates the initialization status of the CAN RAM.
This flag is set to 1 after the MCU has been reset, and is cleared to 0 when CAN RAM initialization is completed.

GSLPSTS Flag

This flag is set to 1 when the CAN module has transitioned to global stop mode, and is cleared to 0 when the CAN module has returned from global stop mode.

GHLTSTS Flag

This flag is set to 1 when the CAN module has transitioned to global test mode, and is cleared to 0 when the CAN module has exited global test mode.

GRSTSTS Flag

This flag is set to 1 when the CAN module has transitioned to global reset mode, and is cleared to 0 when the CAN module has exited global reset mode. This flag remains 1 even when the CAN module has transitioned from global reset mode to global stop mode.

25.4.4.4 RSCFDnCFDGERFL — Global Error Flag Register

Access: RSCFDnCFDGERFL register can be read/written in 32-bit units
RSCFDnCFDGERFLL, RSCFDnCFDGERFLH registers can be read/written in 16-bit units
RSCFDnCFDGERFLLL, RSCFDnCFDGERFLLH, RSCFDnCFDGERFLHL, RSCFDnCFDGERFLHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGERFL: <RSCFDn_base> + 0090_H
RSCFDnCFDGERFLL: <RSCFDn_base> + 0090_H, RSCFDnCFDGERFLH: <RSCFDn_base> + 0092_H
RSCFDnCFDGERFLLL: <RSCFDn_base> + 0090_H, RSCFDnCFDGERFLLH: <RSCFDn_base> + 0091_H,
RSCFDnCFDGERFLHL: <RSCFDn_base> + 0092_H, RSCFDnCFDGERFLHH: <RSCFDn_base> + 0093_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CMPOF	THLES	MES	DEF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R	R	R/W*1

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.106 RSCFDnCFDGERFL Register Contents

Bit	Bit Name	Function
31 to 14, 7, 6, 4	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
13 to 8, 5	Reserved	When read, an undefined value is returned.The write value should be the value after reset.
3	CMPOF	Payload Overflow Flag 0: No payload overflow has occurred. 1: A payload overflow has occurred.
2	THLES	Transmit History Buffer Overflow Status Flag 0: No transmit history buffer overflow has occurred. 1: A transmit history buffer overflow has occurred.
1	MES	FIFO Message Lost Status Flag 0: No FIFO message lost error has occurred. 1: A FIFO message lost error has occurred.
0	DEF	DLC Error Flag 0: No DLC error has occurred. 1: A DLC error has occurred.

All flags in the RSCFDnCFDGERFL register are cleared to 0 in global reset mode.

CMPOF Flag

When a payload overflow occurs in any of channel m (m = 0, 1), the CMPOF flag is set to 1. This flag can be cleared to 0 by writing 0 to this bit by the program.

THLES Flag

The THLES flag is set to 1 when any one of the THLELT flags in the RSCFDnCFDTHLSTSm register (m = 0, 1) is set to 1.
This flag is cleared to 0 when the THLELT flags of all channels are set to 0.

MES Flag

The MES flag is set to 1 when any one of the RFMLT flags in the RSCFDnCFDRFSTSx register (x = 0 to 7) or the CFMLT flags in the RSCFDnCFDCFSTSk register (k = 0 to 5) is set to 1.
This flag is cleared to 0 when all RFMLT flags and CFMLT flags are set to 0.

DEF Flag

The DEF flag is set to 1 when an error has been detected during the DLC check. The program can clear this flag by writing 0 to this bit.
To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

Note: To clear the flag of this register to 0, use a store instruction to write 0 to the given flag and 1 to the other flags.

25.4.4.5 RSCFDnCFDGTSC — Global Timestamp Counter Register

Access: RSCFDnCFDGTSC register can be read only in 32-bit units
RSCFDnCFDGTSCL, RSCFDnCFDGTSCH registers can be read only in 16-bit units

Address: RSCFDnCFDGTSC: <RSCFDn_base> + 0094_H
RSCFDnCFDGTSCL: <RSCFDn_base> + 0094_H, RSCFDnCFDGTSCH: <RSCFDn_base> + 0096_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.107 RSCFDnCFDGTSC Register Contents

Bit	Bit Name	Function
31 to 16	Reserved	These bits are read as the value after reset.
15 to 0	TS[15:0]	Timestamp Value The timestamp counter value can be read. Counter Value: 0000_H to $FFFF_H$

TS[15:0] Bits

When the TS[15:0] bits are read, the read value shows the timestamp counter (16-bit free-running counter) value at that time. When the SOF is detected, the TS[15:0] value is captured and later stored in the receive buffer or the FIFO buffer. Furthermore, the TS[15:0] value is stored in the transmit history buffer. The timestamp counter is initialized in global reset mode.

The timestamp counter starts and stops counting differently, depending on the count source.

- When the TSSS bit in the RSCFDnCFDGCFG register is 0 (pclk):
  The timestamp counter starts counting when the RSCAN module has transitioned to global operating mode.
  This counter stops counting when the RSCAN module has transitioned to global stop mode or global test mode.
- When the TSSS bit is 1 (CANm nominal bit time clock):
  The timestamp counter starts counting when the corresponding channel has transitioned to channel communication mode.
  This counter stops counting when the corresponding channel has transitioned to channel reset mode or channel halt mode.

25.4.4.6 RSCFDnCFDGTINTSTS0 — Global TX Interrupt Status Register 0

Access: RSCFDnCFDGTINTSTS0 register can be read only in 32-bit units
RSCFDnCFDGTINTSTS0L, RSCFDnCFDGTINTSTS0H registers can be read only in 16-bit units
RSCFDnCFDGTINTSTS0LL, RSCFDnCFDGTINTSTS0LH, RSCFDnCFDGTINTSTS0HL, RSCFDnCFDGTINTSTS0HH registers can be read only in 8-bit units

Address: RSCFDnCFDGTINTSTS0: <RSCFDn_base> + 0460_H
RSCFDnCFDGTINTSTS0L: <RSCFDn_base> + 0460_H, RSCFDnCFDGTINTSTS0H: <RSCFDn_base> + 0462_H
RSCFDnCFDGTINTSTS0LL: <RSCFDn_base> + 0460_H, RSCFDnCFDGTINTSTS0LH: <RSCFDn_base> + 0461_H,
RSCFDnCFDGTINTSTS0HL: <RSCFDn_base> + 0462_H, RSCFDnCFDGTINTSTS0HH: <RSCFDn_base> + 0463_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1	—	—	—	THIF0	CFTIF0	TQIF0	TAIF0	TSIF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Note 1. This bit is automatically cleared in the global reset or channel reset mode.

Table 25.108 RSCFDnCFDGTINTSTS0 Register Contents

Bit	Bit Name	Function
31 to 13	Reserved	These bits are read as the value after reset.
12	THIF1	Channel 1 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
11	CFTIF1	Channel 1 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
10	TQIF1	Channel 1 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
9	TAIF1	Channel 1 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
8	TSIF1	Channel 1 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.
7 to 5	Reserved	These bits are read as the value after reset.
4	THIF0	Channel 0 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
3	CFTIF0	Channel 0 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
2	TQIF0	Channel 0 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
1	TAIF0	Channel 0 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.

Bit	Bit Name	Function
0	TSIF0	Channel 0 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.

TSIFm Bits

The TSIFm bit is set to 1 when the TMIEp bit in the RSCFDnCFDTMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnCFDTMSTSp register are set to 10_B (transmit completed without abort request) or 11_B (transmit completed with abort request).
When the TMTRF[1:0] flags are cleared to 00_B under the condition that the TSIFm bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIEp bit to 0 also clears this flag to 0.

TAIFm Bits

The TAIFm bit is set to 1 when the TAIE bit in the RSCFDnCFDCmCTR register is 1 (transmit abort interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnCFDTMSTSp register are set to 01_B (transmit abort completed).
This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to 00_B after the transmit abort is completed.

TQIFm Bits

When the TXQIE bit in the RSCFDnCFDTXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCFDnCFDTXQSTSm register is set to 1 (transmit queue interrupt request), the TQIFm bit is set to 1.
When the TXQIF bit (transmit queue interrupt request) in the RSCFDnCFDTXQSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the TXQIE bit is cleared to 0.

CFTIFm Bits

When the CFTXIE bit in the RSCFDnCFDCFCCk register is set to 1 (transmit/receive FIFO transmit interrupt enabled) and the CFTXIF bit in the RSCFDnCFDCFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), the CFTIFm bit is set to 1.
When the CFTXIF bit is cleared to 0 under the conditions that the CFTIFm bit can be set to 1, this bit is cleared to 0. This flag is also cleared to 0 when the CFTXIE bit is cleared to 0.

THIFm Bits

When the THLIE bit in the RSCFDnCFDTHLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCFDnCFDTHLSTSm register is set to 1 (transmit history interrupt request), the THIFm bit is set to 1.
When the THLIF bit in the RSCFDnCFDTHLSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the THLIE bit is cleared to 0.

25.4.4.7 RSCFDnCFDGFDCFG — Global FD Configuration Register

Access: RSCFDnCFDGFDCFG register can be read/written in 32-bit units
RSCFDnCFDGFDCFG L, RSCFDnCFDGFDCFG H registers can be read/written in 16-bit units
RSCFDnCFDGFDCFG LL, RSCFDnCFDGFDCFG LH, RSCFDnCFDGFDCFG HL, RSCFDnCFDGFDCFG HH registers can be read/written in 8-bit units

Address: RSCFDnCFDGFDCFG: <RSCFDn_base> + 0474_H
RSCFDnCFDGFDCFG L: <RSCFDn_base> + 0474_H, RSCFDnCFDGFDCFG H: <RSCFDn_base> + 0476_H
RSCFDnCFDGFDCFG LL: <RSCFDn_base> + 0474_H, RSCFDnCFDGFDCFG LH: <RSCFDn_base> + 0475_H,
RSCFDnCFDGFDCFG HL: <RSCFDn_base> + 0476_H, RSCFDnCFDGFDCFG HH: <RSCFDn_base> + 0477_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSCCFG[1:0]		—	—	—	—	—	—	—	RPED
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.109 RSCFDnCFDGFDCFG Register Contents

Bit	Bit Name	Function
31 to 10	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
9, 8	TSCCFG[1:0]	Time-stamp capture setting bit b9 b8 0 0: Captured at a sample point in the SOF bit. 0 1: Captured when a valid frame has been transmitted/received. 1 0: Captured at a sample point of the res bit. *1 1 1: Setting prohibited.
7 to 1	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
0	RPED	Protocol exception event detection disabled bit 0: Protocol exception event detection enabled 1: Protocol exception event detection disabled

Note 1. When a Classical CAN frame is transmitted/received, a time-stamp value will be captured at the sample point in the SOF bit.

TSCCFG[1:0] Bits

Select a point where a time-stamp value is captured. Modify this bit only in global reset mode.

PRED Bit

Setting this bit to 1 disables the protocol exception event detection. When a protocol exception event is detected while this bit is set to 1, the event is regarded as a form error and an error frame is output. Modify this bit only in global reset mode.

25.4.5 Details of Receive Rule-related Registers

25.4.5.1 RSCFDnCFDGAFLECTR — Receive Rule Entry Control Register

Access: RSCFDnCFDGAFLECTR register can be read/written in 32-bit units
RSCFDnCFDGAFLECTRL, RSCFDnCFDGAFLECTRH registers can be read/written in 16-bit units
RSCFDnCFDGAFLECTRLL, RSCFDnCFDGAFLECTRLH, RSCFDnCFDGAFLECTRHL, RSCFDnCFDGAFLECTRHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLECTR: <RSCFDn_base> + 0098_H
RSCFDnCFDGAFLECTRL: <RSCFDn_base> + 0098_H, RSCFDnCFDGAFLECTRH: <RSCFDn_base> + $009A_H$
RSCFDnCFDGAFLECTRLL: <RSCFDn_base> + 0098_H, RSCFDnCFDGAFLECTRLH: <RSCFDn_base> + 0099_H,
RSCFDnCFDGAFLECTRHL: <RSCFDn_base> + $009A_H$, RSCFDnCFDGAFLECTRHH: <RSCFDn_base> + $009B_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AFLDAE	—	—	—	AFLPN[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 25.110 RSCFDnCFDGAFLECTR Register Contents

Bit	Bit Name	Function
31 to 9	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
8	AFLDAE	Receive Rule Table Write Enable 0: Receive rule table write is disabled. 1: Receive rule table write is enabled.
7 to 5	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
4 to 0	AFLPN[4:0]	Receive Rule Table Page Number Configuration A page number can be selected from a range of page 0 (00000_B) to page 7 (00111_B).

AFLDAE Bit

Setting this bit to 0 disables the write to the receive rule table. After writes to the receive rule table are completed, set this bit to 0 to disable the write to the table. The receive rule table can be read regardless of the value of this bit.
Set the AFLDAE bit to 1 only in global reset mode.

AFLPN[4:0] Bits

These bits are used to set the page number of the receive rule table. Sixteen receive rules can be set per page.
Set these bits to a value within the range of 00000_B to 00111_B.

25.4.5.2 RSCFDnCFDGAFLCFG0 — Receive Rule Configuration Register 0

Access: RSCFDnCFDGAFLCFG0 register can be read/written in 32-bit units
RSCFDnCFDGAFLCFG0L, RSCFDnCFDGAFLCFG0H registers can be read/written in 16-bit units
RSCFDnCFDGAFLCFG0LL, RSCFDnCFDGAFLCFG0LH, RSCFDnCFDGAFLCFG0HL, RSCFDnCFDGAFLCFG0HH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLCFG0: <RSCFDn_base> + $009C_H$
RSCFDnCFDGAFLCFG0L: <RSCFDn_base> + $009C_H$, RSCFDnCFDGAFLCFG0H: <RSCFDn_base> + $009E_H$
RSCFDnCFDGAFLCFG0LL: <RSCFDn_base> + $009C_H$, RSCFDnCFDGAFLCFG0LH: <RSCFDn_base> + $009D_H$,
RSCFDnCFDGAFLCFG0HL: <RSCFDn_base> + $009E_H$, RSCFDnCFDGAFLCFG0HH: <RSCFDn_base> + $009F_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RNC0[7:0]								RNC1[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.111 RSCFDnCFDGAFLCFG0 Register Contents

Bit	Bit Name	Function
31 to 24	RNC0[7:0]	Number of Rules for Channel 0 Set the number of receive rules exclusively used for channel 0.
23 to 16	RNC1[7:0]	Number of Rules for Channel 1 Set the number of receive rules exclusively used for channel 1.
15 to 0	Reserved	These bits are read as the value after reset. The write value should be the value after reset.

Modify the RSCFDnCFDGAFLCFG0 register only in global reset mode.
Up to 64 × (number of channels) rules can be registered in the receive rule table as the entire unit. The number of receive rules per channel should meet the following conditions.

- The maximum number of rules per channel is 128.
- The total of the number of rules allocated to each channel is not larger than the number of rules that can be registered in the entire unit.

RNC0[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 0 receive rule table.
Set these bits to a value within the range of 00_H to 80_H.

RNC1[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 1 receive rule table.
Set these bits to a value within the range of 00_H to 80_H.

25.4.5.3 RSCFDnCFDGAFLIDj — Receive Rule ID Register (j = 0 to 15)

Access: RSCFDnCFDGAFLIDj register can be read/written in 32-bit units
RSCFDnCFDGAFLIDjL, RSCFDnCFDGAFLIDjH registers can be read/written in 16-bit units
RSCFDnCFDGAFLIDjLL, RSCFDnCFDGAFLIDjLH, RSCFDnCFDGAFLIDjHL, RSCFDnCFDGAFLIDjHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLIDj: <RSCFDn_base> + 1000_H + (10_H × j)
RSCFDnCFDGAFLIDjL: <RSCFDn_base> + 1000_H + (10_H × j),
RSCFDnCFDGAFLIDjH: <RSCFDn_base> + 1002_H + (10_H × j)
RSCFDnCFDGAFLIDjLL: <RSCFDn_base> + 1000_H + (10_H × j),
RSCFDnCFDGAFLIDjLH: <RSCFDn_base> + 1001_H + (10_H × j),
RSCFDnCFDGAFLIDjHL: <RSCFDn_base> + 1002_H + (10_H × j),
RSCFDnCFDGAFLIDjHH: <RSCFDn_base> + 1003_H + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFL IDE	GAFL RTR	GAFL LB	GAFLID[28:16]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLID[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.112 RSCFDnCFDGAFLIDj Register Contents

Bit	Bit Name	Function
31	GAFLIDE	IDE Select 0: Standard ID 1: Extended ID
30	GAFLRTR	RTR Select 0: Data frame 1: Remote frame
29	GAFLLB	Receive Rule Target Message Select 0: When a message transmitted from another CAN node is received 1: When the own transmitted message is received
28 to 0	GAFLID [28:0]	ID Set Set the ID of the receive rule. For the standard ID, set the ID in bits b10 to b0 and set bits b28 to b11 to 0.

Modify the RSCFDnCFDGAFLIDj register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDE Bit

This bit is used to select the ID format (standard ID or extended ID) of the receive rule. This bit is compared with the IDE bit in the received message during the acceptance filter processing.

GAFLRTR Bit

This bit is used to select the frame format (data frame or remote frame) of the receive rule. This bit is compared with the RTR bit in the received message during the acceptance filter processing.

GAFLLB Bit

When this bit is set to 0, data processing using the receive rule is performed when receiving messages transmitted from another CAN node.
When this bit is set to 1 when the mirror function is used, data processing using the receive rule is performed when the CAN node is receiving its own transmitted messages.

GAFLID[28:0] Bits

These bits are used to set the ID field of the receive rule. The ID value set by these bits is compared with the ID of the received message during the acceptance filter processing.

25.4.5.4 RSCFDnCFDGAFLMj — Receive Rule Mask Register (j = 0 to 15)

Access: RSCFDnCFDGAFLMj register can be read/written in 32-bit units
RSCFDnCFDGAFLMjL, RSCFDnCFDGAFLMjH registers can be read/written in 16-bit units
RSCFDnCFDGAFLMjLL, RSCFDnCFDGAFLMjLH, RSCFDnCFDGAFLMjHL, RSCFDnCFDGAFLMjHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLMj: <RSCFDn_base> + 1004_H + (10_H × j)
RSCFDnCFDGAFLMjL: <RSCFDn_base> + 1004_H + (10_H × j),
RSCFDnCFDGAFLMjH: <RSCFDn_base> + 1006_H + (10_H × j)
RSCFDnCFDGAFLMjLL: <RSCFDn_base> + 1004_H + (10_H × j),
RSCFDnCFDGAFLMjLH: <RSCFDn_base> + 1005_H + (10_H × j),
RSCFDnCFDGAFLMjHL: <RSCFDn_base> + 1006_H + (10_H × j),
RSCFDnCFDGAFLMjHH: <RSCFDn_base> + 1007_H + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFL IDEM	GAFL RTRM	—	GAFLIDM[28:16]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLIDM[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.113 RSCFDnCFDGAFLMj Register Contents

Bit	Bit Name	Function
31	GAFLIDEM	IDE Mask 0: The IDE bit is not compared. 1: The IDE bit is compared.
30	GAFLRTRM	RTR Mask 0: The RTR bit is not compared. 1: The RTR bit is compared
29	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
28 to 0	GAFLIDM[28:0]	ID Mask 0: The corresponding ID bit is not compared. 1: The corresponding ID bit is compared.

Modify the RSCFDnCFDGAFLMj register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDEM Bit

When this bit is set to 1, filter processing is performed only for messages of the ID format specified by the GAFLIDE bit in the RSCFDnCFDGAFLIDj register.
When this bit is cleared to 0, the IDs of all the receive messages and the specified IDs are regarded as matched. To set the GAFLIDEM bit to 0, set the GAFLIDM[28:0] bits to all 0 at the same time.

GAFLRTRM Bit

This bit is used to mask the RTR bit of the receive rule.

GAFLIDM[28:0] Bits

These bits are used to mask the corresponding ID bit of the receive rule.

25.4.5.5 RSCFDnCFDGAFLP0_j — Receive Rule Pointer 0 Register (j = 0 to 15)

Access: RSCFDnCFDGAFLP0_j register can be read/written in 32-bit units
RSCFDnCFDGAFLP0_jL, RSCFDnCFDGAFLP0_jH registers can be read/written in 16-bit units
RSCFDnCFDGAFLP0jLL, RSCFDnCFDGAFLP0jLH, RSCFDnCFDGAFLP0jHL, RSCFDnCFDGAFLP0jHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLP0_j: <RSCFDn_base> + 1008_H + (10_H × j)
RSCFDnCFDGAFLP0_jL: <RSCFDn_base> + 1008_H + (10_H × j),
RSCFDnCFDGAFLP0_jH: <RSCFDn_base> + $100A_H$ + (10_H × j)
RSCFDnCFDGAFLP0_jLL: <RSCFDn_base> + 1008_H + (10_H × j),
RSCFDnCFDGAFLP0_jLH: <RSCFDn_base> + 1009_H + (10_H × j),
RSCFDnCFDGAFLP0_jHL: <RSCFDn_base> + $100A_H$ + (10_H × j),
RSCFDnCFDGAFLP0_jHH: <RSCFDn_base> + $100B_H$ + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLDLC[3:0]				GAFLPTR[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLRMV	GAFLRMDP[6:0]							—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Table 25.114 RSCFDnCFDGAFLP0_j Register Contents

Bit	Bit Name	Function
31 to 28	GAFLDLC [3:0]	Receive Rule DLC

b31 b30 b29 b28	Classical CAN Frame	CANFD Frame
0 0 0 0	0 data bytes	
0 0 0 1	1 data byte	
0 0 1 0	2 data bytes	
0 0 1 1	3 data bytes	
0 1 0 0	4 data bytes	
0 1 0 1	5 data bytes	
0 1 1 0	6 data bytes	
0 1 1 1	7 data bytes	
1 0 0 0	8 data bytes	
1 0 0 1	8 data bytes	12 data bytes
1 0 1 0		16 data bytes
1 0 1 1		20 data bytes
1 1 0 0		24 data bytes
1 1 0 1		32 data bytes
1 1 1 0		48 data bytes
1 1 1 1		64 data bytes

27 to 16	GAFLPTR [11:0]	Receive Rule Label Set the 12-bit label information.

Bit	Bit Name	Function
15	GAFLRMV	Receive Buffer Enable 0: No receive buffer is used. 1: A receive buffer is used.
14 to 8	GAFLRMDP [6:0]	Receive Buffer Number Select Set the receive buffer number to store receive messages.
7 to 0	Reserved	These bits are read as the value after reset. The write value should be the value after reset.

Modify the RSCFDnCFDGAFLP0_j register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLDLC[3:0] Bits

These bits are used to set the minimum data length necessary for receiving messages. If the data length of a message that is being filtered is equal to or larger than the value set by the GAFLDLC[3:0] bits, the message passes the DLC check. Setting these bits to 0000_B disables the DLC check function allowing messages with any data length to pass the DLC check.

GAFLPTR[11:0] Bits

These bits are used to set a 12-bit label to be attached to messages that have passed through the filter. A label is attached when a message is stored in the receive buffer or the FIFO buffer.

GAFLRMV Bit

When this bit is set to 1, receive messages that have passed through the filter are stored in the receive buffer selected by the GAFLRMDP[6:0] bits.

GAFLRMDP[6:0] Bits

These bits are used to select the number of the receive buffer that stores receive messages that have passed through the filter when the GAFLRMV bit is set to 1. Set these bits to a value smaller than the value set by the NRXMB[7:0] bits in the RSCFDnCFDRMNB register.

25.4.5.6 RSCFDnCFDGAFLP1_j — Receive Rule Pointer 1 Register (j = 0 to 15)

Access: RSCFDnCFDGAFLP1_j register can be read/written in 32-bit units
RSCFDnCFDGAFLP1_jL, RSCFDnCFDGAFLP1_jH registers can be read/written in 16-bit units
RSCFDnCFDGAFLP1_jLL, RSCFDnCFDGAFLP1_jLH, RSCFDnCFDGAFLP1_jHL, RSCFDnCFDGAFLP1_jHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLP1_j: <RSCFDn_base> + $100C_H$ + (10_H × j)
RSCFDnCFDGAFLP1_jL: <RSCFDn_base> + $100C_H$ + (10_H × j),
RSCFDnCFDGAFLP1_jH: <RSCFDn_base> + $100E_H$ + (10_H × j)
RSCFDnCFDGAFLP1_jLL: <RSCFDn_base> + $100C_H$ + (10_H × j),
RSCFDnCFDGAFLP1_jLH: <RSCFDn_base> + $100D_H$ + (10_H × j),
RSCFDnCFDGAFLP1_jHL: <RSCFDn_base> + $100E_H$ + (10_H × j),
RSCFDnCFDGAFLP1_jHH: <RSCFDn_base> + $100F_H$ + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	GAFLFDP[13:0]													
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.115 RSCFDnCFDGAFLP1_j Register Contents

Bit	Bit Name	Function
31 to 14	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
13 to 8	GAFLFDP [13:8]	Transmit/Receive FIFO Buffer k Select (Bit position –8 = target transmit/receive FIFO buffer number k) 0: Transmit/receive FIFO buffer is not selected. 1: Transmit/receive FIFO buffer is selected.
7 to 0	GAFLFDP [7:0]	Receive FIFO Buffer x Select (Bit position = target receive FIFO buffer number x) 0: Receive FIFO buffer is not selected. 1: Receive FIFO buffer is selected.

Modify the RSCFDnCFDGAFLP1_j register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLFDP[13:0] Bits

These bits are used to specify FIFO buffers that store receive messages that have passed through the filter. Up to eight FIFO buffers are selectable. However, when the GAFLRMV bit in the RSCFDnCFDGAFLP0_j register is set to 1 (a message is stored in the receive buffer), up to seven FIFO buffers can be selected. Only receive FIFO buffers and the transmit/receive FIFO buffer for which the CFM[1:0] bits in the RSCFDnCFDCFCCk register are set to 00_B (receive mode) or 10_B (gateway mode) are selectable.

25.4.6 Details of Receive Buffer-related Registers

25.4.6.1 RSCFDnCFDRMNB — Receive Buffer Number Register

Access: RSCFDnCFDRMNB register can be read/written in 32-bit units
RSCFDnCFDRMNBL, RSCFDnCFDRMNBH registers can be read/written in 16-bit units
RSCFDnCFDRMNBLL, RSCFDnCFDRMNBLH, RSCFDnCFDRMNBHL, RSCFDnCFDRMNBHH registers can be read/written in 8-bit units

Address: RSCFDnCFDRMNB: <RSCFDn_base> + $00A4_H$
RSCFDnCFDRMNBL: <RSCFDn_base> + $00A4_H$, RSCFDnCFDRMNBH: <RSCFDn_base> + $00A6_H$
RSCFDnCFDRMNBLL: <RSCFDn_base> + $00A4_H$, RSCFDnCFDRMNBLH: <RSCFDn_base> + $00A5_H$,
RSCFDnCFDRMNBHL: <RSCFDn_base> + $00A6_H$, RSCFDnCFDRMNBHH: <RSCFDn_base> + $00A7_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RMPLS[1:0]		NRXMB[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.116 RSCFDnCFDRMNB Register Contents

Bit	Bit Name	Function
31 to 10	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
9, 8	RMPLS[1:0]	Receive Buffer Payload Storage Size Select b9 b8 0 0: 8 bytes 0 1: 12 bytes 1 0: 16 bytes 1 1: 20 bytes
7 to 0	NRXMB[7:0]	Receive Buffer Number Configuration Set the number of receive buffers. Set a value of 0 to 32.

Modify the RSCFDnCFDRMNB register only in global reset mode.

RMPLS[1:0] Bits

These bits are used to select the maximum payload size that can be stored in the receive buffer.

NRXMB[7:0] Bits

These bits are used to set the total number of receive buffers of the RS-CANFD module. The maximum value is 16 × (number of channels).
Setting these bits all to 0 makes receive buffers unavailable.

25.4.6.2 RSCFDnCFDRMNDy — Receive Buffer New Data Register (y = 0)

Access: RSCFDnCFDRMNDy register can be read/written in 32-bit units
RSCFDnCFDRMNDyL, RSCFDnCFDRMNDyH registers can be read/written in 16-bit units
RSCFDnCFDRMNDyLL, RSCFDnCFDRMNDyLH, RSCFDnCFDRMNDyHL, RSCFDnCFDRMNDyHH registers can be read/written in 8-bit units

Address: RSCFDnCFDRMNDy: <RSCFDn_base> + $00A8_H$ + (04_H × y)
RSCFDnCFDRMNDyL: <RSCFDn_base> + $00A8_H$ + (04_H × y),
RSCFDnCFDRMNDyH: <RSCFDn_base> + $00AA_H$ + (04_H × y)
RSCFDnCFDRMNDyLL: <RSCFDn_base> + $00A8_H$ + (04_H × y),
RSCFDnCFDRMNDyLH: <RSCFDn_base> + $00A9_H$ + (04_H × y),
RSCFDnCFDRMNDyHL: <RSCFDn_base> + $00AA_H$ + (04_H × y),
RSCFDnCFDRMNDyHH: <RSCFDn_base> + $00AB_H$ + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMNSq (q = y × 32 + 31 to y × 32 + 16 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMNSq (q = y × 32 + 15 to y × 32 + 0 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.117 RSCFDnCFDRMNDy Register Contents

Bit	Bit Name	Function
31 to 16	RMNSq	Receive Buffer Receive Complete Flag q (q = y × 32 + 31 to y × 32 + 16) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.
15 to 0	RMNSq	Receive Buffer Receive Complete Flag q (q = y × 32 + 15 to y × 32 + 0) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.

Write 0 to the RSCFDnCFDRMNDy register in global operating mode or global test mode.

RMNSq Flags (q = 0 to 31)

Each RMNSq flag is set to 1 when the processing for storing a message in the corresponding receive buffer starts.
To clear a flag to 0, the program must write 0 to the flag. Use a store instruction to write “0” to the flag and “1” to other flags. These bits cannot be set to 0 while a message is being stored. The message storing time depends on the storage payload size of the receive buffer. When the RMPLS[1:0] value in the RSCFDnCFDRMNB register is 00_B (8 bytes), the message storing time is 12 pclk clock cycles. When the RMPLS[1:0] value is 11_B (20 bytes), the message storing time is 18 pclk clock cycles. (2 pclk clock cycles per 4 bytes of storage payload size).
These flags are cleared to 0 in global reset mode.

25.4.6.3 RSCFDnCFDRMIDq — Receive Buffer ID Register (q = 0 to 31)

Access: RSCFDnCFDRMIDq register can be read only in 32-bit units
RSCFDnCFDRMIDqL, RSCFDnCFDRMIDqH registers can be read only in 16-bit units
RSCFDnCFDRMIDqLL, RSCFDnCFDRMIDqLH, RSCFDnCFDRMIDqHL, RSCFDnCFDRMIDqHH registers can be read only in 8-bit units

Address: RSCFDnCFDRMIDq: <RSCFDn_base> + 2000_H + (20_H × q)
RSCFDnCFDRMIDqL: <RSCFDn_base> + 2000_H + (20_H × q),
RSCFDnCFDRMIDqH: <RSCFDn_base> + 2002_H + (20_H × q)
RSCFDnCFDRMIDqLL: <RSCFDn_base> + 2000_H + (20_H × q),
RSCFDnCFDRMIDqLH: <RSCFDn_base> + 2001_H + (20_H × q),
RSCFDnCFDRMIDqHL: <RSCFDn_base> + 2002_H + (20_H × q),
RSCFDnCFDRMIDqHH: <RSCFDn_base> + 2003_H + (20_H × q)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMIDE	RMRTR	—	RMID[28:16]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMID[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.118 RSCFDnCFDRMIDq Register Contents

Bit	Bit Name	Function
31	RMIDE	Receive Buffer IDE 0: Standard ID 1: Extended ID
30	RMRTR	Receive Buffer RTR/RRS • When the received message is a classical CAN frame 0: Data frame 1: Remote frame • When the received message is a CANFD frame The RRS bit value of the received message can be read.
29	Reserved	This bit is read as the value after reset.
28 to 0	RMID[28:0]	Receive Buffer ID Data These bits contain the standard ID or extended ID of the received message. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RMIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive buffer.

RMRTR Bit

When the received message is a classical CAN frame, this bit indicates the frame format (data frame or remote frame) of the message stored in the receive buffer. When the received message is a CANFD frame, this bit indicates the RRS bit value in the message.

RMID[28:0] Bits

These bits contain the ID of the message stored in the receive buffer.

25.4.6.4 RSCFDnCFDRMPTRq — Receive Buffer Pointer Register (q = 0 to 31)

Access: RSCFDnCFDRMPTRq register can be read only in 32-bit units
RSCFDnCFDRMPTRqL, RSCFDnCFDRMPTRqH registers can be read only in 16-bit units
RSCFDnCFDRMPTRqLL, RSCFDnCFDRMPTRqLH, RSCFDnCFDRMPTRqHL, RSCFDnCFDRMPTRqHH registers can be read only in 8-bit units

Address: RSCFDnCFDRMPTRq: <RSCFDn_base> + 2004_H + (20_H × q)
RSCFDnCFDRMPTRqL: <RSCFDn_base> + 2004_H + (20_H × q),
RSCFDnCFDRMPTRqH: <RSCFDn_base> + 2006_H + (20_H × q)
RSCFDnCFDRMPTRqLL: <RSCFDn_base> + 2004_H + (20_H × q),
RSCFDnCFDRMPTRqLH: <RSCFDn_base> + 2005_H + (20_H × q),
RSCFDnCFDRMPTRqHL: <RSCFDn_base> + 2006_H + (20_H × q),
RSCFDnCFDRMPTRqHH: <RSCFDn_base> + 2007_H + (20_H × q)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDLC[3:0]				RMPTR[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMTS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.119 RSCFDnCFDRMPTRq Register Contents

Bit	Bit Name	Function
31 to 28	RMDLC [3:0]	Receive Buffer DLC Data

b31 b30 b29 b28	Classical CAN Frame	CANFD Frame
0 0 0 0	0 data bytes	
0 0 0 1	1 data byte	
0 0 1 0	2 data bytes	
0 0 1 1	3 data bytes	
0 1 0 0	4 data bytes	
0 1 0 1	5 data bytes	
0 1 1 0	6 data bytes	
0 1 1 1	7 data bytes	
1 0 0 0	8 data bytes	
1 0 0 1	8 data bytes	12 data bytes
1 0 1 0		16 data bytes
1 0 1 1		20 data bytes
1 1 0 0		24 data bytes
1 1 0 1		32 data bytes
1 1 1 0		48 data bytes
1 1 1 1		64 data bytes

27 to 16	RMPTR [11:0]	Receive Buffer Label Data Label information of the received message.

Bit	Bit Name	Function
15 to 0	RMTS [15:0]	Receive Buffer Timestamp Data Timestamp value of the received message.

RMDLC[3:0] Bits

These bits indicate the data length of the message stored in the receive buffer. The number of bytes of the payload to be stored in the receive buffer is determined by the RMPLS[1:0] bits in the RSCFDnCFDRMNB register.

RMPTR[11:0] Bits

These bits indicate the label information of the message stored in the receive buffer.

RMTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the receive buffer.

25.4.6.5 RSCFDnCFDRMFDSTSq — Receive Buffer CANFD Status Register (q = 0 to 31)

Access: RSCFDnCFDRMFDSTSq register can be read only in 32-bit units
RSCFDnCFDRMFDSTSqL, RSCFDnCFDRMFDSTSqH registers can be read only in 16-bit units
RSCFDnCFDRMFDSTSqLL, RSCFDnCFDRMFDSTSqLH, RSCFDnCFDRMFDSTSqHL, RSCFDnCFDRMFDSTSqHH registers can be read only in 8-bit units

Address: RSCFDnCFDRMFDSTSq: <RSCFDn_base> + 2008_H + (20_H × q)
RSCFDnCFDRMFDSTSqL: <RSCFDn_base> + 2008_H + (20_H × q),
RSCFDnCFDRMFDSTSqH: <RSCFDn_base> + $200A_H$ + (20_H × q)
RSCFDnCFDRMFDSTSqLL: <RSCFDn_base> + 2008_H + (20_H × q),
RSCFDnCFDRMFDSTSqLH: <RSCFDn_base> + 2009_H + (20_H × q),
RSCFDnCFDRMFDSTSqHL: <RSCFDn_base> + $200A_H$ + (20_H × q),
RSCFDnCFDRMFDSTSqHH: <RSCFDn_base> + $200B_H$ + (20_H × q)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RMFDF	RMBRS	RMESI
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.120 RSCFDnCFDRMFDSTSq Register Contents

Bit	Bit Name	Function
31 to 3	Reserved	These bits are read as the value after reset.
2	RMFDF	FDF 0: Classical CAN frame 1: CANFD frame
1	RMBRS	BRS 0: The bit rate in the data area does not change. 1: The bit rate in the data area changes.
0	RMESI	ESI 0: Error active node 1: Error passive node

RMFDF Bit

This bit indicates the FD format (classical CAN frame or CANFD frame) of the message stored in the receive buffer.

RMBRS Bit

When the RMFDF bit is 1, this bit indicates the BRS bit value of the message stored in the receive buffer. When the RMFDF bit is 0, this bit is always read as 0.

RMESI Bit

When the RMFDF bit is 1, this bit indicates the ESI bit value of the message stored in the receive buffer. When the RMFDF bit is 0, this bit is always read as 0.

25.4.6.6 RSCFDnCFDRMDFb_q — Receive Buffer Data Field b Register (b = 0 to 4, q = 0 to 31)

Access: RSCFDnCFDRMDFb_q register can be read only in 32-bit units
RSCFDnCFDRMDFb_qL, RSCFDnCFDRMDFb_qH registers can be read only in 16-bit units
RSCFDnCFDRMDFb_qLL, RSCFDnCFDRMDFb_qLH, RSCFDnCFDRMDFb_qHL, RSCFDnCFDRMDFb_qHH registers can be read only in 8-bit units

Address: RSCFDnCFDRMDFb_q: <RSCFDn_base> + $200C_H$ + (04_H × b) + (20_H × q)
RSCFDnCFDRMDFb_qL: <RSCFDn_base> + $200C_H$ + (04_H × b) + (20_H × q),
RSCFDnCFDRMDFb_qH: <RSCFDn_base> + $200E_H$ + (04_H × b) + (20_H × q)
RSCFDnCFDRMDFb_qLL: <RSCFDn_base> + $200C_H$ + (04_H × b) + (20_H × q),
RSCFDnCFDRMDFb_qLH: <RSCFDn_base> + $200D_H$ + (04_H × b) + (20_H × q),
RSCFDnCFDRMDFb_qHL: <RSCFDn_base> + $200E_H$ + (04_H × b) + (20_H × q),
RSCFDnCFDRMDFb_qHH: <RSCFDn_base> + $200F_H$ + (04_H × b) + (20_H × q)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB4 × b + 3 [7:0]								RMDB4 × b + 2 [7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB4 × b + 1 [7:0]								RMDB4 × b + 0 [7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.121 RSCFDnCFDRMDFb_q Register Contents

Bit	Bit Name	Function
31 to 24	RMDB4 × b + 3 [7:0]	Receive Buffer Data Byte 4 × b + 3
23 to 16	RMDB4 × b + 2 [7:0]	Receive Buffer Data Byte 4 × b + 2
15 to 8	RMDB4 × b + 1 [7:0]	Receive Buffer Data Byte 4 × b + 1
7 to 0	RMDB4 × b + 0 [7:0]	Receive Buffer Data Byte 4 × b + 0 Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCFDnCFDRMPTRq register is smaller than the payload storage size of the receive buffer, data bytes for which no data is set are read as 00_H.
Specify the payload storage size of the receive buffer by the RMPLS[1:0] bits in the RSCFDnCFDRMNB register. Do not read or write the RSCFDnCFDRMDFb_q register corresponding to an area larger than the specified size.

25.4.7 Details of Receive FIFO Buffer-related Registers

25.4.7.1 RSCFDnCFDRFCCx — Receive FIFO Buffer Configuration and Control Register (x = 0 to 7)

Access: RSCFDnCFDRFCCx register can be read/written in 32-bit units
RSCFDnCFDRFCCxL, RSCFDnCFDRFCCxH registers can be read/written in 16-bit units
RSCFDnCFDRFCCxLL, RSCFDnCFDRFCCxLH, RSCFDnCFDRFCCxHL, RSCFDnCFDRFCCxHH registers can be read/written in 8-bit units

Address: RSCFDnCFDRFCCx: <RSCFDn_base> + $00B8_H$ + (04_H × x)
RSCFDnCFDRFCCxL: <RSCFDn_base> + $00B8_H$ + (04_H × x),
RSCFDnCFDRFCCxH: <RSCFDn_base> + $00BA_H$ + (04_H × x)
RSCFDnCFDRFCCxLL: <RSCFDn_base> + $00B8_H$ + (04_H × x),
RSCFDnCFDRFCCxLH: <RSCFDn_base> + $00B9_H$ + (04_H × x),
RSCFDnCFDRFCCxHL: <RSCFDn_base> + $00BA_H$ + (04_H × x),
RSCFDnCFDRFCCxHH: <RSCFDn_base> + $00BB_H$ + (04_H × x)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFIGCV[2:0]			RFIM	—	RFDC[2:0]			—	RFPLS[2:0]			—	—	RFIE	RFE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W

Table 25.122 RSCFDnCFDRFCCx Register Contents

Bit	Bit Name	Function
31 to 16	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
15 to 13	RFIGCV[2:0]	Receive FIFO Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	RFIM	Receive FIFO Interrupt Source Select 0: An interrupt occurs when the condition set by the RFIGCV[2:0] bits is met. 1: An interrupt occurs each time a message has been received.
11	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
10 to 8	RFDC[2:0]	Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7	Reserved	This bit is read as the value after reset. The write value should be the value after reset.

Bit	Bit Name	Function
6 to 4	RFPLS[2:0]	Receive FIFO Buffer Payload Storage Size Select b6 b5 b4 0 0 0: 8 bytes 0 0 1: 12 bytes 0 1 0: 16 bytes 0 1 1: 20 bytes 1 0 0: 24 bytes 1 0 1: 32 bytes 1 1 0: 48 bytes 1 1 1: 64 bytes
3, 2	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
1	RFIE	Receive FIFO Interrupt Enable 0: Receive FIFO interrupt is disabled. 1: Receive FIFO interrupt is enabled.
0	RFE	Receive FIFO Buffer Enable 0: No receive FIFO buffer is used. 1: Receive FIFO buffers are used.

RFIGCV[2:0] Bits

These bits are used to specify the number of received messages for generating a receive FIFO interrupt request when the RFIM bit is set to 0 with a fraction for the total number of buffers (the setting of RFDC[2:0]).
When the RFDC[2:0] bits are set to 001_B (4 messages), set the RFIGCV[2:0] bits to 001_B, 011_B, 101_B, or 111_B. Modify these bits only in global reset mode.

RFIM Bit

This bit is used to select a FIFO interrupt source. Modify this bit only in global reset mode.

RFDC[2:0] Bits

These bits are used to select the number of messages that can be stored in a single receive FIFO buffer. When these bits are set to 000_B, no receive FIFO buffer should be used. Modify these bits only in global reset mode.

RFPLS[2:0] Bits

These bits are used to select the maximum payload size that can be stored in the receive FIFO buffer. Modify these bits only in global reset mode.

RFIE Bit

Setting the RFIE bit to 1 enables receive FIFO interrupts. Modify this bit when the RFE bit set to 0 (no receive FIFO buffer is used).

RFE Bit

Setting the RFE bit to 1 makes receive FIFO buffers available. Clearing this bit to 0 sets the RFEMP flag in the RSCFDnCFDRFSTSx register to 1 (buffer empty). Modify this bit in global operating mode or global test mode. Set this bit to 1 with another instruction after the settings to all bits in the RSCFDnCFDRFCCx register have been done. This bit is cleared to 0 in global reset mode.

25.4.7.2 RSCFDnCFDRFSTSx — Receive FIFO Buffer Status Register (x = 0 to 7)

Access: RSCFDnCFDRFSTSx register can be read/written in 32-bit units
RSCFDnCFDRFSTSxL, RSCFDnCFDRFSTSxH registers can be read/written in 16-bit units
RSCFDnCFDRFSTSxLL, RSCFDnCFDRFSTSxLH, RSCFDnCFDRFSTSxHL, RSCFDnCFDRFSTSxHH registers can be read/written in 8-bit units

Address: RSCFDnCFDRFSTSx: <RSCFDn_base> + $00D8_H$ + (04_H × x)
RSCFDnCFDRFSTSxL: <RSCFDn_base> + $00D8_H$ + (04_H × x),
RSCFDnCFDRFSTSxH: <RSCFDn_base> + $00DA_H$ + (04_H × x)
RSCFDnCFDRFSTSxLL: <RSCFDn_base> + $00D8_H$ + (04_H × x),
RSCFDnCFDRFSTSxLH: <RSCFDn_base> + $00D9_H$ + (04_H × x),
RSCFDnCFDRFSTSxHL: <RSCFDn_base> + $00DA_H$ + (04_H × x),
RSCFDnCFDRFSTSxHH: <RSCFDn_base> + $00DB_H$ + (04_H × x)

Value after reset: $0000\ 0001_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMC[7:0]								—	—	—	—	RFIF	RFMLT	RFFLL	RFEMP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.123 RSCFDnCFDRFSTSx Register Contents

Bit	Bit Name	Function
31 to 16	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
15 to 8	RFMC[7:0]	Receive FIFO Unread Message Counter The number of unread messages stored in the receive FIFO buffer is displayed.
7 to 4	Reserved	These bits are read as the value after reset. When writing to these bits, write the value after reset.
3	RFIF	Receive FIFO Interrupt Request Flag 0: No receive FIFO interrupt request is present. 1: A receive FIFO interrupt request is present.
2	RFMLT	Receive FIFO Message Lost Flag 0: No receive FIFO message is lost. 1: A receive FIFO message is lost.
1	RFFLL	Receive FIFO Buffer Full Status Flag 0: The receive FIFO buffer is not full. 1: The receive FIFO buffer is full.
0	RFEMP	Receive FIFO Buffer Empty Status Flag 0: The receive FIFO buffer contains unread message. 1: The receive FIFO buffer contains no unread message (buffer empty).

RFMC[7:0] Flag

This flag indicates the number of unread messages in the receive FIFO buffer. This flag becomes 00_H when the RFE bit in the RSCFDnCFDRFCCx register is set to 0.
This flag is 00_H in global reset mode.

RFIF Flag

This flag is set to 1 when the receive FIFO interrupt request generation conditions set by the RFIGCV[2:0] bits and the RFIM bit in the RSCFDnCFDRFCCx register are met. This flag is cleared to 0 in global reset mode or by writing 0 to this flag. Modify this bit in global operating mode or global test mode.
To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

RFMLT Flag

This flag is set to 1 when an attempt is made to store a new message while the receive FIFO buffer is full. In this case, the new message is discarded.
This flag is cleared to 0 in global reset mode or by writing 0 to this flag.
Modify this bit in global operating mode or global test mode.
To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

RFFLL Flag

This flag is set to 1 when the number of messages stored in the receive FIFO buffer matches the FIFO buffer depth set by the RFDC[2:0] bits in the RSCFDnCFDRFCCx register.
If the number of messages stored in the receive FIFO buffer becomes smaller than the FIFO buffer depth set by the RFDC[2:0] bits, this flag is cleared to 0. This flag is also cleared to 0 when the RFE bit in the RSCFDnCFDRFCCx register is set to 0 (no receive FIFO buffer is used) or in global reset mode.

RFEMP Flag

This flag is set to 1 when all messages in the receive FIFO buffer have been read. This flag is also set to 1 when the RFE bit in the RSCFDnCFDRFCCx register is 0 or in global reset mode.
This flag is cleared to 0 when even a single received message has been stored in the receive FIFO buffer.

Note: To clear the RFMLT or RFIF flag to 0, use a store instruction to write “0” to the given flag and “1” to the other flags.

25.4.7.3 RSCFDnCFDRFPCTRx — Receive FIFO Buffer Pointer Control Register (x = 0 to 7)

Access: RSCFDnCFDRFPCTRx register can only be written in 32-bit units
RSCFDnCFDRFPCTRxL, RSCFDnCFDRFPCTRxH registers can only be written in 16-bit units
RSCFDnCFDRFPCTRxLL, RSCFDnCFDRFPCTRxLH, RSCFDnCFDRFPCTRxHL, RSCFDnCFDRFPCTRxHH registers can only be written in 8-bit units

Address: RSCFDnCFDRFPCTRx: <RSCFDn_base> + $00F8_H$ + (04_H × x)
RSCFDnCFDRFPCTRxL: <RSCFDn_base> + $00F8_H$ + (04_H × x),
RSCFDnCFDRFPCTRxH: <RSCFDn_base> + $00FA_H$ + (04_H × x)
RSCFDnCFDRFPCTRxLL: <RSCFDn_base> + $00F8_H$ + (04_H × x),
RSCFDnCFDRFPCTRxLH: <RSCFDn_base> + $00F9_H$ + (04_H × x),
RSCFDnCFDRFPCTRxHL: <RSCFDn_base> + $00FA_H$ + (04_H × x),
RSCFDnCFDRFPCTRxHH: <RSCFDn_base> + $00FB_H$ + (04_H × x)

Value after reset: 0000 0000H

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RFPC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 25.124 RSCFDnCFDRFPCTRx Register Contents

Bit	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	RFPC[7:0]	Receive FIFO Pointer Control When these bits are set to FF_H, the read pointer moves to the next unread message in the receive FIFO buffer.

When the RFDMAEx value in the RSCFDnCFDCDTCT register is 1 (DMA transfer request enabled), do not write a value to this register.

RFPC[7:0] Bits

When the RFPC[7:0] bits are set to FF_H, the read pointer moves to the next unread message in the receive FIFO buffer. At this time, the RFMC[7:0] (receive FIFO unread message counter) value in the RSCFDnCFDRFSTSx register is decremented. Read the RSCFDnCFDRFIDx, RSCFDnCFDRFPTRx, RSCFDnCFDRFDSTSx, and RSCFDnCFDRFDFd_x registers to read messages in the receive FIFO buffer, and then write FF_H to the RFPC[7:0] bits. When writing FF_H to these bits, make sure that the RFE bit in the RSCFDnCFDRFCCx register is set to 1 (receive FIFO buffers are used) and the RFEMP flag in the RSCFDnCFDRFSTSx register is 0 (the receive FIFO buffer contains unread messages).

25.4.7.4 RSCFDnCFDRFIDx — Receive FIFO Buffer Access ID Register (x = 0 to 7)

Access: RSCFDnCFDRFIDx register can be read only in 32-bit units
RSCFDnCFDRFIDxL, RSCFDnCFDRFIDxH registers can be read only in 16-bit units
RSCFDnCFDRFIDxLL, RSCFDnCFDRFIDxLH, RSCFDnCFDRFIDxHL, RSCFDnCFDRFIDxHH registers can be read only in 8-bit units

Address: RSCFDnCFDRFIDx: <RSCFDn_base> + 3000_H + (80_H × x)
RSCFDnCFDRFIDxL: <RSCFDn_base> + 3000_H + (80_H × x),
RSCFDnCFDRFIDxH: <RSCFDn_base> + 3002_H + (80_H × x)
RSCFDnCFDRFIDxLL: <RSCFDn_base> + 3000_H + (80_H × x),
RSCFDnCFDRFIDxLH: <RSCFDn_base> + 3001_H + (80_H × x),
RSCFDnCFDRFIDxHL: <RSCFDn_base> + 3002_H + (80_H × x),
RSCFDnCFDRFIDxHH: <RSCFDn_base> + 3003_H + (80_H × x)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFIDE	RFRTR	—	RFID[28:16]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFID[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.125 RSCFDnCFDRFPCTRx Register Contents

Bit	Bit Name	Function
31	RFIDE	Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	RFRTR	Receive FIFO Buffer RTR/RRS • When the received message is a classical CAN frame 0: Data frame 1: Remote frame • When the received message is a CANFD frame The RRS bit value of the received message can be read.
29	Reserved	This bit is read as the value after reset.
28 to 0	RFID[28:0]	Receive FIFO Buffer ID Data The standard ID or extended ID of received message can be read. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive FIFO buffer.

RFRTR Bit

When the received message is a classical CAN frame, this bit indicates the frame format (data frame or remote frame) of the message stored in the receive FIFO buffer. When the received message is a CANFD frame, this bit indicates the RRS bit value in the message.

RFID[28:0] Bits

These bits indicate the ID of the message stored in the receive FIFO buffer.

25.4.7.5 RSCFDnCFDRFPTRx — Receive FIFO Buffer Access Pointer Register (x = 0 to 7)

Access: RSCFDnCFDRFPTRx register can be read only in 32-bit units
RSCFDnCFDRFPTRxL, RSCFDnCFDRFPTRxH registers can be read only in 16-bit units
RSCFDnCFDRFPTRxLL, RSCFDnCFDRFPTRxLH, RSCFDnCFDRFPTRxHL, RSCFDnCFDRFPTRxHH registers can be read only in 8-bit units

Address: RSCFDnCFDRFPTRx: <RSCFDn_base> + 3004_H + (80_H × x)
RSCFDnCFDRFPTRxL: <RSCFDn_base> + 3004_H + (80_H × x),
RSCFDnCFDRFPTRxH: <RSCFDn_base> + 3006_H + (80_H × x)
RSCFDnCFDRFPTRxLL: <RSCFDn_base> + 3004_H + (80_H × x),
RSCFDnCFDRFPTRxLH: <RSCFDn_base> + 3005_H + (80_H × x),
RSCFDnCFDRFPTRxHL: <RSCFDn_base> + 3006_H + (80_H × x),
RSCFDnCFDRFPTRxHH: <RSCFDn_base> + 3007_H + (80_H × x)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDLC[3:0]				RFPTR[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFTS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.126 RSCFDnCFDRFPTRx Register Contents

Bit	Bit Name	Function
31 to 28	RFDLC[3:0]	Receive FIFO Buffer DLC Data

b31 b30 b29 b28	Classical CAN Frame	CANFD Frame
0 0 0 0	0 data bytes	
0 0 0 1	1 data byte	
0 0 1 0	2 data bytes	
0 0 1 1	3 data bytes	
0 1 0 0	4 data bytes	
0 1 0 1	5 data bytes	
0 1 1 0	6 data bytes	
0 1 1 1	7 data bytes	
1 0 0 0	8 data bytes	
1 0 0 1	8 data bytes	12 data bytes
1 0 1 0		16 data bytes
1 0 1 1		20 data bytes
1 1 0 0		24 data bytes
1 1 0 1		32 data bytes
1 1 1 0		48 data bytes
1 1 1 1		64 data bytes

27 to 16	RFPTR[11:0]	Receive FIFO Buffer Label Data Label information of the received message can be read.
15 to 0	RFTS[15:0]	Receive FIFO Buffer Timestamp Data Timestamp value of the received message can be read.

RFDLC[3:0] Bits

These bits contain the data length of the message stored in the receive FIFO buffer.

RFPTR[11:0] Bits

These bits contain the label information of the message stored in the receive FIFO buffer.

RFTS[15:0] Bits

These bits contain the timestamp value of the message stored in the receive FIFO buffer.

25.4.7.6 RSCFDnCFDRFFDSTSx — Receive FIFO CANFD Status Register (x = 0 to 7)

Access: RSCFDnCFDRFFDSTSx register can be read only in 32-bit units
RSCFDnCFDRFFDSTSxL, RSCFDnCFDRFFDSTSxH registers can be read only in 16-bit units
RSCFDnCFDRFFDSTSxLL, RSCFDnCFDRFFDSTSxLH, RSCFDnCFDRFFDSTSxHL, RSCFDnCFDRFFDSTSxHH registers can be read only in 8-bit units

Address: RSCFDnCFDRFFDSTSx: <RSCFDn_base> + 3008_H + (80_H × x)
RSCFDnCFDRFFDSTSxL: <RSCFDn_base> + 3008_H + (80_H × x),
RSCFDnCFDRFFDSTSxH: <RSCFDn_base> + $300A_H$ + (80_H × x)
RSCFDnCFDRFFDSTSxLL: <RSCFDn_base> + 3008_H + (80_H × x),
RSCFDnCFDRFFDSTSxLH: <RSCFDn_base> + 3009_H + (80_H × x),
RSCFDnCFDRFFDSTSxHL: <RSCFDn_base> + $300A_H$ + (80_H × x),
RSCFDnCFDRFFDSTSxHH: <RSCFDn_base> + $300B_H$ + (80_H × x)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RFFDF	RFBRS	RFESI
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.127 RSCFDnCFDRFFDSTSx Register Contents

Bit	Bit Name	Function
31 to 3	Reserved	These bits are read as the value after reset.
2	RFFDF	FDF 0: Classical CAN frame 1: CANFD frame
1	RFBRS	BRS 0: The bit rate in the data area does not change. 1: The bit rate in the data area changes.
0	RFESI	ESI 0: Error active node 1: Error passive node

RFFDF Bit

This bit indicates the FD format (classical CAN frame or CANFD frame) of the message stored in the receive FIFO buffer.

RFBRS Bit

When the RFFDF bit is 1, this bit indicates the BRS bit value of the message stored in the receive FIFO buffer. When the RFFDF bit is 0, this bit is always read as 0.

RFESI Bit

When the RFFDF bit is 1, this bit indicates the ESI bit value of the message stored in the receive FIFO buffer. When the RFFDF bit is 0, this bit is always read as 0.

25.4.7.7 RSCFDnCFDRFDFd_x — Receive FIFO Buffer Access Data Field d Register (d = 0 to 15, x = 0 to 7)

Access: RSCFDnCFDRFDFd_x register can be read only in 32-bit units
RSCFDnCFDRFDFd_xL, RSCFDnCFDRFDFd_xH registers can be read only in 16-bit units
RSCFDnCFDRFDFd_xLL, RSCFDnCFDRFDFd_xLH, RSCFDnCFDRFDFd_xHL, RSCFDnCFDRFDFd_xHH registers can be read only in 8-bit units

Address: RSCFDnCFDRFDFd_x: <RSCFDn_base> + $300C_H$ + (04_H × d) + (80_H × x)
RSCFDnCFDRFDFd_xL: <RSCFDn_base> + $300C_H$ + (04_H × d) + (80_H × x),
RSCFDnCFDRFDFd_xH: <RSCFDn_base> + $300E_H$ + (04_H × d) + (80_H× x)
RSCFDnCFDRFDFd_xLL: <RSCFDn_base> + $300C_H$ + (04_H × d) + (80_H × x),
RSCFDnCFDRFDFd_xLH: <RSCFDn_base> + $300D_H$ + (04_H × d) + (80_H × x),
RSCFDnCFDRFDFd_xHL: <RSCFDn_base> + $300E_H$ + (04_H× d) + (80_H × x),
RSCFDnCFDRFDFd_xHH: <RSCFDn_base> + $300F_H$ + (04_H × d) + (80_H × x)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB4 × d + 3 [7:0]								RFDB4 × d + 2 [7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB4 × d + 1 [7:0]								RFDB4 × d + 0 [7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.128 RSCFDnCFDRFDFd_x Register Contents

Bit	Bit Name	Function
31 to 24	RFDB4 × d + 3 [7:0]	Receive Buffer Data Byte 4 × d + 3
23 to 16	RFDB4 × d + 2 [7:0]	Receive Buffer Data Byte 4 × d + 2
15 to 8	RFDB4 × d + 1 [7:0]	Receive Buffer Data Byte 4 × d + 1
7 to 0	RFDB4 × d + 0 [7:0]	Receive Buffer Data Byte 4 × d + 0 Data for a message stored in the receive buffer can be read.

When the RFDLC[3:0] value in the RSCFDnCFDRFPTRx register is smaller than the payload storage size of the receive FIFO buffer, data bytes for which no data is set are read as 00_H.
Specify the payload storage size of the receive FIFO buffer by the RFPLS[2:0] bits in the RSCFDnCFDRFCCx register.
Do not read or write the RSCFDnCFDRFDFd_x register corresponding to an area larger than the specified size.

25.4.8 Transmit/Receive FIFO Buffer-related Registers

25.4.8.1 RSCFDnCFDCFCCk — Transmit/receive FIFO Buffer Configuration and Control Register k (k = 0 to 5)

Access: RSCFDnCFDCFCCk register can be read/written in 32-bit units
RSCFDnCFDCFCCkL, RSCFDnCFDCFCCkH registers can be read/written in 16-bit units
RSCFDnCFDCFCCkLL, RSCFDnCFDCFCCkLH, RSCFDnCFDCFCCkHL, RSCFDnCFDCFCCkHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCFCCk: <RSCFDn_base> + 0118_H + (04_H × k)
RSCFDnCFDCFCCkL: <RSCFDn_base> + 0118_H + (04_H × k),
RSCFDnCFDCFCCkH: <RSCFDn_base> + $011A_H$ + (04_H × k)
RSCFDnCFDCFCCkLL: <RSCFDn_base> + 0118_H + (04_H × k),
RSCFDnCFDCFCCkLH: <RSCFDn_base> + 0119_H + (04_H × k),
RSCFDnCFDCFCCkHL: <RSCFDn_base> + $011A_H$ + (04_H × k),
RSCFDnCFDCFCCkHH: <RSCFDn_base> + $011B_H$ + (04_H × k)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFITT[7:0]								CFTML[3:0]				CFITR	CFITSS	CFM[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFIGCV[2:0]			CFIM	—	CFDC[2:0]			—	CFPLS[2:0]			—	CFTXIE	CFRXIE	CFE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 25.129 RSCFDnCFDCFCCk Register Contents

Bit	Bit Name	Function
31 to 24	CFITT[7:0]	Set a message transmission interval. Set Value: 00_H to FF_H
23 to 20	CFTML[3:0]	Transmit Buffer Link Configuration Set the transmit buffer number to be linked to the transmit/receive FIFO buffer.
19	CFITR	Transmit/Receive FIFO Interval Timer Resolution 0: Clock dividing pclk by (ITRCP[15:0] bits) 1: Clock dividing pclk by (ITRCP[15:0] bits × 10)
18	CFITSS	Transmit/Receive FIFO Interval Timer Clock Source Select 0: Interval timer clock source selected by the CFITR bit 1: Interval timer clock source is the nominal bit time clock for the channel to which the FIFO is linked.
17 to 16	CFM[1:0]	Transmit/Receive FIFO Mode Select b17 b16 0 0: Receive mode 0 1: Transmit mode 1 0: Gateway mode 1 1: Setting prohibited
15 to 13	CFIGCV[2:0]	Transmit/Receive FIFO Receive Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.

Bit	Bit Name	Function
12	CFIM	Transmit/Receive FIFO Interrupt Source Select 0: • Receive mode/gateway mode When the number of received messages has met the condition set by the CFIGCV[2:0] bits, a FIFO receive interrupt request is generated. • Transmit mode/gateway mode When the buffer becomes empty upon completion of message transmission, a FIFO transmit interrupt request is generated. 1: • Receive mode/gateway mode A FIFO receive interrupt request is generated each time a message has been received. • Transmit mode/gateway mode A FIFO transmit interrupt request is generated each time a message has been transmitted.
11	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
10 to 8	CFDC[2:0]	Transmit/Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
6 to 4	CFPLS[2:0]	Transmit/Receive FIFO Buffer Payload Storage Size Select b6 b5 b4 0 0 0: 8 bytes 0 0 1: 12 bytes 0 1 0: 16 bytes 0 1 1: 20 bytes 1 0 0: 24 bytes 1 0 1: 32 bytes 1 1 0: 48 bytes 1 1 1: 64 bytes
3	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
2	CFTXIE	Transmit/Receive FIFO Transmit Interrupt Enable 0: Transmit/receive FIFO transmit interrupt is disabled. 1: Transmit/receive FIFO transmit interrupt is enabled.
1	CFRXIE	Transmit/Receive FIFO Receive Interrupt Enable 0: Transmit/receive FIFO receive interrupt is disabled. 1: Transmit/receive FIFO receive interrupt is enabled.
0	CFE	Transmit/Receive FIFO Buffer Enable 0: No transmit/receive FIFO buffer is used. 1: Transmit/receive FIFO buffers are used.

CFITT[7:0] Bits

These bits are used to set a message transmission interval when transmitting messages continuously from a transmit/receive FIFO buffer whose CFM[1:0] bits are set to 01_B (transmit mode) or 10_B (gateway mode).
Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITT[7:0] bits.

CFTML[3:0] Bits

These bits are used to set the number of transmit buffer on the channel which will be linked to transmit/receive FIFO buffer k when the CFM[1:0] bits are set to 01_B (transmit mode) or 10_B (gateway mode). There are three transmit/receive FIFO buffers per channel, so channel number m of FIFO buffer k is calculated as m = k/3 (integer division). The actual assigned transmit buffer number p linked to FIFO buffer k will be ((16 × m) + CFTML[3:0]).

See Table 25.11 and Table 25.12, as for the relationship between transmit/receive FIFO buffer k and transmit buffer p.
Setting the CFDC[2:0] bits to 001_B or more enables the setting of the CFTML[3:0] bits.
Do not link to any transmit buffer which is already allocated to a transmit queue on the identical channel or to another transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFITR Bit

This bit is enabled when the CFITSS bit is 0.
When this bit is 0, the interval timer clock source is the pclk/2 clock divided by the value of the ITRCP[15:0] bits in the RSCFDnCFDGCFG register.
When this bit is 1, the interval timer clock source is the pclk/2 clock divided by (the value of the ITRCP[15:0] bits in the RSCFDnCFDGCFG register × 10).
Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFITSS Bit

When this bit is 0, the clock selected by the CFITR bit is the count source of the interval timer.
When this bit is 1, the nominal bit time clock of the channel to which the FIFO is linked is the count source of the interval timer. Use this count source only for the channel that does not handle the CANFD frames.
Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFM[1:0] Bits

These bits are used to select transmit/receive FIFO mode. Modify these bits only in global reset mode.

CFIGCV[2:0] Bits

These bits are used to specify the number of received messages for generating a transmit/receive FIFO receive interrupt request when the CFM[1:0] bits are set to 00_B (receive mode) or 10_B (gateway mode) and the CFIM bit is set to 0 with a fraction for the total number of buffers (the setting of CFDC[2:0]).
When the CFDC[2:0] bits are set to 001_B (4 messages), set the CFIGCV[2:0] bits to 001_B, 011_B, 101_B, or 111_B.
Modify these bits only in global reset mode.

CFIM Bit

This bit is used to select a transmit/receive FIFO interrupt source. Modify this bit only in global reset mode.

CFDC[2:0] Bits

These bits are used to set the number of messages that can be stored in a single transmit/receive FIFO buffer. When these bits are set to 000_B, do not use a transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFPLS[2:0] Bits

These bits are used to select the maximum payload size that can be stored in the transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFTXIE Bit

When this bit is set to 1 and the CFTXIF flag in the RSCFDnCFDCFSTSk register is set to 1, a transmit/receive FIFO transmit interrupt request is generated.
Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used).

CFRXIE Bit

When this bit is set to 1 and the CFRXIF flag in the RSCFDnCFDCFSTSk register is set to 1, a transmit/receive FIFO receive interrupt request is generated.
Modify this bit with the CFE bit set to 0.

CFE Bit

Setting this bit to 1 makes transmit/receive FIFO buffers available.
When this bit is set to 0 in transmit mode or gateway mode, if a message in the transmit/receive FIFO buffer is being transmitted or will be transmitted next, the transmit/receive FIFO buffer becomes empty after completion of transmission of that message, or upon detection of a CAN bus error, or arbitration-lost. In other cases or in receive mode, the transmit/receive FIFO buffer becomes empty immediately.
This bit is cleared to 0 when the following conditions are met.

- Receive mode: Global reset mode
- Transmit mode or gateway mode: Channel reset mode

Modify this bit in the following mode.

- Receive mode: Global operating mode or global test mode
- Transmit mode or gateway mode: Channel communication mode or channel halt mode

After all other bits in the RSCFDnCFDCFCCk register have been set, set this bit to 1 by using another instruction.

25.4.8.2 RSCFDnCFDCFSTSk — Transmit/receive FIFO Buffer Status Register (k = 0 to 5)

Access: RSCFDnCFDCFSTSk register can be read/written in 32-bit units
RSCFDnCFDCFSTSkL, RSCFDnCFDCFSTSkH registers can be read/written in 16-bit units
RSCFDnCFDCFSTSkLL, RSCFDnCFDCFSTSkLH, RSCFDnCFDCFSTSkHL, RSCFDnCFDCFSTSkHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCFSTSk: <RSCFDn_base> + 0178_H + (04_H × k)
RSCFDnCFDCFSTSkL: <RSCFDn_base> + 0178_H + (04_H × k),
RSCFDnCFDCFSTSkH: <RSCFDn_base> + $017A_H$ + (04_H × k)
RSCFDnCFDCFSTSkLL: <RSCFDn_base> + 0178_H + (04_H × k),
RSCFDnCFDCFSTSkLH: <RSCFDn_base> + 0179_H + (04_H × k),
RSCFDnCFDCFSTSkHL: <RSCFDn_base> + $017A_H$ + (04_H × k),
RSCFDnCFDCFSTSkHH: <RSCFDn_base> + $017B_H$ + (04_H × k)

Value after reset: $0000\ 0001_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFMC[7:0]								—	—	—	CFTXIF	CFRXIF	CFMLT	CFFLL	CFEMP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.130 RSCFDnCFDCFSTSk Register Contents

Bit	Bit Name	Function
31 to 16	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
15 to 8	CFMC[7:0]	Transmit/Receive FIFO Message Counter The number of messages stored in the transmit/receive FIFO buffer.
7 to 5	Reserved	These bits are read as the value after reset. When writing to these bits, write the value after reset.
4	CFTXIF	Transmit/Receive FIFO Transmit Interrupt Request Flag 0: No transmit/receive FIFO transmit interrupt request is present. 1: A transmit/receive FIFO transmit interrupt request is present.
3	CFRXIF	Transmit/Receive FIFO Receive Interrupt Request Flag 0: No transmit/receive FIFO receive interrupt request is present. 1: A transmit/receive FIFO receive interrupt request is present.
2	CFMLT	Transmit/Receive FIFO Message Lost Flag 0: No transmit/receive FIFO message is lost. 1: A transmit/receive FIFO message is lost.
1	CFFLL	Transmit/Receive FIFO Buffer Full Status Flag 0: The transmit/receive FIFO buffer is not full. 1: The transmit/receive FIFO buffer is full.
0	CFEMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: The transmit/receive FIFO buffer contains messages. 1: The transmit/receive FIFO buffer contains no message (buffer empty).

CFMC[7:0] Bits

The CFMC[7:0] bits indicate the following values that depend on the setting of the CFM[1:0] bits in the RSCFDnCFDCFCCk register.

- When CFM[1:0] value is 01_B (transmit mode): Number of untransmitted messages in the buffer
- When CFM[1:0] value is 00_B (receive mode): Number of unread received messages in the buffer
- When CFM[1:0] value is 10_B (gateway mode): Number of untransmitted received messages in the buffer

These bits are cleared to 0 when any of the following conditions is met.

- When CFM[1:0] value is 00_B: In global reset mode
- When CFM[1:0] value is 01_B or 10_B: In channel reset mode
- When the CFE bit in the RSCFDnCFDCFCCk register is cleared to 0.

CFTXIF Flag

The CFTXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 01_B or 10_B, and the factor selected by the CFIM bit in the RSCFDnCFDCFCCk register occurs

The CFTXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFTXIF flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.
To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFRXIF Flag

The CFRXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00B or 10B, and the factor selected by the CFIM bit in the RSCFDnCFDCFCCk register occurs

The CFRXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFRXIF flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.
To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFMLT Flag

The CFMLT flag is set to 1 when any of the following conditions is met.

- When an attempt is made to store a new message while the transmit/receive FIFO buffer is full. In this case, the new message is discarded.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFMLT flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFFLL Flag

The CFFLL flag is set to 1 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer matches the FIFO buffer depth set by the CFDC[2:0] bits in the RSCFDnCFDCFCCk register.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer becomes smaller than the FIFO buffer depth set by the CFDC[2:0] bits.
- When the CFE bit in the RSCFDnCFDCFCCk register is 0 (no transmit/receive FIFO buffer is used): When not in the transmit abort
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

CFEMP Flag

The CFEMP flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B: All messages have been read, or in global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: All messages have been transmitted, or in channel reset mode
- When the CFE bit is 0 (no transmit/receive FIFO buffer is used): Not in the transmit abort

The CFEMP flag is cleared to 0 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B or 10_B: At least one received message has been stored in the transmit/receive FIFO buffer.
- When the CFM[1:0] bits are set to 01_B: A value of FF_H has been written to the RSCFDnCFDCFPCTRk register after data was written to the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDFd_k registers.

Note: To clear CFTXIF, CFRXIF, or CFMLT flag to 0, the program must write 0. When writing, use a store instruction to write “0” to the given flag and “1” to other flags.

25.4.8.3 RSCFDnCFDCFPCTRk — Transmit/receive FIFO Buffer Pointer Control Register (k = 0 to 5)

Access: RSCFDnCFDCFPCTRk register can only be written in 32-bit units
RSCFDnCFDCFPCTRkL, RSCFDnCFDCFPCTRkH registers can only be written in 16-bit units
RSCFDnCFDCFPCTRkLL, RSCFDnCFDCFPCTRkLH, RSCFDnCFDCFPCTRkHL, RSCFDnCFDCFPCTRkHH registers can only be written in 8-bit units

Address: RSCFDnCFDCFPCTRk: <RSCFDn_base> + $01D8_H$ + (04_H × k)
RSCFDnCFDCFPCTRkL: <RSCFDn_base> + $01D8_H$ + (04_H × k),
RSCFDnCFDCFPCTRkH: <RSCFDn_base> + $01DA_H$ + (04_H × k)
RSCFDnCFDCFPCTRkLL: <RSCFDn_base> + $01D8_H$ + (04_H × k),
RSCFDnCFDCFPCTRkLH: <RSCFDn_base> + $01D9_H$ + (04_H × k),
RSCFDnCFDCFPCTRkHL: <RSCFDn_base> + $01DA_H$ + (04_H × k),
RSCFDnCFDCFPCTRkHH: <RSCFDn_base> + $01DB_H$ + (04_H × k)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CFPC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 25.131 RSCFDnCFDCFPCTRk Register Contents

Bit	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	CFPC[7:0]	Transmit/Receive FIFO Pointer Control • Receive mode: Writing FF_H to these bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. • Transmit mode: Writing FF_H to these bits moves the write pointer to the next stage of the transmit/receive FIFO buffer. • Gateway mode: Setting prohibited

When the corresponding transmit/receive FIFO buffer is the first transmit/receive FIFO buffer (k = 3 × m) allocated to channel m and when the CFDMAEm bit in the RSCFDnCFDCDTCT register is 1 (DMA transfer request enabled), do not write a value to this register.

CFPC[7:0] Bits

- Receive mode (CFM[1:0] value in the RSCFDnCFDCFCCk register is 00_B):
  Writing FF_H to the CFPC[7:0] bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value (transmit/receive FIFO message counter) in the RSCFDnCFDCFSTSk register is decremented. Read the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDFd_k registers to read messages from the transmit/receive FIFO buffer, and then write FF_H to the CFPC[7:0] bits.
  When writing FF_H to these bits, make sure that the CFE bit in the RSCFDnCFDCFCCk register is set to 1 (transmit/receive FIFO buffers are used) and the CFEMP flag in the RSCFDnCFDCFSTSk register is 0 (the transmit/receive FIFO buffer contains messages).
- Transmit mode (CFM[1:0] value in the RSCFDnCFDCFCCk register is 01_B):
  Writing FF_H to the CFPC[7:0] bits stores the data written to the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDFd_k registers in the transmit/receive FIFO buffer and moves the write pointer to the next stage of the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value is incremented. Write transmit messages to the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDFd_k registers before writing FF_H to the CFPC[7:0] bits.
  When writing FF_H to these bits, make sure that the CFE bit in the RSCFDnCFDCFCCk register is set to 1 and the CFFLL flag in the RSCFDnCFDCFSTSk register is 0 (the transmit/receive FIFO buffer is not full).
- Gateway mode (CFM[1:0] value in the RSCFDnCFDCFCCk register is 10_B):
  Setting prohibited

25.4.8.4 RSCFDnCFDCFIDk — Transmit/receive FIFO Buffer Access ID Register (k = 0 to 5)

Access: RSCFDnCFDCFIDk register can be read/written in 32-bit units
RSCFDnCFDCFIDkL, RSCFDnCFDCFIDkH registers can be read/written in 16-bit units
RSCFDnCFDCFIDkLL, RSCFDnCFDCFIDkLH, RSCFDnCFDCFIDkHL, RSCFDnCFDCFIDkHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCFIDk: <RSCFDn_base> + 3400_H + (80_H × k)
RSCFDnCFDCFIDkL: <RSCFDn_base> + 3400_H + (80_H × k),
RSCFDnCFDCFIDkH: <RSCFDn_base> + 3402_H + (80_H × k)
RSCFDnCFDCFIDkLL: <RSCFDn_base> + 3400_H + (80_H × k),
RSCFDnCFDCFIDkLH: <RSCFDn_base> + 3401_H + (80_H × k),
RSCFDnCFDCFIDkHL: <RSCFDn_base> + 3402_H + (80_H × k),
RSCFDnCFDCFIDkHH: <RSCFDn_base> + 3403_H + (80_H × k)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFIDE	CFRTR	THLEN	CFID[28:16]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFID[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.132 RSCFDnCFDCFIDk Register Contents

Bit	Bit Name	Function
31	CFIDE	Transmit/Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	CFRTR	Transmit/Receive FIFO Buffer RTR/RRS • When the CFM[1:0] value is 01_B (transmit mode) – When the transmit message is a classical CAN frame 0: Data frame 1: Remote frame – When the transmit message is a CANFD frame Write 0 to this bit. • When the CFM[1:0] value is 00_B (receive mode) – When the received message is a classical CAN frame 0: Data frame 1: Remote frame – When the received message is a CANFD frame The RRS bit value of the received message can be read.
29	THLEN	Transmit History Data Store Enable This bit is valid only when the CFM[1:0] value is 01_B (transmit mode). 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	CFID[28:0]	Transmit/Receive FIFO Buffer ID Data • When CFM[1:0] value is 01_B (transmit mode): Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11. • When CFM[1:0] value is 00_B (receive mode): Standard ID or extended ID in the received message can be read. For standard ID, read bits 10 to 0. Bits 28 to 11 are read as 0.

This register is writable only when the CFM[1:0] value in the RSCFDnCFDCFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). This RSCFDnCFDCFIDk register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

CFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, this bit is used to set the ID format of the message to be transmitted from the transmit/receive FIFO buffer.

CFRTR Bit

If the the received message is a classical CAN frame, this bit indicates the data format (data frame or remote frame) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. If the received message is a CANFD frame, this bit indicates the RRS bit value of the received message.
When the CFM[1:0] value is 01_B, this bit is used to set the data format of the message to be transmitted from the transmit/receive FIFO buffer.
When the CFFDF bit in the RSCFDnCFDCFFDCSTSk register is 1 (CANFD frame), set this bit to 0.

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, buffer type, and timestamp) of transmit messages is stored in the transmit history buffer after transmission is completed.
This bit is enabled when the CFM[1:0] value is 01_B (transmit mode).

CFID[28:0] Bits

These bits contain the ID of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B.
When the CFM[1:0] value is 01_B, this bit is used to set the ID of the message to be transmitted from the transmit/receive FIFO buffer.

25.4.8.5 RSCFDnCFDCFPTRk — Transmit/receive FIFO Buffer Access Pointer Register (k = 0 to 5)

Access: RSCFDnCFDCFPTRk register can be read/written in 32-bit units
RSCFDnCFDCFPTRkL, RSCFDnCFDCFPTRkH registers can be read/written in 16-bit units
RSCFDnCFDCFPTRkLL, RSCFDnCFDCFPTRkLH, RSCFDnCFDCFPTRkHL, RSCFDnCFDCFPTRkHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCFPTRk: <RSCFDn_base> + 3404_H + (80_H × k)
RSCFDnCFDCFPTRkL: <RSCFDn_base> + 3404_H + (80_H × k),
RSCFDnCFDCFPTRkH: <RSCFDn_base> + 3406_H + (80_H × k)
RSCFDnCFDCFPTRkLL: <RSCFDn_base> + 3404_H + (80_H × k),
RSCFDnCFDCFPTRkLH: <RSCFDn_base> + 3405_H + (80_H × k),
RSCFDnCFDCFPTRkHL: <RSCFDn_base> + 3406_H + (80_H × k),
RSCFDnCFDCFPTRkHH: <RSCFDn_base> + 3407_H + (80_H × k)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDLC[3:0]				CFPTR[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFTS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.133 RSCFDnCFDCFPTRk Register Contents

Bit	Bit Name	Function
31 to 28	CFDLC[3:0]	Transmit/Receive FIFO Buffer DLC Data

b31 b30 b29 b28	Classical CAN Frame	CANFD Frame
0 0 0 0	0 data bytes	
0 0 0 1	1 data byte	
0 0 1 0	2 data bytes	
0 0 1 1	3 data bytes	
0 1 0 0	4 data bytes	
0 1 0 1	5 data bytes	
0 1 1 0	6 data bytes	
0 1 1 1	7 data bytes	
1 0 0 0	8 data bytes	
1 0 0 1	8 data bytes	12 data bytes
1 0 1 0		16 data bytes
1 0 1 1		20 data bytes
1 1 0 0		24 data bytes
1 1 0 1		32 data bytes
1 1 1 0		48 data bytes
1 1 1 1		64 data bytes

Bit	Bit Name	Function
27 to 16	CFPTR[11:0]	Transmit/Receive FIFO Buffer Label Data • When CFM[1:0] value is 01_B (transmit mode): Set the label information to be stored in the transmit history buffer. Only bits CFPTR[7:0] are valid. • When CFM[1:0] value is 00_B (receive mode): The label information of the received message can be read.
15 to 0	CFTS[15:0]	Transmit/Receive FIFO Buffer Timestamp Data These bits are valid only when the CFM[1:0] value is 00_B (receive mode). The timestamp value of the received message can be read.

This register is writable only when the CFM[1:0] value in the RSCFDnCFDCFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

CFDLC[3:0] Bits

These bits indicate the data length of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B.
When the CFM[1:0] value is 01_B, these bits are used to set the data length of the message to be transmitted from the transmit/receive FIFO buffer.
When the CFDLC[3:0] bits are set to 1001_B or more while the CFFDF bit in the RSCFDnCFDCFFDCSTSk register is 0 (CAN frame), 8-byte data is transmitted actually. When the CFFDF bit is 1 (CANFD frame), the settable value range varies depending on the settings of the TMME bit in the RSCFDnCFDCmFDCFG register and the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

- When TMME bit = 0 (transmit buffer merge mode disabled):
  A value of 0000_B to 1111_B is settable. If the specified data length exceeds the payload storage size specified by the CFPLS[2:0] bits, excessive payloads are padded by CCH.
- When TMME bit = 1 (transmit buffer merge mode enabled):
  Set the data length within the payload storage size specified by the CFPLS[2:0] bits.

CFPTR[11:0] Bits

These bits indicate the label information attached to the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, the CFPTR[7:0] value is stored in the transmit history buffer when message transmission has been completed.

CFTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the transmit/receive FIFO buffer.
These bits are valid when the CFM[1:0] value is 00_B.

25.4.8.6 RSCFDnCFDCFFDCSTSk — Transmit/Receive FIFO CANFD Configuration/ Status Register (k = 0 to 5)

Access: RSCFDnCFDCFFDCSTSk register can be read/written in 32-bit units
RSCFDnCFDCFFDCSTSkL, RSCFDnCFDCFFDCSTSkH registers can be read/written in 16-bit units
RSCFDnCFDCFFDCSTSkLL, RSCFDnCFDCFFDCSTSkLH, RSCFDnCFDCFFDCSTSkHL, RSCFDnCFDCFFDCSTSkHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCFFDCSTSk: <RSCFDn_base> + 3408_H + (80_H × k)
RSCFDnCFDCFFDCSTSkL: <RSCFDn_base> + 3408_H + (80_H × k),
RSCFDnCFDCFFDCSTSkH: <RSCFDn_base> + $340A_H$ + (80_H × k)
RSCFDnCFDCFFDCSTSkLL: <RSCFDn_base> + 3408_H + (80_H × k),
RSCFDnCFDCFFDCSTSkLH: <RSCFDn_base> + 3409_H + (80_H × k),
RSCFDnCFDCFFDCSTSkHL: <RSCFDn_base> + $340A_H$ + (80_H × k),
RSCFDnCFDCFFDCSTSkHH: <RSCFDn_base> + $340B_H$ + (80_H × k)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CFFDF	CFBRS	CFESI
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 25.134 RSCFDnCFDCFFDCSTSk Register Contents

Bit	Bit Name	Function
31 to 3	Reserved	These bits are read as the value after reset.
2	CFFDF	FDF 0: Classical CAN frame 1: CANFD frame
1	CFBRS	BRS 0: The bit rate in the data area does not change. 1: The bit rate in the data area changes.
0	CFESI	ESI 0: Error active node 1: Error passive node

This register is writable only when the CFM[1:0] value in the RSCFDnCFDCFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). Do not read or write this register when the CFM[1:0] value is 10_B (gateway mode).

CFFDF Bit

When the CFM[1:0] value is 00_B, this bit indicates the FD format (classical CAN frame or CANFD frame) of the message stored in the transmit/receive FIFO buffer. When the CFM[1:0] value is 01_B, this bit is used to set the FD format of the message to be transmitted from the transmit/receive FIFO buffer.

CFBRS Bit

When the CFM[1:0] value is 00_B, if the CFFDF bit is 1, this bit indicates the BRS bit value of the message stored in the transmit/receive FIFO buffer. If the CFFDF bit is 0, this bit is always read as 0.
When the CFM[1:0] value is 01_B, if the CFFDF bit is 1, this bit is used to set the BRS bit value of the message to be transmitted from the transmit/receive FIFO buffer. If the CFFDF bit is 0, write 0 to this bit.

CFESI Bit

When the CFM[1:0] value is 00B, if the CFFDF bit is 1, this bit indicates the ESI bit value of the message stored in the transmit/receive FIFO buffer. If the CFFDF bit is 0, this bit is always read as 0.
When the CFM[1:0] value is 01B, if the CFFDF bit is 1, this bit is used to set the ESI bit value of the message to be transmitted from the transmit/receive FIFO buffer. The set value is transmitted when the ESIC bit in the RSCFDnCFDCmFDCFG register is 1 and the channel is in the error active state. When the channel is in the error passive state, the ESI bit value that shows an error passive node is transmitted regardless of this bit value. When the CFFDF bit is 0, write 0 to this bit.

25.4.8.7 RSCFDnCFDCFDFd_k — Transmit/receive FIFO Buffer Access Data Field d Register (d = 0 to 15, k = 0 to 5)

Access: RSCFDnCFDCFDFd_k register can be read/written in 32-bit units
RSCFDnCFDCFDFd_kL, RSCFDnCFDCFDFd_kH registers can be read/written in 16-bit units
RSCFDnCFDCFDFd_kLL, RSCFDnCFDCFDFd_kLH, RSCFDnCFDCFDFd_kHL, RSCFDnCFDCFDFd_kHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCFDFd_k: <RSCFDn_base> + $340C_H$ + (04_H × d) + (80_H × k)
RSCFDnCFDCFDFd_kL: <RSCFDn_base> + $340C_H$ + (04_H × d) + (80_H × k),
RSCFDnCFDCFDFd_kH: <RSCFDn_base> + $340E_H$ + (04_H × d) + (80_H × k)
RSCFDnCFDCFDFd_kLL: <RSCFDn_base> + $340C_H$ + (04_H × d) + (80_H × k),
RSCFDnCFDCFDFd_kLH: <RSCFDn_base> + $340D_H$ + (04_H × d) + (80_H × k),
RSCFDnCFDCFDFd_kHL: <RSCFDn_base> + $340E_H$ + (04_H × d) + (80_H × k),
RSCFDnCFDCFDFd_kHH: <RSCFDn_base> + $340F_H$ + (04_H × d) + (80_H × k)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB4 × d + 3 [7:0]								CFDB4 × d + 2 [7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB4 × d + 1 [7:0]								CFDB4 × d + 1 [7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.135 RSCFDnCFDCFDFd_k Register Contents

Bit	Bit Name	Function
31 to 24	CFDB4 × d + 3 [7:0]	Transmit/Receive FIFO Buffer Data Byte 4 × d + 3 Transmit/Receive FIFO Buffer Data Byte 4 × d + 2 Transmit/Receive FIFO Buffer Data Byte 4 × d + 1 Transmit/Receive FIFO Buffer Data Byte 4 × d + 0 • When CFM[1:0] value is 01_B (transmit mode): Set the transmit/receive FIFO buffer data. • When CFM[1:0] value is 00_B (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.
23 to 16	CFDB4 × d + 2 [7:0]	
15 to 8	CFDB4 × d + 1 [7:0]	
7 to 0	CFDB4 × d + 0 [7:0]	

This register is writable only when the CFM[1:0] value in the RSCFDnCFDCFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). When the CFDLC[3:0] value in the RSCFDnCFDCFPTRk register is smaller than the payload storage size of the transmit/receive FIFO buffer, data bytes for which no data is set are read as 00_H.
Specify the payload storage size of the transmit/receive FIFO buffer by the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register. Do not read or write the RSCFDnCFDCFDFd_k register corresponding to an area larger than the specified size. This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

25.4.9 Details of FIFO Status-related Registers

25.4.9.1 RSCFDnCFDFESTS — FIFO Empty Status Register

Access: RSCFDnCFDFESTS register can be read only in 32-bit units
RSCFDnCFDFESTSL, RSCFDnCFDFESTSH registers can be read only in 16-bit units
RSCFDnCFDFESTSLL, RSCFDnCFDFESTSLH, RSCFDnCFDFESTSHL, RSCFDnCFDFESTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDFESTS: <RSCFDn_base> + 0238_H
RSCFDnCFDFESTSL: <RSCFDn_base> + 0238_H, RSCFDnCFDFESTSH: <RSCFDn_base> + $023A_H$
RSCFDnCFDFESTSLL: <RSCFDn_base> + 0238_H, RSCFDnCFDFESTSLH: <RSCFDn_base> + 0239_H,
RSCFDnCFDFESTSHL: <RSCFDn_base> + $023A_H$, RSCFDnCFDFESTSHH: <RSCFDn_base> + $023B_H$

Value after reset: 03FF $FFFF_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CF5 EMP	CF4 EMP	CF3 EMP	CF2 EMP	CF1 EMP	CF0 EMP	RF7 EMP	RF6 EMP	RF5 EMP	RF4 EMP	RF3 EMP	RF2 EMP	RF1 EMP	RF0 EMP
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.136 RSCFDnCFDFESTS Register Contents

Bit	Bit Name	Function
31 to 14	Reserved	These bits are read as the value after reset.
13	CF5EMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: Transmit/receive FIFO buffer k contains a message. 1: Transmit/receive FIFO buffer k contains no message. (k = 0 to 5)
12	CF4EMP	
11	CF3EMP	
10	CF2EMP	
9	CF1EMP	
8	CF0EMP	
7	RF7EMP	Receive FIFO Buffer Empty Status Flag 0: Receive FIFO buffer x contains an unread message. 1: Receive FIFO buffer x contains no unread message (buffer empty). (x = 0 to 7)
6	RF6EMP	
5	RF5EMP	
4	RF4EMP	
3	RF3EMP	
2	RF2EMP	
1	RF1EMP	
0	RF0EMP	

This RSCFDnCFDFESTS register is set to 03FF $FFFF_H$ in global reset mode.

CFkEMP Flag (k = 0 to 5)

The CFkEMP flag is set to 1 when the CFEMP flag in the RSCFDnCFDCFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)). When the CFEMP flag is cleared to 0 (the transmit/receive FIFO buffer contains messages), the CFkEMP flag is cleared to 0.

RFxEMP Flag (x = 0 to 7)

The RFxEMP flag is set to 1 when the RFEMP flag in the RSCFDnCFDRFSTSx register is set to 1 (the receive FIFO buffer contains no unread message). When the RFEMP flag is cleared to 0 (the receive FIFO buffer contains unread messages), the RFxEMP flag is cleared to 0.

25.4.9.2 RSCFDnCFDFFSTS — FIFO Full Status Register

Access: RSCFDnCFDFFSTS register can be read only in 32-bit units
RSCFDnCFDFFSTSL, RSCFDnCFDFFSTSH registers can be read only in 16-bit units
RSCFDnCFDFFSTSLL, RSCFDnCFDFFSTSLH, RSCFDnCFDFFSTSHL, RSCFDnCFDFFSTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDFFSTS: <RSCFDn_base> + $023C_H$
RSCFDnCFDFFSTSL: <RSCFDn_base> + $023C_H$, RSCFDnCFDFFSTSH: <RSCFDn_base> + $023E_H$
RSCFDnCFDFFSTSLL: <RSCFDn_base> + $023C_H$, RSCFDnCFDFFSTSLH: <RSCFDn_base> + $023D_H$,
RSCFDnCFDFFSTSHL: <RSCFDn_base> + $023E_H$, RSCFDnCFDFFSTSHH: <RSCFDn_base> + $023F_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CF5 FLL	CF4 FLL	CF3 FLL	CF2 FLL	CF1 FLL	CF0 FLL	RF7 FLL	RF6 FLL	RF5 FLL	RF4 FLL	RF3 FLL	RF2 FLL	RF1 FLL	RF0 FLL
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.137 RSCFDnCFDFFSTS Register Contents

Bit	Bit Name	Function
31 to 14	Reserved	These bits are read as the value after reset.
13	CF5FLL	Transmit/Receive FIFO Buffer Full Status Flag 0: Transmit/receive buffer k is not full. 1: Transmit/receive buffer k is full. (k = 0 to 5)
12	CF4FLL	
11	CF3FLL	
10	CF2FLL	
9	CF1FLL	
8	CF0FLL	
7	RF7FLL	Receive FIFO Buffer Full Status Flag 0: Receive FIFO buffer x is not full. 1: Receive FIFO buffer x is full. (x = 0 to 7)
6	RF6FLL	
5	RF5FLL	
4	RF4FLL	
3	RF3FLL	
2	RF2FLL	
1	RF1FLL	
0	RF0FLL	

The RSCFDnCFDFFSTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkFLL Flag (k = 0 to 5)

The CFkFLL flag is set to 1 when the CFFLL flag in the RSCFDnCFDCFSTSk register is set to 1 (the transmit/receive FIFO buffer is full).
When the CFFLL flag is cleared to 0 (the transmit/receive FIFO buffer is not full), the CFkFLL flag is cleared to 0.

RFxFLL Flag (x = 0 to 7)

The RFxFLL flag is set to 1 when the RFFLL flag in the RSCFDnCFDRFSTSx register is set to 1 (the receive FIFO buffer is full). When the RFFLL flag is cleared to 0 (the receive FIFO buffer is not full), the RFxFLL flag is cleared to 0.

25.4.9.3 RSCFDnCFDFMSTS — FIFO Message Lost Status Register

Access: RSCFDnCFDFMSTS register can be read only in 32-bit units
RSCFDnCFDFMSTSL, RSCFDnCFDFMSTSH registers can be read only in 16-bit units
RSCFDnCFDFMSTSLL, RSCFDnCFDFMSTSLH, RSCFDnCFDFMSTSHL, RSCFDnCFDFMSTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDFMSTS: <RSCFDn_base> + 0240_H
RSCFDnCFDFMSTSL: <RSCFDn_base> + 0240_H, RSCFDnCFDFMSTSH: <RSCFDn_base> + 0242_H
RSCFDnCFDFMSTSLL: <RSCFDn_base> + 0240_H, RSCFDnCFDFMSTSLH: <RSCFDn_base> + 0241_H,
RSCFDnCFDFMSTSHL: <RSCFDn_base> + 0242_H, RSCFDnCFDFMSTSHH: <RSCFDn_base> + 0243_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CF5 MLT	CF4 MLT	CF3 MLT	CF2 MLT	CF1 MLT	CF0 MLT	RF7 MLT	RF6 MLT	RF5 MLT	RF4 MLT	RF3 MLT	RF2 MLT	RF1 MLT	RF0 MLT
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.138 RSCFDnCFDFMSTS Register Contents

Bit	Bit Name	Function
31 to 14	Reserved	These bits are read as the value after reset.
13	CF5MLT	Transmit/Receive FIFO Buffer Message Lost Status Flag 0: No transmit/receive FIFO buffer k message is lost. 1: A transmit/receive FIFO buffer k message is lost. (k = 0 to 5)
12	CF4MLT	
11	CF3MLT	
10	CF2MLT	
9	CF1MLT	
8	CF0MLT	
7	RF7MLT	Receive FIFO Buffer Message Lost Status Flag 0: No receive FIFO buffer x message is lost. 1: A receive FIFO buffer x message is lost. (x = 0 to 7)
6	RF6MLT	
5	RF5MLT	
4	RF4MLT	
3	RF3MLT	
2	RF2MLT	
1	RF1MLT	
0	RF0MLT	

The RSCFDnCFDFMSTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkMLT Flag (k = 0 to 5)

The CFkMLT flag is set to 1 when the CFMLT flag in the RSCFDnCFDCFSTSk register is set to 1 (a transmit/receive FIFO message is lost). When the CFMLT flag is cleared to 0, the CFkMLT flag is cleared to 0.

RFxMLT Flag (x = 0 to 7)

The RFxMLT flag is set to 1 when the RFMLT flag in the RSCFDnCFDRFSTSx register is set to 1 (a receive FIFO message is lost). When the RFMLT flag is cleared to 0, the RFxMLT flag is cleared to 0.

25.4.9.4 RSCFDnCFDRFISTS — Receive FIFO Buffer Interrupt Flag Status Register

Access: RSCFDnCFDRFISTS register can be read only in 32-bit units
RSCFDnCFDRFISTSL, RSCFDnCFDRFISTSH registers can be read only in 16-bit units
RSCFDnCFDRFISTSLL, RSCFDnCFDRFISTSLH, RSCFDnCFDRFISTSHL, RSCFDnCFDRFISTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDRFISTS: <RSCFDn_base> + 0244_H
RSCFDnCFDRFISTSL: <RSCFDn_base> + 0244_H, RSCFDnCFDRFISTSH: <RSCFDn_base> + 0246_H
RSCFDnCFDRFISTSLL: <RSCFDn_base> + 0244_H, RSCFDnCFDRFISTSLH: <RSCFDn_base> + 0245_H,
RSCFDnCFDRFISTSHL: <RSCFDn_base> + 0246_H, RSCFDnCFDRFISTSHH: <RSCFDn_base> + 0247_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RF7IF	RF6IF	RF5IF	RF4IF	RF3IF	RF2IF	RF1IF	RF0IF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.139 RSCFDnCFDRFISTS Register Contents

Bit	Bit Name	Function
31 to 8	Reserved	These bits are read as the value after reset.
7	RF7IF	Receive FIFO Buffer Interrupt Request Status Flag 0: No receive FIFO buffer x interrupt request is present. 1: A receive FIFO buffer x interrupt request is present. (x = 0 to 7)
6	RF6IF	
5	RF5IF	
4	RF4IF	
3	RF3IF	
2	RF2IF	
1	RF1IF	
0	RF0IF	

The RSCFDnCFDRFISTS register is cleared to $0000\ 0000_H$ in global reset mode.

RFxIF Flag (x = 0 to 7)

The RFxIF flag is set to 1 when the RFIF flag in the RSCFDnCFDRFSTSx register is set to 1 (a receive FIFO interrupt request is present). When the RFIF flag is cleared to 0, the RFxIF flag is cleared to 0.

25.4.9.5 RSCFDnCFDCFRISTS — Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register

Access: RSCFDnCFDCFRISTS register can be read only in 32-bit units
RSCFDnCFDCFRISTSL, RSCFDnCFDCFRISTSH registers can be read only in 16-bit units
RSCFDnCFDCFRISTSLL, RSCFDnCFDCFRISTSLH, RSCFDnCFDCFRISTSHL, RSCFDnCFDCFRISTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDCFRISTS: <RSCFDn_base> + 0248_H
RSCFDnCFDCFRISTSL: <RSCFDn_base> + 0248_H, RSCFDnCFDCFRISTSH: <RSCFDn_base> + $024A_H$
RSCFDnCFDCFRISTSLL: <RSCFDn_base> + 0248_H, RSCFDnCFDCFRISTSLH: <RSCFDn_base> + 0249_H,
RSCFDnCFDCFRISTSHL: <RSCFDn_base> + $024A_H$, RSCFDnCFDCFRISTSHH: <RSCFDn_base> + $024B_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CF5 RXIF	CF4 RXIF	CF3 RXIF	CF2 RXIF	CF1 RXIF	CF0 RXIF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.140 RSCFDnCFDCFRISTS Register Contents

Bit	Bit Name	Function
31 to 6	Reserved	These bits are read as the value after reset.
5	CF5RXIF	Transmit/Receive FIFO Buffer Receive Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k receive interrupt request is present. 1: A transmit/receive FIFO buffer k receive interrupt request is present. (k = 0 to 5)
4	CF4RXIF	
3	CF3RXIF	
2	CF2RXIF	
1	CF1RXIF	
0	CF0RXIF	

The RSCFDnCFDCFRISTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkRXIF Flag (k = 0 to 5)

The CFkRXIF flag is set to 1 when the CFRXIF flag in the RSCFDnCFDCFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). When the CFRXIF flag is cleared to 0, the CFkRXIF flag is cleared to 0.

25.4.9.6 RSCFDnCFDCFTISTS — Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register

Access: RSCFDnCFDCFTISTS register can be read only in 32-bit units
RSCFDnCFDCFTISTSL, RSCFDnCFDCFTISTSH registers can be read only in 16-bit units
RSCFDnCFDCFTISTSLL, RSCFDnCFDCFTISTSLH, RSCFDnCFDCFTISTSHL, RSCFDnCFDCFTISTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDCFTISTS: <RSCFDn_base> + $024C_H$
RSCFDnCFDCFTISTSL: <RSCFDn_base> + $024C_H$, RSCFDnCFDCFTISTSH: <RSCFDn_base> + $024E_H$
RSCFDnCFDCFTISTSLL: <RSCFDn_base> + $024C_H$, RSCFDnCFDCFTISTSLH: <RSCFDn_base> + $024D_H$,
RSCFDnCFDCFTISTSHL: <RSCFDn_base> + $024E_H$, RSCFDnCFDCFTISTSHH: <RSCFDn_base> + $024F_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CF5 TXIF	CF4 TXIF	CF3 TXIF	CF2 TXIF	CF1 TXIF	CF0 TXIF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.141 RSCFDnCFDCFTISTS Register Contents

Bit	Bit Name	Function
31 to 6	Reserved	These bits are read as the value after reset.
5	CF5TXIF	Transmit/Receive FIFO Buffer Transmit Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k transmit interrupt request is present. 1: A transmit/receive FIFO buffer k transmit interrupt request is present. (k = 0 to 5)
4	CF4TXIF	
3	CF3TXIF	
2	CF2TXIF	
1	CF1TXIF	
0	CF0TXIF	

The RSCFDnCFDCFTISTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkTXIF Flag (k = 0 to 5)

The CFkTXIF flag is set to 1 when the CFTXIF flag in the RSCFDnCFDCFSTSk register is set to 1 (a transmit/receive FIFO transmit interrupt request is present). When the CFTXIF flag is cleared to 0, the CFkTXIF flag is cleared to 0.

25.4.10 Details of FIFO DMA-related Registers

25.4.10.1 RSCFDnCFDCDTCT — DMA Enable Register

Access: RSCFDnCFDCDTCT register can be read/written in 32-bit units
RSCFDnCFDCDTCTL, RSCFDnCFDCDTCTH registers can be read/written in 16-bit units
RSCFDnCFDCDTCTLL, RSCFDnCFDCDTCTLH, RSCFDnCFDCDTCTHL, RSCFDnCFDCDTCTHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCDTCT: <RSCFDn_base> + 0490_H
RSCFDnCFDCDTCTL: <RSCFDn_base> + 0490_H
RSCFDnCFDCDTCTH: <RSCFDn_base> + 0492_H
RSCFDnCFDCDTCTLL: <RSCFDn_base> + 0490_H,
RSCFDnCFDCDTCTLH: <RSCFDn_base> + 0491_H,
RSCFDnCFDCDTCTHL: <RSCFDn_base> + 0492_H,
RSCFDnCFDCDTCTHH: <RSCFDn_base> + 0493_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CF DMAE1	CF DMAE0	RF DMAE7	RF DMAE6	RF DMAE5	RF DMAE4	RF DMAE3	RF DMAE2	RF DMAE1	RF DMAE0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.142 RSCFDnCFDCDTCT Register Contents

Bit	Bit Name	Function
31 to 10	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
9	CFDMAE1	Transmit/Receive FIFO Buffer 3 DMA Enable 0: A DMA transfer request of transmit/receive FIFO buffer 3 is disabled. 1: A DMA transfer request of transmit/receive FIFO buffer 3 is enabled.
8	CFDMAE0	Transmit/Receive FIFO Buffer 0 DMA Enable 0: A DMA transfer request of transmit/receive FIFO buffer 0 is disabled. 1: A DMA transfer request of transmit/receive FIFO buffer 0 is enabled.
7	RFDMAE7	Receive FIFO Buffer x DMA Enable 0: A DMA transfer request of receive FIFO buffer x is disabled. 1: A DMA transfer request of receive FIFO buffer x is enabled. (x = 0 to 7)
6	RFDMAE6	
5	RFDMAE5	
4	RFDMAE4	
3	RFDMAE3	
2	RFDMAE2	
1	RFDMAE1	
0	RFDMAE0	

Modify the RSCFDnCFDCDTCT register in global operating mode or global test mode.

CFDMAEm Bit

This bit is used to enable DMA transfer for transmit/receive FIFO buffer 3 × m (the first transmit/receive FIFO buffer allocated to channel m). DMA transfer is enabled only for transmit/receive FIFO buffers for which the CFM[1:0] bits in the RSCFDnCFDCFCCk register is set to 00_B (receive mode). Set this bit to 0 when the CFM[1:0] value is 01_B (transmit mode) or 10_B (gateway mode).

RFDMAEx Bit

This bit is used to enable DMA transfer for receive FIFO buffer x.

25.4.10.2 RSCFDnCFDCDTSTS — DMA Status Register

Access: RSCFDnCFDCDTSTS register can be read only in 32-bit units
RSCFDnCFDCDTSTSL, RSCFDnCFDCDTSTSH registers can be read only in 16-bit units
RSCFDnCFDCDTSTSLL, RSCFDnCFDCDTSTSLH, RSCFDnCFDCDTSTSHL, RSCFDnCFDCDTSTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDCDTSTS: <RSCFDn_base> + 0494_H
RSCFDnCFDCDTSTSL: <RSCFDn_base> + 0494_H
RSCFDnCFDCDTSTSH: <RSCFDn_base> + 0496_H
RSCFDnCFDCDTSTSLL: <RSCFDn_base> + 0494_H,
RSCFDnCFDCDTSTSLH: <RSCFDn_base> + 0495_H,
RSCFDnCFDCDTSTSHL: <RSCFDn_base> + 0496_H,
RSCFDnCFDCDTSTSHH: <RSCFDn_base> + 0497_H

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CFDMASTS1	CFDMASTS0	RFDMASTS7	RFDMASTS6	RFDMASTS5	RFDMASTS4	RFDMASTS3	RFDMASTS2	RFDMASTS1	RFDMASTS0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.143 RSCFDnCFDCDTSTS Register Contents

Bit	Bit Name	Function
31 to 10	Reserved	These bits are read as the value after reset.
9	CFDMASTS1	Transmit/Receive FIFO Buffer 3 DMA Status 0: DMA transfer of transmit/receive FIFO buffer 3 is not in progress. 1: DMA transfer of transmit/receive FIFO buffer 3 is in progress.
8	CFDMASTS0	Transmit/Receive FIFO Buffer 0 DMA Status 0: DMA transfer of transmit/receive FIFO buffer 0 is not in progress. 1: DMA transfer of transmit/receive FIFO buffer 0 is in progress.
7	RFDMASTS7	Receive FIFO Buffer x DMA Status 0: DMA transfer of receive FIFO buffer x is not in progress. 1: DMA transfer of receive FIFO buffer x is in progress. (x = 0 to 7)
6	RFDMASTS6	
5	RFDMASTS5	
4	RFDMASTS4	
3	RFDMASTS3	
2	RFDMASTS2	
1	RFDMASTS1	
0	RFDMASTS0	

CFDMASTSm Bit

When DMA transfer is enabled (CFDMAEm bit in the RSCFDnCFDCDTCT register is 1) for the transmit/receive FIFO buffer 3 × m (the first transmit/receive FIFO buffer allocated to channel m) while the transmit/receive FIFO buffer contains one of more messages, the CFDMASTSm bit is set to 1 indicating that DMA transfer is in progress.
When all messages in the transmit/receive FIFO buffer have been transferred or DMA transfer is disabled (CFDMAEm bit is 0), the CFDMASTSm bit is cleared to 0 indicating that DMA transfer has been completed. If the CFDMAEm bit is set to 0 during DMA transfer, the CFDMASTSm bit is cleared to 0 after the ongoing DMA transfer has been completed (when the message that is being transferred has been transferred to the last byte in the payload storage area).
These bits are cleared to 0 in global reset mode.

RFDMASTSx Bit

When DMA transfer is enabled (corresponding RFDMAEx bit in the RSCFDnCFDCDTCT register is 1) for the receive FIFO buffer x and the receive FIFO buffer contains one of more messages, the RFDMAEx bit is set to 1 indicating that DMA transfer is in progress.

When all messages in the receive FIFO buffer x have been transferred or DMA transfer is disabled (RFDMAEx bit = 0), the RFDMASTSx bit is cleared to 0 indicating that DMA transfer has been completed. If the RFDMAEx bit is set to 0 during DMA transfer, the RFDMASTSx bit is cleared to 0 after the ongoing DMA transfer has been completed (when the message that is being transferred has been transferred to the last byte in the payload storage area)

These bits are cleared to 0 in global reset mode.

25.4.11 Details of Transmit Buffer-related Registers

25.4.11.1 RSCFDnCFDTMCp — Transmit Buffer Control Register (p = 0 to 31)

Access: RSCFDnCFDTMCp registers can be read/written in 8-bit units

Address: RSCFDnCFDTMCp: <RSCFDn_base> + 0250_H + (01_H × p)

Value after reset: 00H

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	TMOM	TMTAR	TMTR
Value after reset:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W*1	R/W*1

Note 1. The only effective value for writing to this bit is 1, which sets the bit. Otherwise writing to the bit results in retention of its state.

Table 25.144 RSCFDnCFDTMCp Register Contents

Bit	Bit Name	Function
7 to 3	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
2	TMOM	One-Shot Transmission Enable 0: One-shot transmission is disabled. 1: One-shot transmission is enabled.
1	TMTAR	Transmit Abort Request 0: Transmit abort is not requested. 1: Transmit abort is requested.
0	TMTR	Transmit Request 0: Transmission is not requested. 1: Transmission is requested.

When the RSCFDnCFDTMCp register meets any of the following conditions, set it to 00_H.

- The RSCFDnCFDTMCp register corresponds to the transmit buffer number selected by the CFTML[3:0] bits in the RSCFDnCFDCFCCk register (p = m × 16 + the value of CFTML[3:0] bits).
- The RSCFDnCFDTMCp register corresponds to the transmit buffer allocated to the transmit queue by the TXQDC[3:0] bits in the RSCFDnCFDTXQCCm (m = 0, 1) register (p = (m × 16 + 15) to (m × 16 + 15 . the value of TXQDC[3:0] bits)).
- RSCFDnCFDTMCp register (p = (m × 16) + 1, (m × 16) + 2, (m × 16) + 4, or (m × 16) + 5) corresponding to the transmit buffer allocated as a payload storage area when the TMME bit in the RSCFDnCFDCmFDCFG register is 1 (transmit buffer merge mode)

Bits in the RSCFDnCFDTMCp register are all cleared to 0 in channel reset mode. Modify the RSCFDnCFDTMCp register in channel communication mode or channel halt mode.

TMOM Bit

Setting this bit to 1 enables one-shot transmission. When transmission fails, retransmission defined in the CAN protocol is not performed.

Modify the TMOM bit when the TMTRM flag in the RSCFDnCFDTMSTSp register is set to 0. Set the TMOM bit to 1 together with the TMTR bit.

TMTAR Bit

Setting this bit to 1 generates a transmit abort request for the message stored in the transmit buffer. However, a message that is being transmitted or one that will be transmitted next cannot be aborted.
The TMTAR bit can be set to 1 when TMTR bit is 1.
The TMTAR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed.
- An error or arbitration loss has been detected.

If this bit becomes 0 at the same time as the program writes 1 to this bit, this bit becomes 0.

TMTR Bit

Setting this bit to 1 transmits the message stored in the transmit buffer.
The TMTR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed after the TMTAR bit was set to 1.
- An error or arbitration-lost has been detected with the TMOM bit set to 1.

Set the TMTR bit to 1 when the value of TMTRF[1:0] in the RSCFDnCFDTMSTSp register is 00_B.

25.4.11.2 RSCFDnCFDTMSTSp — Transmit Buffer Status Register (p = 0 to 31)

Access: RSCFDnCFDTMSTSp registers can be read/written in 8-bit units

Address: RSCFDnCFDTMSTSp: <RSCFDn_base> + $02D0_H$ + (01_H × p)

Value after reset: 00_H

Bit:	7	6	5	4	3	2	1	0
	—	—	—	TMTARM	TMTRM	TMTRF[1:0]		TMTSTS
Value after reset:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R

Table 25.145 RSCFDnCFDTMSTSp Register Contents

Bit	Bit Name	Function
7 to 5	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
4	TMTARM	Transmit Buffer Transmit Abort Request Status Flag 0: No transmit abort request is present. 1: A transmit abort request is present.
3	TMTRM	Transmit Buffer Transmit Request Status Flag 0: No transmit request is present. 1: A transmit request is present.
2, 1	TMTRF[1:0]	Transmit Buffer Transmit Result Status Flag b2 b1 0 0: Transmission is in progress or no transmit request is present. 0 1: Transmit abort has been completed. 1 0: Transmission has been completed (without transmit abort request). 1 1: Transmission has been completed (with transmit abort request).
0	TMTSTS	Transmit Buffer Transmit Status Flag 0: Transmission is not in progress. 1: Transmission is in progress.

The RSCFDnCFDTMSTSp register is cleared to all 0 in channel reset mode.

TMTARM Flag

The TMTARM flag is set to 1 when the TMTAR bit in the RSCFDnCFDTMCp register is set to 1.
The TMTARM flag is set to 0 when the TMTAR bit in the RSCFDnCFDTMCp register is set to 0.

TMTRM Flag

The TMTRM flag is set to 1 when the TMTR bit in the RSCFDnCFDTMCp register is set to 1.
The TMTRM flag is set to 0 when the TMTR bit in the RSCFDnCFDTMCp register is set to 0.

TMTRF[1:0] Flag

This flag indicates the result of transmission from the transmit buffer.
00_B: Transmission is in progress or no transmit request is present.
01_B: Transmission from the transmit buffer was aborted.
10_B: Transmission has been completed with the TMTAR bit in the RSCFDnCFDTMCp register set to 0 (transmit abort is not requested).
11_B: Transmission has been completed with the TMTAR bit in the RSCFDnCFDTMCp register set to 1 (transmit abort is requested).
Write 00_B to the TMTRF[1:0] flag in channel communication mode or channel halt mode. Do not write any value other than 00_B to this flag.

TMTSTS Flag

This flag is set to 1 when transmission from the transmit buffer starts, and is cleared to 0 when transmission from the transmit buffer has been completed or terminated due to a bus error or arbitration lost.

25.4.11.3 RSCFDnCFDTMIDp — Transmit Buffer ID Register (p = 0 to 31)

Access: RSCFDnCFDTMIDp register can be read/written in 32-bit units
RSCFDnCFDTMIDpL, RSCFDnCFDTMIDpH registers can be read/written in 16-bit units
RSCFDnCFDTMIDpLL, RSCFDnCFDTMIDpLH, RSCFDnCFDTMIDpHL, RSCFDnCFDTMIDpHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTMIDp: <RSCFDn_base> + 4000_H + (20_H × p)
RSCFDnCFDTMIDpL: <RSCFDn_base> + 4000_H + (20_H × p),
RSCFDnCFDTMIDpH: <RSCFDn_base> + 4002_H + (20_H × p)
RSCFDnCFDTMIDpLL: <RSCFDn_base> + 4000_H + (20_H × p),
RSCFDnCFDTMIDpLH: <RSCFDn_base> + 4001_H + (20_H × p),
RSCFDnCFDTMIDpHL: <RSCFDn_base> + 4002_H + (20_H × p),
RSCFDnCFDTMIDpHH: <RSCFDn_base> + 4003_H + (20_H × p)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIDE	TMRTR	THLEN	TMID[28:16]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMID[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.146 RSCFDnCFDTMIDp Register Contents

Bit	Bit Name	Function
31	TMIDE	Transmit Buffer IDE 0: Standard ID 1: Extended ID
30	TMRTR	Transmit Buffer RTR/RRS • When the transmit message is a classical CAN frame 0: Data frame 1: Remote frame • When the transmit message is a CANFD frame Write 0 to this bit.
29	THLEN	Transmit History Data Store Enable 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	TMID[28:0]	Transmit Buffer ID Data Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11.

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write data to this register. If this register is allocated to the transmit queue, only write data to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMIDE Bit

This bit is used to set the ID format of the message to be transmitted from the transmit buffer.

TMRTR Bit

This bit is used to set the data format of the message to be transmitted from the transmit buffer.
Set this bit to 0 when the TMFDF bit in the RSCFDnCFDTMFDCTRp register is 1 (CANFD frame).

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, buffer type, and timestamp) of transmit messages is stored in the transmit history buffer after transmission is completed.

TMID[28:0] Bits

These bits are used to set the ID of the message to be transmitted from the transmit buffer.

25.4.11.4 RSCFDnCFDTMPTRp — Transmit Buffer Pointer Register (p = 0 to 31)

Access: RSCFDnCFDTMPTRp register can be read/written in 32-bit units
RSCFDnCFDTMPTRpL, RSCFDnCFDTMPTRpH registers can be read/written in 16-bit units
RSCFDnCFDTMPTRpLL, RSCFDnCFDTMPTRpLH, RSCFDnCFDTMPTRpHL, RSCFDnCFDTMPTRpHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTMPTRp: <RSCFDn_base> + 4004_H + (20_H × p)
RSCFDnCFDTMPTRpL: <RSCFDn_base> + 4004_H + (20_H × p),
RSCFDnCFDTMPTRpH: <RSCFDn_base> + 4006_H + (20_H × p)
RSCFDnCFDTMPTRpLL: <RSCFDn_base> + 4004_H + (20_H × p),
RSCFDnCFDTMPTRpLH: <RSCFDn_base> + 4005_H + (20_H × p),
RSCFDnCFDTMPTRpHL: <RSCFDn_base> + 4006_H + (20_H × p),
RSCFDnCFDTMPTRpHH: <RSCFDn_base> + 4007_H + (20_H × p)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDLC[3:0]				—	—	—	—	TMPTR[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.147 RSCFDnCFDTMPTRp Register Contents

Bit	Bit Name	Function
31 to 28	TMDLC[3:0]	Transmit Buffer DLC Data

b31 b30 b29 b28	Classical CAN Frame	CANFD Frame
0 0 0 0	0 data bytes	
0 0 0 1	1 data byte	
0 0 1 0	2 data bytes	
0 0 1 1	3 data bytes	
0 1 0 0	4 data bytes	
0 1 0 1	5 data bytes	
0 1 1 0	6 data bytes	
0 1 1 1	7 data bytes	
1 0 0 0	8 data bytes	
1 0 0 1	8 data bytes	12 data bytes
1 0 1 0		16 data bytes
1 0 1 1		20 data bytes
1 1 0 0		24 data bytes
1 1 0 1		32 data bytes
1 1 1 0		48 data bytes
1 1 1 1		64 data bytes

27 to 24	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
23 to 16	TMPTR[7:0]	Transmit Buffer Label Data Set the label information to be stored in the transmit history buffer.
15 to 0	Reserved	These bits are read as the value after reset. The write value should be the value after reset.

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMDLC[3:0] Bits

These bits are used to set the data length of the message to be transmitted from the transmit buffer when the TMRTR bit in the RSCFDnCFDTMIDp register is set to 0 (data frame).
When the TMDLC[3:0] bits are set to 1001_B or more while the TMFDF bit in the RSCFDnCFDTMFDCTRp register is 0 (classical CAN frame), 8-byte data is transmitted actually. When the MFDF bit is 1 (CANFD frame), the settable value range varies depending on the setting of the TMME bit in the RSCFDnCFDCmFDCFG register.

- When the TMME bit = 0 (transmit buffer merge mode disabled):
  A value of 0000_B to 1111_B is settable. If a value larger than 1100_B is set, payloads exceeding 20 bytes are padded by CCH.
- When the TMME bit = 1 (transmit buffer merge mode enabled):
  When the corresponding transmit buffer number p = (m × 16) + 0 or (m × 16) + 3, a value of 0000_B to 1111_B is settable. In other cases, set a value of 0000_B to 1011_B (20 data bytes).

When the TMRTR bit is 1 (remote frame), set the data length of a message to be requested.

TMPTR[7:0] Bits

When message transmission has been completed, the TMPTR[7:0] value is stored in the transmit history buffer.

25.4.11.5 RSCFDnCFDTMFDCTRp — Transmit Buffer CANFD Configuration Register (p = 0 to 31)

Access: RSCFDnCFDTMFDCTRp register can be read/written in 32-bit units
RSCFDnCFDTMFDCTRpL, RSCFDnCFDTMFDCTRpH registers can be read/written in 16-bit units
RSCFDnCFDTMFDCTRpLL, RSCFDnCFDTMFDCTRpLH, RSCFDnCFDTMFDCTRpHL, RSCFDnCFDTMFDCTRpHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTMFDCTRp: <RSCFDn_base> + 4008_H + (20_H × p)
RSCFDnCFDTMFDCTRpL: <RSCFDn_base> + 4008_H + (20_H × p),
RSCFDnCFDTMFDCTRpH: <RSCFDn_base> + $400A_H$ + (20_H × p)
RSCFDnCFDTMFDCTRpLL: <RSCFDn_base> + 4008_H + (20_H × p),
RSCFDnCFDTMFDCTRpLH: <RSCFDn_base> + 4009_H + (20_H × p),
RSCFDnCFDTMFDCTRpHL: <RSCFDn_base> + $400A_H$ + (20_H × p),
RSCFDnCFDTMFDCTRpHH: <RSCFDn_base> + $400B_H$ + (20_H × p)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TMFDF	TMBRS	TMESI
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 25.148 RSCFDnCFDTMFDCTRp Register Contents

Bit	Bit Name	Function
31 to 3	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
2	TMFDF	FDF 0: Classical CAN frame 1: CANFD frame
1	TMBRS	BRS 0: The bit rate in the data area does not change. 1: The bit rate in the data area changes.
0	TMESI	ESI 0: Error active node 1: Error passive node

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is 0 (transmission not requested). When this register is linked to the transmit/receive FIFO buffer, do not write data to this register. When this register is allocated to the transmit queue, write data only to transmit buffer p (p = m × 16 + 15) of the corresponding channel.

TMFDF Bit

This bit is used to set the FD format of the message to be transmitted from the transmit buffer.

TMBRS Bit

When this bit is set to 1 while the TMFDF bit is 1, the data area of a transmit message is transmitted at the data bit rate. When the TMFDF bit is 0, write 0 to this bit.

TMESI Bit

This bit is used to set the ESI bit value of the message to be transmitted from the transmit buffer when the TMFDF bit is 1. The set value is transmitted when the ESIC bit in the RSCFDnCFDCmFDCFG register is 1 and the channel is in the error active state. When the channel is in the error passive state, the ESI bit value that shows an error passive node is transmitted regardless of this bit value. When the TMFDF bit is 0, write 0 to this bit.

25.4.11.6 RSCFDnCFDTMDFb_p — Transmit Buffer Data Field b Register (b = 0 to 4, p = 0 to 31)

Access: RSCFDnCFDTMDFb_p register can be read/written in 32-bit units
RSCFDnCFDTMDFb_pL, RSCFDnCFDTMDFb_pH registers can be read/written in 16-bit units
RSCFDnCFDTMDFb_pLL, RSCFDnCFDTMDFb_pLH, RSCFDnCFDTMDFb_pHL, RSCFDnCFDTMDFb_pHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTMDFb_p: <RSCFDn_base> + $400C_H$ + (04_H × b) + (20_H × p)
RSCFDnCFDTMDFb_pL: <RSCFDn_base> + $400C_H$ + (04_H × b) + (20_H × p),
RSCFDnCFDTMDFb_pH: <RSCFDn_base> + $400E_H$ + (04_H × b) + (20_H × p)
RSCFDnCFDTMDFb_pLL: <RSCFDn_base> + $400C_H$ + (04_H × b) + (20_H × p),
RSCFDnCFDTMDFb_pLH: <RSCFDn_base> + $400D_H$ + (04_H × b) + (20_H × p),
RSCFDnCFDTMDFb_pHL: <RSCFDn_base> + $400E_H$ + (04_H × b) + (20_H × p),
RSCFDnCFDTMDFb_pHH: <RSCFDn_base> + $400F_H$ + (04_H × b) + (20_H × p)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB4 × b + 3 [7:0]								TMDB4 × b + 2 [7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB4 × b + 1 [7:0]								TMDB4 × b + 0 [7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.149 RSCFDnCFDTMDFb_p Register Contents

Bit	Bit Name	Function
31 to 24	TMDB4 × b + 3 [7:0]	Transmit Buffer Data Byte 4 × b + 3
23 to 16	TMDB4 × b + 2 [7:0]	Transmit Buffer Data Byte 4 × b + 2
15 to 8	TMDB4 × b + 1 [7:0]	Transmit Buffer Data Byte 4 × b + 1
7 to 0	TMDB4 × b + 0 [7:0]	Transmit Buffer Data Byte 4 × b + 0 Set the transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

25.4.11.7 RSCFDnCFDTMIECy — Transmit Buffer Interrupt Enable Configuration Register (y = 0)

Access: RSCFDnCFDTMIECy register can be read/written in 32-bit units
RSCFDnCFDTMIECyL, RSCFDnCFDTMIECyH registers can be read/written in 16-bit units
RSCFDnCFDTMIECyLL, RSCFDnCFDTMIECyLH, RSCFDnCFDTMIECyHL, RSCFDnCFDTMIECyHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTMIECy: <RSCFDn_base> + 0390_H + (04_H × y)
RSCFDnCFDTMIECyL: <RSCFDn_base> + 0390_H + (04_H × y),
RSCFDnCFDTMIECyH: <RSCFDn_base> + 0392_H + (04_H × y)
RSCFDnCFDTMIECyLL: <RSCFDn_base> + 0390_H + (04_H × y),
RSCFDnCFDTMIECyLH: <RSCFDn_base> + 0391_H + (04_H × y),
RSCFDnCFDTMIECyHL: <RSCFDn_base> + 0392_H + (04_H × y),
RSCFDnCFDTMIECyHH: <RSCFDn_base> + 0393_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIEp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMIEp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.150 RSCFDnCFDTMIECy Register Contents

Bit	Bit Name	Function
31 to 16	TMIEp	Transmit Buffer Interrupt Enable p (p = y × 32 + 31 to y × 32 + 16) 0: Transmit buffer interrupt is disabled 1: Transmit buffer interrupt is enabled
15 to 0	TMIEp	Transmit Buffer Interrupt Enable p (p = y × 32 + 15 to y × 32 + 0) 0: Transmit buffer interrupt is disabled. 1: Transmit buffer interrupt is enabled.

TMIEp Bits (p = 0 to 31)

When any of these bits is set to 1 and the corresponding transmission has been completed, a transmit buffer interrupt request is generated.

Modify these bits when the TMTRM flag in the corresponding RSCFDnCFDTMSTSp register is 0 (no transmit request is present).

Write 0 to bits corresponding to transmit buffers linked to transmit/receive FIFO buffers or transmit buffers allocated to the transmit queue.

When the TMME bit in the RSCFDnCFDCmFDCFG register is 1 (transmit buffer merge mode enable), set the bit corresponding to the transmit buffer allocated as a payload storage area to 0.

Table 25.151 shows the bit assignment.

Table 25.151 TMIEp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

25.4.12 Details of Transmit Buffer Status-related Registers

25.4.12.1 RSCFDnCFDTMTRSTSy — Transmit Buffer Transmit Request Status Register (y = 0)

Access: RSCFDnCFDTMTRSTSy register can be read only in 32-bit units
RSCFDnCFDTMTRSTSyL, RSCFDnCFDTMTRSTSyH registers can be read only in 16-bit units
RSCFDnCFDTMTRSTSyLL, RSCFDnCFDTMTRSTSyLH, RSCFDnCFDTMTRSTSyHL, RSCFDnCFDTMTRSTSyHH registers can be read only in 8-bit units

Address: RSCFDnCFDTMTRSTSy: <RSCFDn_base> + 0350_H + (04_H × y)
RSCFDnCFDTMTRSTSyL: <RSCFDn_base> + 0350_H + (04_H × y),
RSCFDnCFDTMTRSTSyH: <RSCFDn_base> + 0352_H + (04_H × y)
RSCFDnCFDTMTRSTSyLL: <RSCFDn_base> + 0350_H + (04_H × y),
RSCFDnCFDTMTRSTSyLH: <RSCFDn_base> + 0351_H + (04_H × y),
RSCFDnCFDTMTRSTSyHL: <RSCFDn_base> + 0352_H + (04_H × y),
RSCFDnCFDTMTRSTSyHH: <RSCFDn_base> + 0353_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTRSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTRSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.152 RSCFDnCFDTMTRSTSy Register Contents

Bit	Bit Name	Function
31 to 16	TMTRSTSp	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit request is present. 1: A transmit request is present.
15 to 0	TMTRSTSp	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit request is present. 1: A transmit request is present.

TMTRSTSp Flags (p = 0 to 31)

These flags indicate the status of the TMTR bit in the RSCFDnCFDTMCp register.

When the TMTR bit is set to 1 (transmission is requested), the corresponding TMTRSTSp flag is set to 1.

The corresponding TMTRSTSp flag is cleared to 0 when the TMTR bit is set to 0 (transmission is not requested) or in channel reset mode.

Table 25.153 shows the bit assignment.

Table 25.153 TMTRSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

25.4.12.2 RSCFDnCFDTMTARSTSy — Transmit Buffer Transmit Abort Request Status Register (y = 0)

Access: RSCFDnCFDTMTARSTSy register can be read only in 32-bit units
RSCFDnCFDTMTARSTSyL, RSCFDnCFDTMTARSTSyH registers can be read only in 16-bit units
RSCFDnCFDTMTARSTSyLL, RSCFDnCFDTMTARSTSyLH, RSCFDnCFDTMTARSTSyHL, RSCFDnCFDTMTARSTSyHH registers can be read only in 8-bit units

Address: RSCFDnCFDTMTARSTSy: <RSCFDn_base> + 0360_H + (04_H × y)
RSCFDnCFDTMTARSTSyL: <RSCFDn_base> + 0360_H + (04_H × y),
RSCFDnCFDTMTARSTSyH: <RSCFDn_base> + 0362_H + (04_H × y)
RSCFDnCFDTMTARSTSyLL: <RSCFDn_base> + 0360_H + (04_H × y),
RSCFDnCFDTMTARSTSyLH: <RSCFDn_base> + 0361_H + (04_H × y),
RSCFDnCFDTMTARSTSyHL: <RSCFDn_base> + 0362_H + (04_H × y),
RSCFDnCFDTMTARSTSyHH: <RSCFDn_base> + 0363_H + (04_H × y)

Value after reset: $0000\ 0001_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTARSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTARSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.154 RSCFDnCFDTMTARSTSy Register Contents

Bit	Bit Name	Function
31 to 16	TMTARSTSp	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit abort request is present. 1: A transmit abort request is present.
15 to 0	TMTARSTSp	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit abort request is present. 1: A transmit abort request is present.

TMTARSTSp Flags (p = 0 to 31)

These flags indicate the status of the TMTAR bit in the RSCFDnCFDTMCp register.

When the TMTAR bit is set to 1 (transmit abort is requested), the corresponding TMTARSTSp flag is set to 1.

The corresponding TMTARSTSp flag is cleared to 0 when the TMTAR bit is set to 0 (transmit abort is not requested) or in channel reset mode.

Table 25.155 shows the bit assignment.

Table 25.155 TMTARSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

25.4.12.3 RSCFDnCFDTMTCSTSy — Transmit Buffer Transmit Complete Status Register (y = 0)

Access: RSCFDnCFDTMTCSTSy register can be read only in 32-bit units
RSCFDnCFDTMTCSTSyL, RSCFDnCFDTMTCSTSyH registers can be read only in 16-bit units
RSCFDnCFDTMTCSTSyLL, RSCFDnCFDTMTCSTSyLH, RSCFDnCFDTMTCSTSyHL, RSCFDnCFDTMTCSTSyHH registers can be read only in 8-bit units

Address: RSCFDnCFDTMTCSTSy: <RSCFDn_base> + 0370_H + (04_H × y)
RSCFDnCFDTMTCSTSyL: <RSCFDn_base> + 0370_H + (04_H × y),
RSCFDnCFDTMTCSTSyH: <RSCFDn_base> + 0372_H + (04_H × y)
RSCFDnCFDTMTCSTSyLL: <RSCFDn_base> + 0370_H + (04_H × y),
RSCFDnCFDTMTCSTSyLH: <RSCFDn_base> + 0371_H + (04_H × y),
RSCFDnCFDTMTCSTSyHL: <RSCFDn_base> + 0372_H + (04_H × y),
RSCFDnCFDTMTCSTSyHH: <RSCFDn_base> + 0373_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTCSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTCSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.156 RSCFDnCFDTMTCSTSy Register Contents

Bit	Bit Name	Function
31 to 16	TMTCSTSp	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission has not been completed. 1: Transmission has been completed.
15 to 0	TMTCSTSp	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission has not been completed. 1: Transmission has been completed.

TMTCSTSp Flags (p = 0 to 31)

When the TMTRF[1:0] flag in the RSCFDnCFDTMSTSp register is set to 10_B (transmission has been completed (without transmit abort request)) or 11_B (transmission has been completed (with transmit abort request)), the corresponding TMTCSTSp flag is set to 1.

A TMTCSTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00_B or in channel reset mode.

Table 25.157 shows the bit assignment.

Table 25.157 TMTCSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

25.4.12.4 RSCFDnCFDTMTASTSy — Transmit Buffer Transmit Abort Status Register (y = 0)

Access: RSCFDnCFDTMTASTSy register can be read only in 32-bit units
RSCFDnCFDTMTASTSyL, RSCFDnCFDTMTASTSyH registers can be read only in 16-bit units
RSCFDnCFDTMTASTSyLL, RSCFDnCFDTMTASTSyLH, RSCFDnCFDTMTASTSyHL, RSCFDnCFDTMTASTSyHH registers can be read only in 8-bit units

Address: RSCFDnCFDTMTASTSy: <RSCFDn_base> + 0380_H + (04_H × y)
RSCFDnCFDTMTASTSyL: <RSCFDn_base> + 0380_H + (04_H × y),
RSCFDnCFDTMTASTSyH: <RSCFDn_base> + 0382_H + (04_H × y)
RSCFDnCFDTMTASTSyLL: <RSCFDn_base> + 0380_H + (04_H × y),
RSCFDnCFDTMTASTSyLH: <RSCFDn_base> + 0381_H + (04_H × y),
RSCFDnCFDTMTASTSyHL: <RSCFDn_base> + 0382_H + (04_H × y),
RSCFDnCFDTMTASTSyHH: <RSCFDn_base> + 0383_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTASTSp (p = y × 32 + 31 to y x 32 + 16 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTASTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.158 RSCFDnCFDTMTASTSy Register Contents

Bit	Bit Name	Function
31 to 16	TMTASTSp	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission is not aborted 1: Transmission is aborted
15 to 0	TMTASTSp	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission is not aborted. 1: Transmission is aborted.

TMTASTSp Flags (p = 0 to 31)

When the TMTRF[1:0] flag in the RSCFDnCFDTMSTSp register is set to 01_B (transmit abort has been completed), the corresponding TMTASTSp flag is set to 1.

A TMTASTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00_B or in channel reset mode.

Table 25.159 shows the bit assignment.

Table 25.159 TMTASTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

25.4.13 Details of Transmit Queue-related Registers

25.4.13.1 RSCFDnCFDTXQCCm — Transmit Queue Configuration and Control Register (m = 0, 1)

Access: RSCFDnCFDTXQCCm register can be read/written in 32-bit units
RSCFDnCFDTXQCCmL, RSCFDnCFDTXQCCmH registers can be read/written in 16-bit units
RSCFDnCFDTXQCCmLL, RSCFDnCFDTXQCCmLH, RSCFDnCFDTXQCCmHL, RSCFDnCFDTXQCCmHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTXQCCm: <RSCFDn_base> + $03A0_H$ + (04_H × m)
RSCFDnCFDTXQCCmL: <RSCFDn_base> + $03A0_H$ + (04_H × m),
RSCFDnCFDTXQCCmH: <RSCFDn_base> + $03A2_H$ + (04_H × m)
RSCFDnCFDTXQCCmLL: <RSCFDn_base> + $03A0_H$ + (04_H × m),
RSCFDnCFDTXQCCmLH: <RSCFDn_base> + $03A1_H$ + (04_H × m),
RSCFDnCFDTXQCCmHL: <RSCFDn_base> + $03A2_H$ + (04_H × m),
RSCFDnCFDTXQCCmHH: <RSCFDn_base> + $03A3_H$ + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQIM	TXQIE	TXQDC[3:0]				—	—	—	—	—	—	—	TXQE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 25.160 RSCFDnCFDTXQCCm Register Contents

Bit	Bit Name	Function
31 to 14	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
13	TXQIM	Transmit Queue Interrupt Source Select 0: When the transmit queue becomes empty upon completion of message transmission, a transmit queue interrupt request is generated. 1: A transmit queue interrupt request is generated each time a message has been transmitted.
12	TXQIE	Transmit Queue Interrupt Enable 0: Transmit queue interrupt is disabled. 1: Transmit queue interrupt is enabled.
11 to 8	TXQDC[3:0]	Transmit Queue Depth Configuration Setting these bits to g (g = 2 to 15) makes the (g + 1)-buffer transmit queue available. Setting these bits to 0 disables the transmit queue. Setting these bits to 1 is prohibited. For transmit buffer merge mode, set g to 2 to 9.
7 to 1	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
0	TXQE	Transmit Queue Enable 0: The transmit queue is not used. 1: The transmit queue is used.

TXQIM Bit

This bit is used to select a transmit queue interrupt source. Modify this bit in channel reset mode.

TXQIE Bit

When the TXQIE bit is set to 1 and the source selected by the TXQIM bit occurs, an interrupt request is generated.
Set the TXQE bit to 0 before modifying the TXQIE bit.

TXQDC[3:0] Bits

These bits are used to specify the number of transmit buffers to be allocated to the transmit queues. Transmit buffers are allocated to transmit queues in descending order of buffer number, that is, from (m × 16 + 15) to (m × 16 + 0). For examples of how buffer allocation is done, see Figure 25.9.

When the TMME bit in the RSCFDnCFDCmFDCFG register is 1 (transmit buffer merge mode), transmit buffers (m × 16 + 5) to (m × 16 + 0) are merged and cannot be allocated to the transmit queue. Therefore, do not set TXQDC[3:0] bits to 10 to 15.

Modify these bits only in channel reset mode.

TXQE Bit

Setting this bit to 1 makes the transmit queue available. Modify this bit in channel communication mode or channel halt mode. This bit is cleared to 0 in channel reset mode.

Before setting the TXQE bit to 1, set the TXQDC[3:0] bits to 0010_B or more.

25.4.13.2 RSCFDnCFDTXQSTSm — Transmit Queue Status Register (m = 0, 1)

Access: RSCFDnCFDTXQSTSm register can be read/written in 32-bit units
RSCFDnCFDTXQSTSmL, RSCFDnCFDTXQSTSmH registers can be read/written in 16-bit units
RSCFDnCFDTXQSTSmLL, RSCFDnCFDTXQSTSmLH, RSCFDnCFDTXQSTSmHL, RSCFDnCFDTXQSTSmHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTXQSTSm: <RSCFDn_base> + $03C0_H$ + (04_H × m)
RSCFDnCFDTXQSTSmL: <RSCFDn_base> + $03C0_H$ + (04_H × m),
RSCFDnCFDTXQSTSmH: <RSCFDn_base> + $03C2_H$ + (04_H × m)
RSCFDnCFDTXQSTSmLL: <RSCFDn_base> + $03C0_H$ + (04_H × m),
RSCFDnCFDTXQSTSmLH: <RSCFDn_base> + $03C1_H$ + (04_H × m),
RSCFDnCFDTXQSTSmHL: <RSCFDn_base> + $03C2_H$ + (04_H × m),
RSCFDnCFDTXQSTSmHH: <RSCFDn_base> + $03C3_H$ + (04_H × m)

Value after reset: $0000\ 0001_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQIF	TXQFLL	TXQEMP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 25.161 RSCFDnCFDTXQSTSm Register Contents

Bit	Bit Name	Function
31 to 13	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
12 to 8	Reserved	When read, the undefined value is returned. When writing to these bits, write the value after reset.
7 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	TXQIF	Transmit Queue Interrupt Request Flag 0: No transmit queue interrupt request is present. 1: A transmit queue interrupt request is present.
1	TXQFLL	Transmit Queue Full Status Flag 0: The transmit queue is not full. 1: The transmit queue is full.
0	TXQEMP	Transmit Queue Empty Status Flag 0: The transmit queue contains messages. 1: The transmit queue contains no message (transmit queue empty).

TXQIF Flag

The TXQIF flag is set to 1 when the event specified by the TXQIM bit in the RSCFDnCFDTXQCCm register has occurred.

The TXQIF flag is cleared to 0 in channel reset mode or by writing 0 to this flag. This flag is not cleared to 0 by setting the TXQE bit in the RSCFDnCFDTXQCCm register to 0 (the transmit queue is not used).

TXQFLL Flag

The TXQFLL flag is set to 1 when the number of messages set for the transmit queue matches the transmit queue depth set by the TXQDC[3:0] bits in the RSCFDnCFDTXQCCm register.
This flag is cleared to 0 in any of the following cases.

- The number of messages set for the transmit queue is smaller than the transmit queue depth set by the TXQDC[3:0] bits.
- In channel reset mode

TXQEMP Flag

The TXQEMP flag is cleared to 0 when even a single message is set for the transmit queue.
This flag is set to 1 in any of the following cases.

- The TXQE bit is set to 0 (the transmit queue is not used).
- The transmit queue becomes empty.
- In channel reset mode

25.4.13.3 RSCFDnCFDTXQPCTRm — Transmit Queue Pointer Control Register (m = 0, 1)

Access: RSCFDnCFDTXQPCTRm register can only be written in 32-bit units
RSCFDnCFDTXQPCTRmL, RSCFDnCFDTXQPCTRmH registers can only be written in 16-bit units
RSCFDnCFDTXQPCTRmLL, RSCFDnCFDTXQPCTRmLH, RSCFDnCFDTXQPCTRmHL, RSCFDnCFDTXQPCTRmHH registers can only be written in 8-bit units

Address: RSCFDnCFDTXQPCTRm: <RSCFDn_base> + $03E0_H$ + (04_H × m)
RSCFDnCFDTXQPCTRmL: <RSCFDn_base> + $03E0_H$ + (04_H × m),
RSCFDnCFDTXQPCTRmH: <RSCFDn_base> + $03E2_H$ + (04_H × m)
RSCFDnCFDTXQPCTRmLL: <RSCFDn_base> + $03E0_H$ + (04_H × m),
RSCFDnCFDTXQPCTRmLH: <RSCFDn_base> + $03E1_H$ + (04_H × m),
RSCFDnCFDTXQPCTRmHL: <RSCFDn_base> + $03E2_H$ + (04_H × m),
RSCFDnCFDTXQPCTRmHH: <RSCFDn_base> + $03E3_H$ + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 25.162 RSCFDnCFDTXQPCTRm Register Contents

Bit	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	TXQPC[7:0]	Transmit Queue Pointer Control Writing FF_H to these bits moves the write pointer of the transmit queue to the next queue buffer.

TXQPC[7:0] Bits

Writing FF_H to the TXQPC[7:0] bits moves the write pointer to the next transmit queue buffer and generates a transmit request of the message. Write transmit messages to the RSCFDnCFDTMIDp, RSCFDnCFDTMPTRp, RSCFDnCFDTMFDCTRp, and RSCFDnCFDTMDFb_p registers (p = 15 and 31) before writing FF_H to the TXQPC[7:0] bits.

When writing FF_H to these bits, make sure that the TXQE bit in the RSCFDnCFDTXQCCm register is set to 1 (the transmit queue is used) and the TXQFLL flag in the RSCFDnCFDTXQSTSm register is 0 (he transmit queue is not full).

25.4.14 Details of Transmit History-related Registers

25.4.14.1 RSCFDnCFDTHLCCm — Transmit History Configuration and Control Register (m = 0, 1)

Access: RSCFDnCFDTHLCCm register can be read/written in 32-bit units
RSCFDnCFDTHLCCmL, RSCFDnCFDTHLCCmH registers can be read/written in 16-bit units
RSCFDnCFDTHLCCmLL, RSCFDnCFDTHLCCmLH, RSCFDnCFDTHLCCmHL, RSCFDnCFDTHLCCmHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTHLCCm: <RSCFDn_base> + 0400_H + (04_H × m)
RSCFDnCFDTHLCCmL: <RSCFDn_base> + 0400_H + (04_H × m),
RSCFDnCFDTHLCCmH: <RSCFDn_base> + 0402_H + (04_H × m)
RSCFDnCFDTHLCCmLL: <RSCFDn_base> + 0400_H + (04_H × m),
RSCFDnCFDTHLCCmLH: <RSCFDn_base> + 0401_H + (04_H × m),
RSCFDnCFDTHLCCmHL: <RSCFDn_base> + 0402_H + (04_H × m),
RSCFDnCFDTHLCCmHH: <RSCFDn_base> + 0403_H + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLDTE	THLIM	THLIE	—	—	—	—	—	—	—	THLE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 25.163 RSCFDnCFDTHLCCm Register Contents

Bit	Bit Name	Function
31 to 11	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
10	THLDTE	Transmit History Target Buffer Select 0: Entry from transmit/receive FIFO buffers and transmit queue 1: Entry from transmit buffers, transmit/receive FIFO buffers, and transmit queue
9	THLIM	Transmit History Interrupt Source Select 0: When 12 sets of data have been stored in the transmit history buffer 1: When a single set of transmit history data has been stored
8	THLIE	Transmit History Interrupt Enable 0: Transmit history interrupt is disabled. 1: Transmit history interrupt is enabled.
7 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	THLE	Transmit History Buffer Enable 0: Transmit history buffer is not used. 1: Transmit history buffer is used.

THLDTE Bit

When this bit is set to 0, the transmit history data of messages transmitted from transmit/receive FIFO buffers and the transmit queue is stored in the transmit history buffer. When this bit is set to 1, the transmit history data of messages transmitted from transmit buffers, transmit/receive FIFO buffers, and the transmit queue is stored in the transmit history buffer.

Modify this bit only in channel reset mode.

THLIM Bit

This bit is used to select a transmit history interrupt source.
Modify this bit only in channel reset mode.

THLIE Bit

When the THLIE bit is set to 1 and the source selected by the THLIM bit has occurred, a transmit history interrupt request is generated. Modify the THLIE bit only when the THLE bit set to 0.

THLE Bit

Setting this bit to 1 makes the transmit history buffer available. When data transmission from the buffer selected by the THLDTE bit has been completed, the transmit history data of transmit messages is stored in the transmit history buffer. Modify this bit in channel communication mode or channel halt mode.
This bit is cleared to 0 in channel reset mode.

25.4.14.2 RSCFDnCFDTHLSTSm — Transmit History Status Register (m = 0, 1)

Access: RSCFDnCFDTHLSTSm register can be read/written in 32-bit units
RSCFDnCFDTHLSTSmL, RSCFDnCFDTHLSTSmH registers can be read/written in 16-bit units
RSCFDnCFDTHLSTSmLL, RSCFDnCFDTHLSTSmLH, RSCFDnCFDTHLSTSmHL, RSCFDnCFDTHLSTSmHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTHLSTSm: <RSCFDn_base> + 0420_H + (04_H × m)
RSCFDnCFDTHLSTSmL: <RSCFDn_base> + 0420_H + (04_H × m),
RSCFDnCFDTHLSTSmH: <RSCFDn_base> + 0422_H + (04_H × m)
RSCFDnCFDTHLSTSmLL: <RSCFDn_base> + 0420_H + (04_H × m),
RSCFDnCFDTHLSTSmLH: <RSCFDn_base> + 0421_H + (04_H × m),
RSCFDnCFDTHLSTSmHL: <RSCFDn_base> + 0422_H + (04_H × m),
RSCFDnCFDTHLSTSmHH: <RSCFDn_base> + 0423_H + (04_H × m)

Value after reset: $0000\ 0001_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THLMC[4:0]					—	—	—	—	THLIF	THLELT	THLFLL	THLEMP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state

Table 25.164 RSCFDnCFDTHLSTSm Register Contents

Bit	Bit Name	Function
31 to 13	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
12 to 8	THLMC[4:0]	Transmit History Buffer Unread Data Counter These bits indicate the number of unread data sets stored in the transmit history buffer.
7 to 4	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
3	THLIF	Transmit History Interrupt Request Flag 0: No transmit history interrupt request is present. 1: A transmit history interrupt request is present.
2	THLELT	Transmit History Buffer Overflow Flag 0: Transmit history buffer overflow has not occurred. 1: Transmit history buffer overflow has occurred.
1	THLFLL	Transmit history Buffer Full Status Flag 0: Transmit history buffer is not full. 1: Transmit history buffer is full.
0	THLEMP	Transmit History Buffer Empty Status Flag 0: Transmit history buffer contains unread data. 1: Transmit history buffer contains no unread data (buffer empty).

THLMC[4:0] Bits

These bits indicate the number of unread data sets stored in the transmit history buffer. These bits are cleared to 0 in channel reset mode.

THLIF Flag

The THLIF flag is set to 1 when the interrupt source specified with the THLIM bit in the RSCFDnCFDTHLCCm register occurs.
This flag is cleared to 0 in channel reset mode or by the program writing 0 to this flag.
To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.
When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

THLELT Flag

The THLELT flag is set to 1 when an attempt is made to store new transmit history data while the transmit history buffer is full. In this case, the new data is discarded. This flag becomes 0 in channel reset mode or by the program writing 0 to this flag.
To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.
When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

THLFLL Flag

The THLFLL flag is set to 1 when 16 data sets have been stored in the transmit history buffer, and is cleared to 0 when the number of data sets stored in the transmit history buffer has decreased to less than 16. This bit is also cleared to 0 in channel reset mode or when the THLE bit in the RSCFDnCFDTHLCCm register is set to 0 (transmit history buffer is not used).

THLEMP Flag

The THLEMP flag is cleared to 0 when even a single set of transmit history data has been stored in the transmit history buffer.
This flag is set to 1 when all the data in the transmit history buffer has been read. This flag is also set to 1 in channel reset mode or when the THLE bit in the RSCFDnCFDTHLCCm register is set to 0 (transmit history buffer is not used).

Note: To clear THLIF or THLELT flag to 0, the program must write 0. When writing, use a store instruction to write “0” to the given flag and “1” to other flags.

25.4.14.3 RSCFDnCFDTHLPCTRm — Transmit History Pointer Control Register (m = 0, 1)

Access: RSCFDnCFDTHLPCTRm register can only be written in 32-bit units
RSCFDnCFDTHLPCTRmL, RSCFDnCFDTHLPCTRmH registers can only be written in 16-bit units
RSCFDnCFDTHLPCTRmLL, RSCFDnCFDTHLPCTRmLH, RSCFDnCFDTHLPCTRmHL, RSCFDnCFDTHLPCTRmHH registers can only be written in 8-bit units

Address: RSCFDnCFDTHLPCTRm: <RSCFDn_base> + 0440_H + (04_H × m)
RSCFDnCFDTHLPCTRmL: <RSCFDn_base> + 0440_H + (04_H × m),
RSCFDnCFDTHLPCTRmH: <RSCFDn_base> + 0442_H + (04_H × m)
RSCFDnCFDTHLPCTRmLL: <RSCFDn_base> + 0440_H + (04_H × m),
RSCFDnCFDTHLPCTRmLH: <RSCFDn_base> + 0441_H + (04_H × m),
RSCFDnCFDTHLPCTRmHL: <RSCFDn_base> + 0442_H + (04_H × m),
RSCFDnCFDTHLPCTRmHH: <RSCFDn_base> + 0443_H + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	THLPC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 25.165 RSCFDnCFDTHLPCTRm Register Contents

Bit	Bit Name	Function
31 to 8	Reserved	When writing to these bits, write the value after reset.
7 to 0	THLPC[7:0]	Transmit History List Pointer Control Writing FF_H to these bits moves the read pointer to the next unread data in the transmit history buffer.

THLPC[7:0] Bits

When the THLPC[7:0] bits are set to FF_H, the read pointer moves to the next data in the transmit history buffer. At this time, the THLMC[4:0] (transmit history buffer unread data counter) value in the RSCFDnCFDTHLSTSm register is decremented. Write FF_H to the THLPC[7:0] bits after reading from the RSCFDnCFDTHLACCm register.
When writing FF_H to these bits, make sure that the THLE bit in the RSCFDnCFDTHLCCm register is set to 1 (transmit history buffer is used) and the THLEMP flag in the RSCFDnCFDTHLSTSm register is 0.

25.4.14.4 RSCFDnCFDTHLACCm — Transmit History Access Register (m = 0, 1)

Access: RSCFDnCFDTHLACCm register can be read only in 32-bit units
RSCFDnCFDTHLACCmL, RSCFDnCFDTHLACCmH registers can be read only in 16-bit units
RSCFDnCFDTHLACCmLL, RSCFDnCFDTHLACCmLH, RSCFDnCFDTHLACCmHL, RSCFDnCFDTHLACCmHH registers can be read only in 8-bit units

Address: RSCFDnCFDTHLACCm: <RSCFDn_base> + 6000_H + (04_H × m)
RSCFDnCFDTHLACCmL: <RSCFDn_base> + 6000_H + (04_H × m),
RSCFDnCFDTHLACCmH: <RSCFDn_base> + 6002_H + (04_H × m)
RSCFDnCFDTHLACCmLL: <RSCFDn_base> + 6000_H + (04_H × m),
RSCFDnCFDTHLACCmLH: <RSCFDn_base> + 6001_H + (04_H × m),
RSCFDnCFDTHLACCmHL: <RSCFDn_base> + 6002_H + (04_H × m),
RSCFDnCFDTHLACCmHH: <RSCFDn_base> + 6003_H + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TID[7:0]								—	BN[3:0]				BT[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.166 RSCFDnCFDTHLACCm Register Contents

Bit	Bit Name	Function
31 to 16	TMTS[15:0]	Timestamp Data The timestamp data of stored data can be read.
15 to 8	TID[7:0]	Label Data The label information of stored data can be read.
7	Reserved	This bit is read as the value after reset.
6 to 3	BN[3:0]	Buffer Number Data The buffer number of transmit source (transmit buffer, transmit/receive FIFO or transmit queue) can be read.
2 to 0	BT[2:0]	Buffer Type Data b2 b1 b0 0 0 1: Transmit buffer 0 1 0: Transmit/receive FIFO buffer 1 0 0: Transmit queue

TMTS[15:0] Bits

Timestamp values in transmit history data stored in the transmit history buffer are displayed.

TID[7:0] Bits

These bits indicate the label information of transmit history data stored in the transmit history buffer.

BN[3:0] Bits

These bits indicate the transmit source buffer number in the transmit history data stored in the transmit history buffer.

BT[2:0] Bits

These bits indicate the type of the transmit source buffer in the transmit history data stored in the transmit history buffer.

25.4.15 Details of Test-related Registers

25.4.15.1 RSCFDnCFDGTSTCFG — Global Test Configuration Register

Access: RSCFDnCFDGTSTCFG register can be read/written in 32-bit units
RSCFDnCFDGTSTCFGL, RSCFDnCFDGTSTCFGH registers can be read/written in 16-bit units
RSCFDnCFDGTSTCFGLL, RSCFDnCFDGTSTCFGLH, RSCFDnCFDGTSTCFGHL, RSCFDnCFDGTSTCFGHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGTSTCFG: <RSCFDn_base> + 0468_H
RSCFDnCFDGTSTCFGL: <RSCFDn_base> + 0468_H, RSCFDnCFDGTSTCFGH: <RSCFDn_base> + $046A_H$
RSCFDnCFDGTSTCFGLL: <RSCFDn_base> + 0468_H, RSCFDnCFDGTSTCFGLH: <RSCFDn_base> + 0469_H,
RSCFDnCFDGTSTCFGHL: <RSCFDn_base> + $046A_H$, RSCFDnCFDGTSTCFGHH: <RSCFDn_base> + $046B_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	RTMPS[6:0]						
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	C1ICBCE	C0ICBCE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 25.167 RSCFDnCFDGTSTCFG Register Contents

Bit	Bit Name	Function
31 to 23	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
22 to 16	RTMPS[6:0]	RAM Test Page Configuration Set a value within a range of page 0 (00_H) to page 27 ($1B_H$).
15 to 2	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
1	C1ICBCE	CAN1 Inter-Channel Communication Test Enable 0: CAN1 inter-channel communication test is disabled. 1: CAN1 inter-channel communication test is enabled.
0	C0ICBCE	CAN0 Inter-Channel Communication Test Enable 0: CAN0 inter-channel communication test is disabled. 1: CAN0 inter-channel communication test is enabled.

Modify the RSCFDnCFDGTSTCFG register only in global test mode.

RTMPS[6:0] Bits

These bits are used to set the RAM test target page number for RAM test. Set a value in the range of 00_H to $1B_H$, inclusive. Do not access the RAM area higher than the 192th byte on the last page (the setting of the RTMPS bit is $1B_H$).

C1ICBCE Bit

Setting this bit to 1 enables the channel 1 inter-channel communication test.
This bit is cleared to 0 in global reset mode.

C0ICBCE Bit

Setting this bit to 1 enables the channel 0 inter-channel communication test.
This bit is cleared to 0 in global reset mode.

25.4.15.2 RSCFDnCFDGTSTCTR — Global Test Control Register

Access: RSCFDnCFDGTSTCTR register can be read/written in 32-bit units
RSCFDnCFDGTSTCTRL, RSCFDnCFDGTSTCTRH registers can be read/written in 16-bit units
RSCFDnCFDGTSTCTRLL, RSCFDnCFDGTSTCTRLH, RSCFDnCFDGTSTCTRHL, RSCFDnCFDGTSTCTRHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGTSTCTR: <RSCFDn_base> + $046C_H$
RSCFDnCFDGTSTCTRL: <RSCFDn_base> + $046C_H$, RSCFDnCFDGTSTCTRH: <RSCFDn_base> + $046E_H$
RSCFDnCFDGTSTCTRLL: <RSCFDn_base> + $046C_H$, RSCFDnCFDGTSTCTRLH: <RSCFDn_base> + $046D_H$,
RSCFDnCFDGTSTCTRHL: <RSCFDn_base> + $046E_H$, RSCFDnCFDGTSTCTRHH: <RSCFDn_base> + $046F_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RTME	—	ICBCTME
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W

Table 25.168 RSCFDnCFDGTSTCTR Register Contents

Bit	Bit Name	Function
31 to 3	Reserved	These bits are read as the value after reset. The write value should be the value after reset.
2	RTME	RAM Test Enable 0: RAM test is disabled. 1: RAM test is enabled.
1	Reserved	This bit is read as the value after reset. The write value should be the value after reset.
0	ICBCTME	Communication Test between Channels Enable 0: Communication test between channels disabled 1: Communication test between channels enabled

RTME Bit

Setting this bit to 1 enables the RAM test. Modify this bit only in global test mode.
This bit is cleared to 0 in global reset mode.

1. Set the GMDC[1:0] bits in the RSCFDnCFDGCTR register to 10_B (Global test mode).
2. Set the RTME bit to 1.
3. Check that the RTME bit is set to 1.

ICBCTME Bit

When this bit is set to 1, a communication test is enabled between the channels for which the CmICBCE bit (m = 0, 1) in the RSCFDnCFDGTSTCFG register has been set to 1. Modify the ICBCTME bit only in global test mode.
This bit is cleared to 0 in global reset mode.

25.4.15.3 RSCFDnCFDGLOCKK — Global Lock Key Register

Access: RSCFDnCFDGLOCKK register can only be written in 32-bit units
RSCFDnCFDGLOCKKL, RSCFDnCFDGLOCKKH registers can only be written in 16-bit units

Address: RSCFDnCFDGLOCKK: <RSCFDn_base> + $047C_H$
RSCFDnCFDGLOCKKL: <RSCFDn_base> + $047C_H$, RSCFDnCFDGLOCKKH: <RSCFDn_base> + $047E_H$

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCK[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1

Note 1. Writing to these bits is effective only when the RS-CANFD module is in global test mode.

Table 25.169 RSCFDnCFDGLOCKK Register Contents

Bit	Bit Name	Function
31 to 16	Reserved	When writing these bits, write the value after reset.
15 to 0	LOCK[15:0]	Lock Key These bits are key bits to release protection of test mode.

The RSCFDnCFDGLOCKK register releases protection of special test bits and is write only.
For the protection release data, see section 25.11.4.2, Procedure for Releasing the Protection.

LOCK[15:0] Bits

Writing the protection release data to the LOCK[15:0] bits in succession enables writing 1 to the RTME bit in the RSCFDnCFDGTSTCTR register.
After the protection has been released, writing to the I/O register area (<RSCFDn_base> + 0000_H to <RSCFDn_base> + $05FF_H$) of the CAN (except the RAM) enables the protection again.
Reading from the I/O register area of the CAN or reading from/writing to other areas does not enable the protection.

25.4.15.4 RSCFDnCFDRPGACCr — RAM Test Page Access Register (r = 0 to 63)

Access: RSCFDnCFDRPGACCr register can be read/written in 32-bit units
RSCFDnCFDRPGACCrL, RSCFDnCFDRPGACCrH registers can be read/written in 16-bit units
RSCFDnCFDRPGACCrLL, RSCFDnCFDRPGACCrLH, RSCFDnCFDRPGACCrHL, RSCFDnCFDRPGACCrHH registers can be read/written in 8-bit units

Address: RSCFDnCFDRPGACCr: <RSCFDn_base> + 6400_H + (04_H × r)
RSCFDnCFDRPGACCrL: <RSCFDn_base> + 6400_H + (04_H × r),
RSCFDnCFDRPGACCrH: <RSCFDn_base> + 6402_H + (04_H × r)
RSCFDnCFDRPGACCrLL: <RSCFDn_base> + 6400_H + (04_H × r),
RSCFDnCFDRPGACCrLH: <RSCFDn_base> + 6401_H + (04_H × r),
RSCFDnCFDRPGACCrHL: <RSCFDn_base> + 6402_H + (04_H × r),
RSCFDnCFDRPGACCrHH: <RSCFDn_base> + 6403_H + (04_H × r)

Value after reset: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDTA[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDTA[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.170 RSCFDnCFDRPGACCr Register Contents

Bit	Bit Name	Function
31 to 0	RDTA[31:0]	RAM Data Test Access Data can be read and written in RSCAN RAM.

Modify the RSCFDnCFDRPGACCr register in global test mode with the RTME bit in the RSCFDnCFDGTSTCTR register set to 1 (RAM test is enabled).
The RSCFDnCFDRPGACCr register is readable and writable when the RTME bit is set to 1.

25.5 Interrupt Sources and DMA Trigger

25.5.1 Interrupt Sources

The RS-CANFD module has 8 interrupts that are grouped into global interrupts and channel interrupts.

- Global interrupts (2 sources):
  Receive FIFO interrupt
  Global error interrupt
- Channel interrupts (3 sources/channel):
  CANm transmit interrupt (m = 0, 1)
  - CANm transmit complete interrupt
  - CANm transmit abort interrupt
  - CANm transmit/receive FIFO transmit complete interrupt (in transmit mode, gateway mode)
  - CANm transmit history interrupt
  - CANm transmit queue Interrupt

  CANm transmit/receive FIFO receive complete interrupt (in receive mode, gateway mode)
  CANm error interrupt

When an interrupt request is generated, the corresponding interrupt request flag is set to 1 (interrupt request present). In that case, when the interrupt enable bit is set to 1 (enabling interrupts), an interrupt request is output from the RS-CANFD module. (Generation of interrupts also depends on the interrupt control register settings of the interrupt controller.)
Setting the interrupt request flag to 0 (no interrupt request present) or setting the interrupt enable bit to 0 (disabling interrupts) clears the current interrupt request. The current interrupt request is still output until the interrupt request flag is cleared.
Table 25.171 lists the CAN interrupt sources. Figure 25.2 shows the CAN global interrupt block diagram. Figure 25.3 shows the CAN channel interrupt block diagram.

Table 25.171 List of CAN Interrupt Sources

RCAN					INTC	DMA
	Interrupt Source (Unit Interrupt Signal)		Corresponding Interrupt Request Flag	Corresponding Interrupt Enable Bit	Request Source Name	Transfer Request Signal
Global interrupts	Receive FIFO (INTRCANGRECC)	Receive FIFO 0	RFIF in the RSCFDn(CFD)RFSTS0 register	RFIE in the RSCFDn(CFD)RFCC0 register	RFI	RXF_DMA0
		Receive FIFO 1	RFIF in the RSCFDn(CFD)RFSTS1 register	RFIE in the RSCFDn(CFD)RFCC1 register		RXF_DMA1
		Receive FIFO 2	RFIF in the RSCFDn(CFD)RFSTS2 register	RFIE in the RSCFDn(CFD)RFCC2 register		RXF_DMA2
		Receive FIFO 3	RFIF in the RSCFDn(CFD)RFSTS3 register	RFIE in the RSCFDn(CFD)RFCC3 register		RXF_DMA3
		Receive FIFO 4	RFIF in the RSCFDn(CFD)RFSTS4 register	RFIE in the RSCFDn(CFD)RFCC4 register		RXF_DMA4
		Receive FIFO 5	RFIF in the RSCFDn(CFD)RFSTS5 register	RFIE in the RSCFDn(CFD)RFCC5 register		RXF_DMA5
		Receive FIFO 6	RFIF in the RSCFDn(CFD)RFSTS6 register	RFIE in the RSCFDn(CFD)RFCC6 register		RXF_DMA6
		Receive FIFO 7	RFIF in the RSCFDn(CFD)RFSTS7 register	RFIE in the RSCFDn(CFD)RFCC7 register		RXF_DMA7

Table 25.171 List of CAN Interrupt Sources

RCAN					INTC	DMA
	Interrupt Source (Unit Interrupt Signal)		Corresponding Interrupt Request Flag	Corresponding Interrupt Enable Bit	Request Source Name	Transfer Request Signal
Global interrupts	Global error (INTRCANGERR)		• DEF in the RSCFDn(CFD)GERFL register • MES in the RSCFDn(CFD)GERFL register • THLES in the RSCFDn(CFD)GERFL register • CMPOF in the RSCFDnCFDGERFL register	• DEIE in the RSCFDn(CFD)GCTR register • MEIE in the RSCFDn(CFD)GCTR register • THLEIE in the RSCFDn(CFD)GCTR register • CMPOFIE in the RSCFDnCFDGCTR register	GERI	—
Channel interrupts (m = 0, 1)	CANm transmit (INTRCANmTRX)	CANm transmit complete	TMTRF[1:0] in the RSCFDn(CFD)TMSTSp register	TMIEp in the RSCFDn(CFD)TMIECy register	CTXIm	—
		CANm transmit abort	TMTRF[1:0] in the RSCFDn(CFD)TMSTSp register	TAIE in the RSCFDn(CFD)CmCTR register		—
		CANm transmit/ receive FIFO transmit complete	CFTXIF in the RSCFDn(CFD)CFSTSk register	CFTXIE in the RSCFDn(CFD)CFCCk register		—
		CANm transmit queue	TXQIF in the RSCFDn(CFD)TXQSTSm register	TXQIE in the RSCFDn(CFD)TXQCCm register		—
		CANm transmit history	THLIF in the RSCFDn(CFD)THLSTSm register	THLIE in the RSCFDn(CFD)THLCCm register		—
	CANm transmit/receive FIFO receive complete (INTRCANmREC)		CFRXIF in the RSCFDn(CFD)CFSTSk register	CFRXIE in the RSCFDn(CFD)CFCCk register	CFRXIm	COM_DMAm

Table 25.171 List of CAN Interrupt Sources

RCAN				INTC	DMA
	Interrupt Source (Unit Interrupt Signal)	Corresponding Interrupt Request Flag	Corresponding Interrupt Enable Bit	Request Source Name	Transfer Request Signal
Channel interrupts (m = 0, 1)	CANm error (INTRCANmERR)	• BEF in the RSCFDn(CFD)CmERFL register • ALF in the RSCFDn(CFD)CmERFL register • BLF in the RSCFDn(CFD)CmERFL register • OVLF in the RSCFDn(CFD)CmERFL register • BORF in the RSCFDn(CFD)CmERFL register • BOEF in the RSCFDn(CFD)CmERFL register • EPF in the RSCFDn(CFD)CmERFL register • EWF in the RSCFDn(CFD)CmERFL register • SOCO in the RSCFDnCFDCmFDSTS register • EOCO in the RSCFDnCFDCmFDSTS register • TDCVF in the RSCFDnCFDCmFDSTS register	• BEIE in the RSCFDn(CFD)CmCTR register • ALIE in the RSCFDn(CFD)CmCTR register • BLIE in the RSCFDn(CFD)CmCTR register • OLIE in the RSCFDn(CFD)CmCTR register • BORIE in the RSCFDn(CFD)CmCTR register • BOEIE in the RSCFDn(CFD)CmCTR register • EPIE in the RSCFDn(CFD)CmCTR register • EWIE in the RSCFDn(CFD)CmCTR register • SOCOIE in the RSCFDnCFDCmCTR register • EOCOIE in the RSCFDnCFDCmCTR register • TDCVFIE in the RSCFDnCFDCmCTR register	CERIm	—

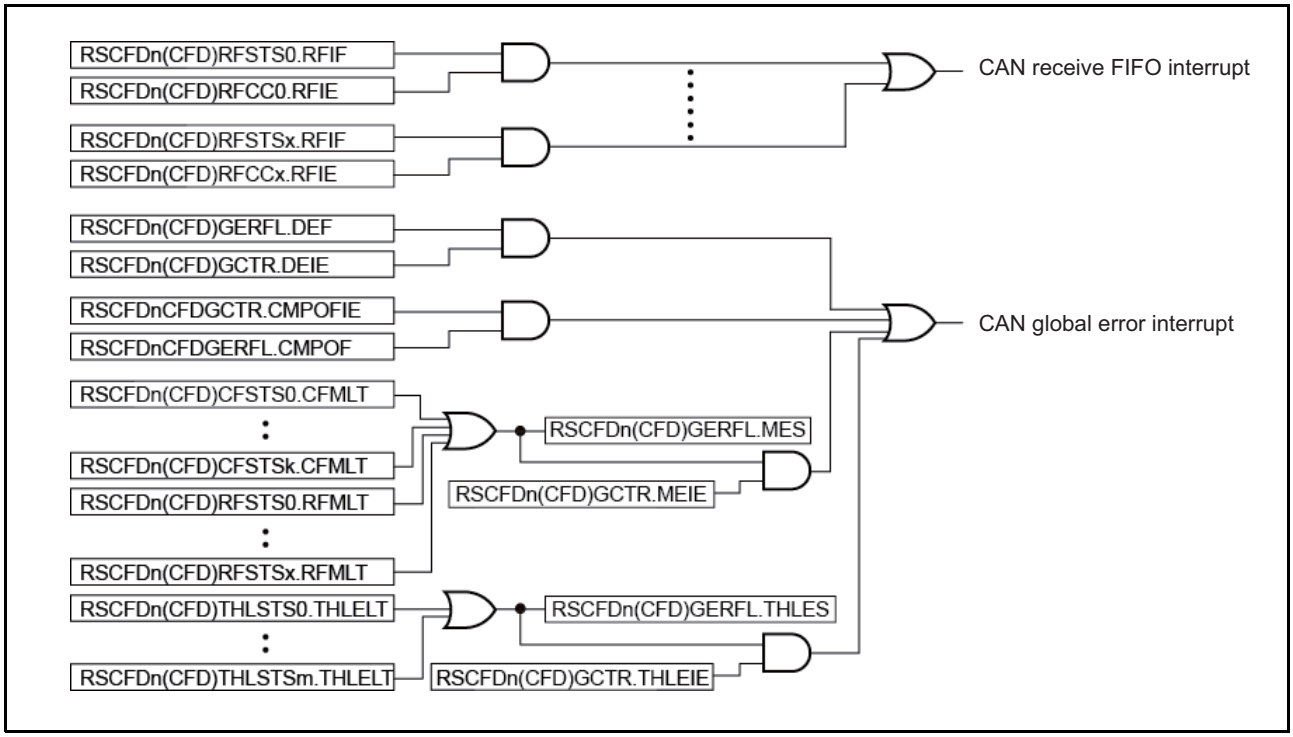


Figure 25.2 CAN Global Interrupt Block Diagram

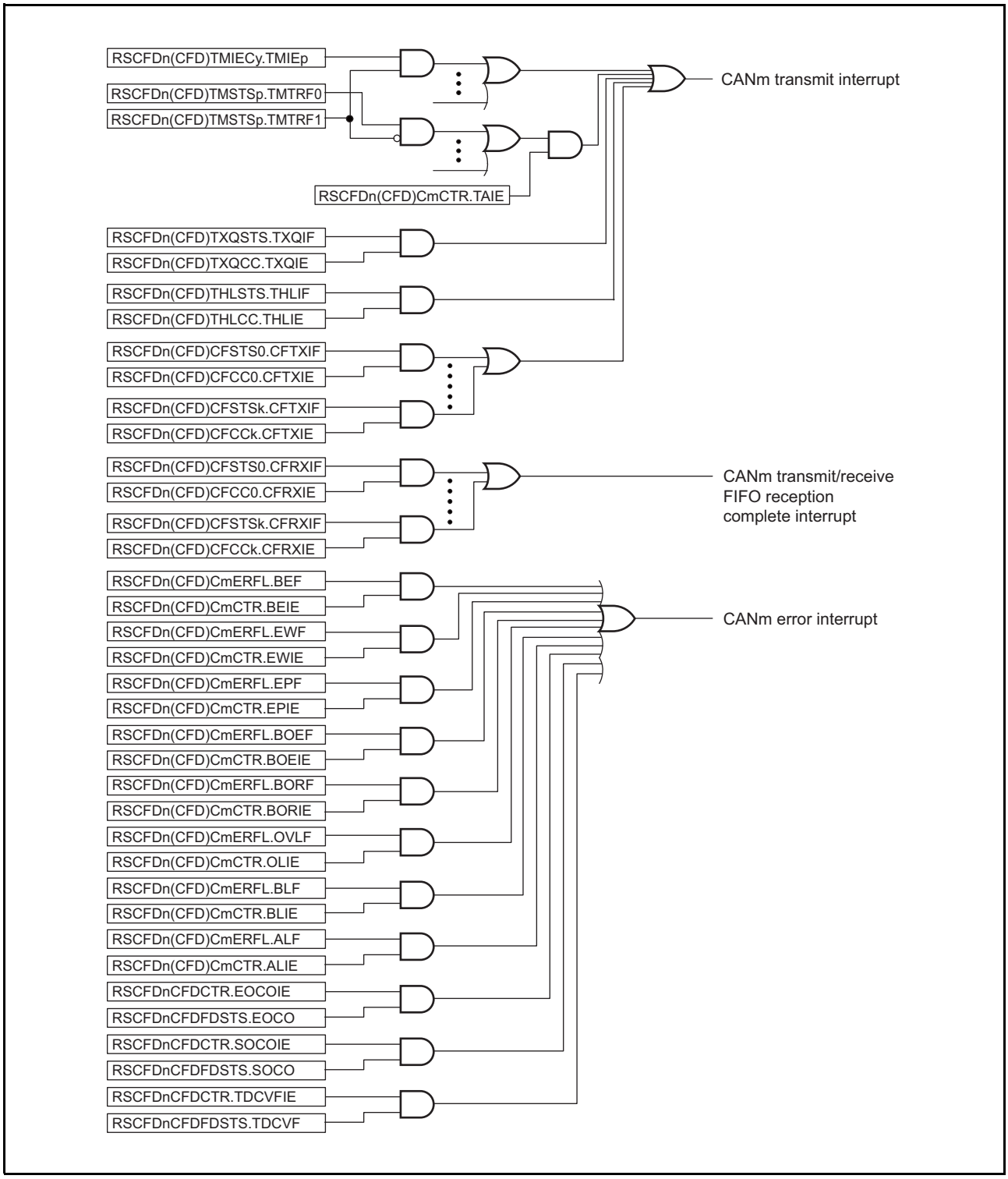


Figure 25.3 CAN Channel Interrupt Block Diagram

25.5.2 DMA Trigger (Only in CANFD Mode)

In CANFD mode, receive FIFO buffers can be related to DMA channels. The following 10 FIFO buffers can be related.

- All receive FIFO buffers x (x = 0 to 7)
- The first transmit/receive FIFO buffer k (k = 3 × m, m = 0, 1) allocated to channel m

When the DMA enable bit (RFDMAEx or CFDMAEm bit in the RSCFDnCFDCDTCT register) is set to 1 and an unread message is remaining in the related FIFO, a DMA transfer request trigger is generated.

25.6 CAN Modes

The RS-CANFD module has four global modes to control the entire RS-CANFD module status and four channel modes to control individual channel status. Details of global modes are described in section 25.6.1, Global Modes, and details of channel modes are described in section 25.6.2, Table 25.5, Channel Mode State Transition Chart.

- Global stop mode:Stops the clocks of the entire module to achieve low power consumption.
- Global reset mode:Performs initial settings for the entire module.
- Global test mode:Performs test settings and performs the RAM test.
- Global operating mode:Makes the entire module operable.
- Channel stop mode:Stops the channel clock.
- Channel reset mode:Performs initial settings for the channels.
- Channel halt mode:Stops CAN communication and allows channel testing.
- Channel communication mode:Performs CAN communication.

25.6.1 Global Modes

Figure 25.4 shows the transitions of global modes.

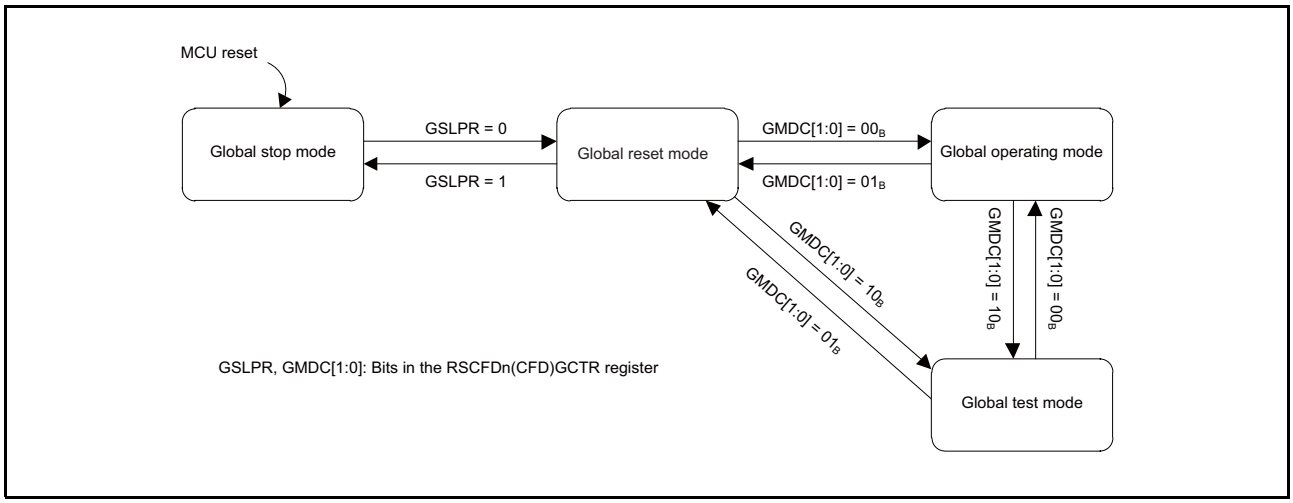


Figure 25.4 **Transitions of Global Modes**

In some cases, global mode transitions also force channel mode transitions. Table 25.172 shows the channel mode transitions depending on the global mode setting dictated by the GMDC[1:0] bits and the GSLPR bit.

Table 25.172 Transitions of Channel Modes Depending on Global Mode Setting (GMDC[1:0] and GSLPR Bits)

	Channel Mode after Setting			
Channel Mode before Setting	GMDC[1:0] = 00_B GSLPR = 0 (Global Operation)	GMDC[1:0] = 10_B GSLPR = 0 (Global Test)	GMDC[1:0] = 01_B GSLPR = 0 (Global Reset)	GMDC[1:0] = 01_B GSLPR = 1 (Global Stop)
Channel communication	Channel communication	Channel halt	Channel reset	Transition prohibited
Channel halt	Channel halt	Channel halt	Channel reset	Transition prohibited
Channel reset	Channel reset	Channel reset	Channel reset	Channel stop
Channel stop	Channel stop	Channel stop	Channel stop	Channel stop

Note 1. GMDC[1:0], GSLPR: Bits in the RSCFDn(CFD)GCTR register

Table 25.173 shows the global mode transition time.

Table 25.173 Global Mode Transition Time

Mode before Transition	Mode after Transition	Maximum Transition Time
Global stop	Global reset	Three pclk cycles
Global reset	Global stop	Three pclk cycles
Global reset	Global test	Ten pclk cycles
Global reset	Global operating	Ten pclk cycles
Global test	Global reset	Two CAN bit times,*1, *2
Global test	Global operating	Three pclk cycles
Global operating	Global reset	Two CAN bit times,*1, *2
Global operating	Global test	Two CAN frames,*1

Note 1. CAN frame time and CAN bit time of the lowest communication speed of the channels in use
Note 2. In CANFD mode, this time value is the CAN bit time of the nominal bit rate.

25.6.1.1 Global Stop Mode

In global stop mode, clocks of the CAN do not run and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained. Only the clock used by the CPU for writing to the GSLPR bit runs in this mode.
After the MCU is reset, the CAN module transitions to global stop mode. Setting the GSLPR bit in the RSCFDn(CFD)GCTR register to 1 (in global stop mode) in global reset mode sets the CSLPR bit in each of the RSCFDn(CFD)CmCTR register to 1 (channel stop mode). Afterwards, if all channels are forced to transition to channel stop mode, the CAN module transitions to global stop mode. The GSLPR bit should not be modified in global operating mode or global test mode.

25.6.1.2 Global Reset Mode

In global reset mode, RS-CANFD module settings are performed. When the RS-CANFD module transitions to global reset mode, some registers are initialized. For registers to be initialized, see Table 25.176, Registers Initialized in Global Reset Mode or Channel Reset Mode and Table 25.177, Registers Initialized Only in Global Reset Mode.
Setting the GMDC[1:0] bits in the RSCFDn(CFD)GCTR register to 01_B sets the CHMDC[1:0] bits in each of the RSCFDn(CFD)CmCTR registers (m = 0, 1) to 01_B (channel reset mode). If all channels are forced to transition to channel reset mode, the CAN module transitions to global reset mode. Channels that are already in channel reset mode or channel stop mode do not transition (because the CHMDC[1:0] bits have already been set to 01_B).

25.6.1.3 Global Test Mode

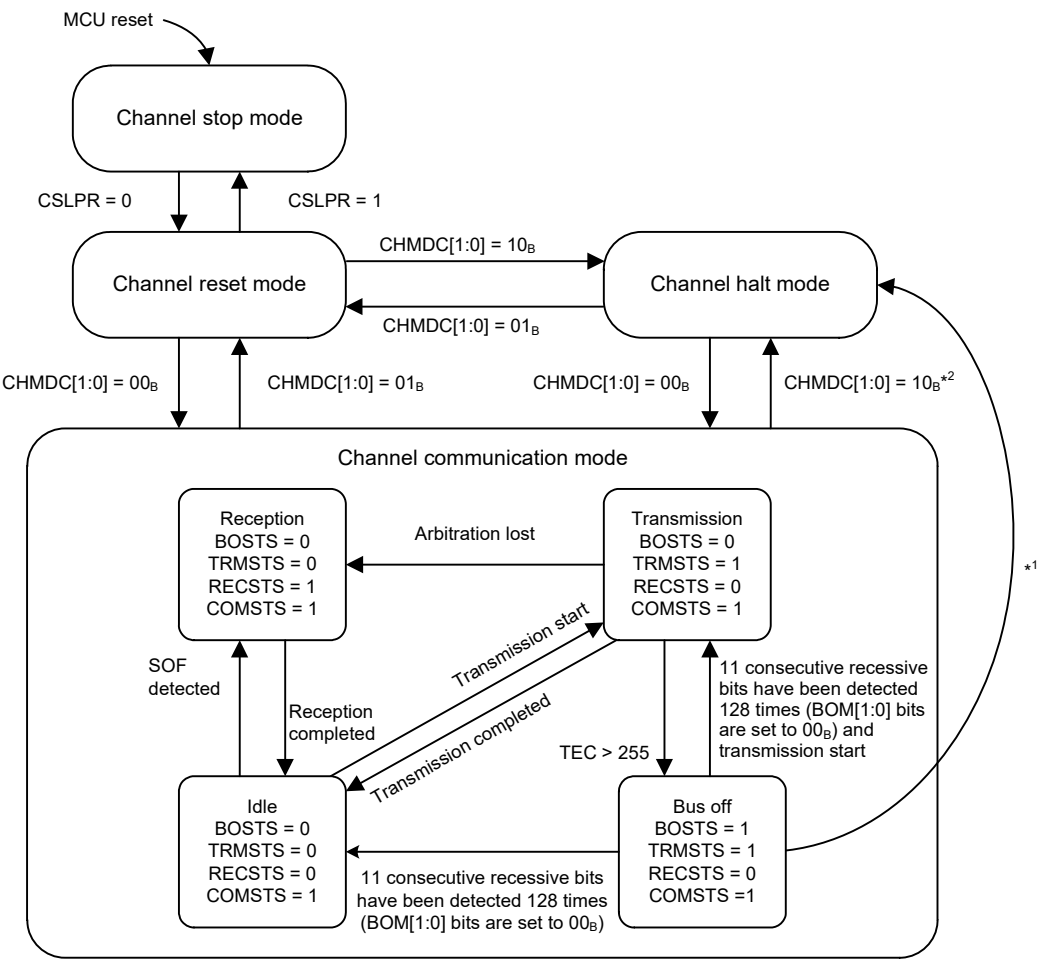
In global test mode, settings for test-related registers are performed. When the CAN module transitions to global test mode, all CAN communications are disabled.
Setting the GMDC[1:0] bits in the RSCFDn(CFD)GCTR register to 10_B sets the CHMDC[1:0] bits in each of the RSCFDn(CFD)CmCTR register to 10_B (channel halt mode). If all channels are forced to transition to channel halt mode, the CAN module transitions to global test mode. Channels that are in channel stop mode, channel reset mode, or channel halt mode do not transition.

25.6.1.4 Global Operating Mode

The RS-CANFD module operates in global operating mode.
When the GMDC[1:0] bits in the RSCFDn(CFD)GCTR register are set to 00_B, the RS-CANFD module transitions to global operating mode.

25.6.2 Channel Modes

Figure 25.5 shows a channel mode state transition chart. Table 25.174 shows the channel mode transition time.



Note: CHMDC[1:0], CSLPR, BOM[1:0]: Bits in the RSCFDn(CFD)CmCTR register (m = 0 to 1)
BOSTS, TRMSTS, RECSTS, COMSTS: Bits in the RSCFDn(CFD)CmSTS register

Note 1. Timing of transition from bus off state to channel halt mode
- When BOM[1:0] = 01_B: Transition to channel halt mode when TEC exceeds 255
- When BOM[1:0] = 10_B: Transition to channel halt mode when 11 consecutive recessive bits have been detected 128 times
- When BOM[1:0] = 11_B: Transition to channel halt mode when the CHMDC[1:0] bits are set to 10_B

Note 2. While the CAN bus is locked at the dominant level, transition to channel halt mode is not made. In that case, enter channel reset mode.

Figure 25.5 Channel Mode State Transition Chart

Table 25.174 Channel Mode Transition Time

Mode before Transition	Mode after Transition	Maximum Transition Time
Channel stop	Channel reset	Three pclk cycles
Channel reset	Channel stop	Three pclk cycles
Channel reset	Channel halt	Three CANm bit times*1
Channel reset	Channel communication	Four CANm bit times*1
Channel halt	Channel reset	Two CANm bit times*1
Channel halt	Channel communication	Four CANm bit times*1
Channel communication	Channel reset	Two CANm bit times*1
Channel communication	Channel halt	Two CANm frames

Note 1. In CANFD mode, this time value is the CANm bit time of the nominal bit rate.

25.6.2.1 Channel Stop Mode

In channel stop mode, clocks are not supplied to channels and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained.
Each channel enters channel stop mode after the MCU is reset. Channels also transition to channel stop mode when the GSLPR bit in the RSCFDn(CFD)CmCTR register (m = 0, 1) is set to 1 (channel stop mode) in channel reset mode. The GSLPR bit should not be modified in channel communication mode and channel halt mode.

25.6.2.2 Channel Reset Mode

In channel reset mode, channel settings are performed. When a channel transitions to channel reset mode, some channel-related registers are initialized. For registers to be initialized, see Table 25.176, Registers Initialized in Global Reset Mode or Channel Reset Mode.
When the CHMDC[1:0] bits in the RSCFDn(CFD)CmCTR register are set to 01_B (channel reset mode) during CAN communication, communication is terminated before it is completed and the channel transitions to channel reset mode. Table 25.175 shows the operation when the CHMDC[1:0] bits are set to 01_B (channel reset mode) during CAN communication.

25.6.2.3 Channel Halt Mode

In channel halt mode, settings for test-related registers of channels are performed. When a channel transitions to channel halt mode, CAN communication of the channel stops.
Table 25.175 shows operation when the CHMDC[1:0] bits are set to 10_B (channel halt mode) during CAN communication.

Table 25.175 Operation in Transitions to Channel Reset Mode/Channel Halt Mode

Mode	During Reception	During Transmission	Bus Off State
Channel reset (CHMDC[1:0] = 01_B)	Transitions to channel reset mode before reception is completed.*1	Transitions to channel reset mode before transmission is completed.*1	Transitions to channel reset mode before bus off recovery.
Channel halt*3 (CHMDC[1:0] = 10_B)	Transitions to channel halt mode after reception is completed.*2	Transitions to channel halt mode after transmission is completed.	[When BOM[1:0] = 00_B] Transitions to channel halt mode (CHMDC[1:0] = 10_B) only after bus off recovery. [When BOM[1:0] = 01_B] Transitions to channel halt mode automatically when the condition for transition to bus off state is met. [When BOM[1:0] = 10_B] Transitions to channel halt mode automatically after bus off recovery. [When BOM[1:0] = 11_B] Transitions to channel halt mode immediately after the CHMDC[1:0] bits are set to 10_B before bus off recovery.

Note 1. To allow transition to channel reset mode after communication is completed, set the CHMDC[1:0] bits to 10_B and confirm that communication has been completed and transition to channel halt mode has been made, and then set the CHMDC[1:0] bits to 01_B.
Note 2. While the CAN bus is locked at the dominant level, transition to channel halt mode is not made. In that case, enter channel reset mode. The CAN bus status can be confirmed with the BLF flag of the RSCFDn(CFD)CmERFL register that becomes 1 when dominant lock is detected.
Note 3. In classical CAN mode, when the transition from channel reset mode to channel halt mode is to be made, set the RSCANnCmCFG register in channel reset mode and then shift to channel halt mode. In CANFD mode, set the RSCFDnCFDCmNCFG register and the RSCFDnCFDCmDCFG register, and then make a transition.

25.6.2.4 Channel Communication Mode

In channel communication mode, CAN communication is performed. Each channel has the following communication states during CAN communication.

- Idle: Neither reception nor transmission is in progress.
- Reception: Receiving a message sent from another node.
- Transmission: Transmitting a message.
- Bus off: Isolated from CAN communication.

When the CHMDC[1:0] bits in the RSCFDn(CFD)CmCTR register are set to 00_B, the channel transitions to channel communication mode. After that, once 11 consecutive recessive bits have been detected, the COMSTS flag in the RSCFDn(CFD)CmSTS register (m = 0, 1) is set to 1 (communication is ready) and transmission and reception are enabled on the CAN network as an active node. At this time, transmission and reception of messages can be started.

25.6.2.5 Bus Off State

A channel transitions to the bus off state according to the transmit/receive error counter increment/decrement rules of the CAN specifications.
The conditions for returning from the bus off state are determined by the BOM[1:0] bits in the RSCFDn(CFD)CmCTR register.

- When BOM[1:0] = 00_B:
  Bus off recovery is compliant with the CAN specifications. After 11 consecutive recessive bits have been detected 128 times, a channel returns from the bus off state to the CAN communication ready state (error active state). At that time, the TEC[7:0] and REC[7:0] bits in the RSCFDn(CFD)CmSTS register are initialized to 00_H, the BORF flag in the RSCFDn(CFD)CmERFL register is set to 1 (bus off recovery is detected), and a bus off recovery interrupt request is generated. When the CHMDC[1:0] bits in the RSCFDn(CFD)CmCTR register are set to 10_B (channel halt mode) in the bus off state, the channel transitions to channel halt mode after bus off recovery has been completed (11 consecutive recessive bits have been detected 128 times).
- When BOM[1:0] = 01_B:
  When a channel transitions to the bus off state, the CHMDC[1:0] bits are set to 10_B and the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00_H. The BORF flag is not set to 1, and bus off recovery interrupt request is not generated.
- When BOM[1:0] = 10_B:
  When a channel has transitioned to the bus off state, the CHMDC[1:0] bits are set to 10_B. After bus off recovery has been completed (11 consecutive recessive bits have been detected 128 times), the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00_H, the BORF flag is set to 1, and a bus off recovery interrupt request is generated.
- When BOM[1:0] = 11_B:
  When the CHMDC[1:0] bits are set to 10_B in the bus off state, the channel transitions to channel halt mode before bus off recovery is completed. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00_H, but the BORF flag is not set to 1. Also, a bus off recovery interrupt is not generated.
  However, the BORF flag becomes 1 and a bus off recovery interrupt request is generated if a CAN module transitions to error active state (by detecting 128 times of 11 consecutive recessive bits) before CHMDC[1:0] bits are set to 10_B.

If the RS-CANFD module causes the channel to transition to channel halt mode simultaneously with a program write to the CHMDC[1:0] bits, the program write takes precedence. An automatic transition to channel halt mode when the BOM[1:0] bits are set to 01_B or 10_B is made only when the CHMDC[1:0] bits are 00_B (channel communication mode). Furthermore, setting the RTBO bit in the RSCFDn(CFD)CmCTR register to 1 allows a forced return from the bus off state. As soon as the RTBO bit is set to 1, the state changes to the error active state. After 11 consecutive recessive bits have been detected, the CAN module becomes ready for communication. In this case, the BORF flag is not set to 1 and the TEC[7:0] and REC[7:0] bits are initialized to 00_H. Write 1 to the RTBO bit only when the BOM[1:0] value is 00_B. Writing the RTBO bit to 1 in a state other than the bus off state is ignored, and the RTBO bit is immediately set to 0.

25.6.3 Initializing Registers by Transition to CAN Mode

Table 25.176 lists bits and flags to be initialized by a transition to channel reset mode. These bits and flags are also initialized by a transition to global reset mode. Furthermore, Table 25.177 lists bits and flags to be initialized only by a transition to global reset mode.

Table 25.176 Registers Initialized in Global Reset Mode or Channel Reset Mode

Register	Bit/Flag
RSCFDn(CFD)CmCTR register	(ROM), CRCT, CTMS[1:0], CTME, CHMDC[1:0]
RSCFDn(CFD)CmSTS register	CHLTSTS, EPSTS, BOSTS, TRMSTS, RECSTS, COMSTS, (ESIF), REC[7:0], TEC[7:0]
RSCFDn(CFD)CmERFL register	CRCREG[14:0], ADERR, B0ERR, B1ERR, CERR, AERR, FERR, SERR, ALF, BLF, OVLF, BORF, BOEF, EPF, EWF, BEF
RSCFDnCFDCmFDCTR register	EOCCLR, SOCCLR
RSCFDnCFDCmFDSTS register	SOC[7:0], EOC[7:0], SOCO, EOCO, TDCVF, TDCR[6:0]
RSCFDnCFDCmFDCRC register	CRCREG[20:0],SCNT[3:0]
RSCFDn(CFD)CFCCk register	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFE
RSCFDn(CFD)CFSTSk register	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFMC[7:0], CFFLL, CFEMP, CFMLT, CFRXIF, CFTXIF
RSCFDn(CFD)CFTISTS register	CFkTXIF
RSCFDn(CFD)TMCp register	TMOM, TMTAR, TMTR
RSCFDn(CFD)TMSTSp register	TMTARM, TMTRM, TMTRF[1:0], TMTSTS
RSCFDn(CFD)TMTRSTSy register	TMTRSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCFDn(CFD)TMTARSTSy register	TMTARSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCFDn(CFD)TMTCSTSy register	TMTCSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCFDn(CFD)TMTASTSy register	TMTASTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCFDn(CFD)TXQCCm register	TXQE
RSCFDn(CFD)TXQSTSm register	TXQIF, TXQFLL, TXQEMP
RSCFDn(CFD)THLCCm register	THLE
RSCFDn(CFD)THLSTSm register	THLMC[4:0], THLIF, THLELT, THLFLL, THLEMP
RSCFDn(CFD)GTINTSTS0 register	TSIFm, TAIFm, TQIFm, CFTIFm, THIFm (m =0, 1)

Note: Bits and flags in parentheses exist only in registers for use in CANFD mode.

Table 25.177 Registers Initialized Only in Global Reset Mode

Register	Bit/Flag
RSCFDn(CFD)GSTS register	GHLTSTS
RSCFDn(CFD)GERFL register	EEF0, EEF1, (CMPOF), THLES, MES, DEF
RSCFDn(CFD)GTSC register	TS[15:0]
RSCFDn(CFD)RMNDy register	RMNSq
RSCFDn(CFD)RFCCx register	RFE
RSCFDn(CFD)RFSTSx register	RFMC[7:0], RFIF, RFMLT, RFFLL, RFEMP
RSCFDn(CFD)CFCCk register	When transmit/receive FIFO buffer is in receive mode: CFE
RSCFDn(CFD)CFSTSk register	When transmit/receive FIFO buffer is in receive mode: CFMC[7:0], CFFLL, CFEMP, CFTXIF, CFRXIF, CFMLT
RSCFDn(CFD)FESTS register	CFkEMP, RFxEMP
RSCFDn(CFD)FFSTS register	CFkFLL, RFxFLL
RSCFDn(CFD)FMSTS register	CFkMLT, RFxMLT
RSCFDn(CFD)RFISTS register	RFxIF
RSCFDn(CFD)CFRISTS register	CFkRXIF
RSCFDnCFDCDTCT register	CFDMAEm, RFDMAEx
RSCFDnCFDCDTSTS register	CFDMASTSm, RFDMASTSx
RSCFDn(CFD)GTSTCFG register	RTMPS[6:0], C0ICBCE, C1ICBCE
RSCFDn(CFD)GTSTCTR register	RTME, ICBCTME

Note: Bits and flags in parentheses exist only in registers for use in CANFD mode.

25.7 Reception Function

There are two reception types.

- Reception by receive buffers:
  Zero to 32 receive buffers can be shared by all channels. Since messages stored in receive buffers are overwritten at each reception, the latest receive data can always be read.
- Reception by receive FIFO buffers and transmit/receive FIFO buffers (receive mode):
  Eight receive FIFO buffers can be shared by all channels and three dedicated transmit/receive FIFO buffers are provided for each channel. Messages of up to the number of buffer stages specified with the RFDC[2:0] and CFDC[2:0] bits can be stored in FIFO buffers and can be read sequentially from the oldest.

25.7.1 Data Processing Using the Receive Rule Table

Data processing using the receive rule table allows dispatching of selected messages to the specified buffer. Data processing includes acceptance filter processing, DLC filter processing, routing processing, label addition processing, and mirror function processing.
Up to 128 receive rules can be registered per channel and up to (64 × number of channels) total receive rules can be registered in the entire module. (Up to 128 receive rules can be registered in this module that has two channels.) Set receive rules for each channel. Receive rules cannot be shared with other channels. If receive rules are not set, no messages can be received. Figure 25.6 illustrates how receive rules are registered.

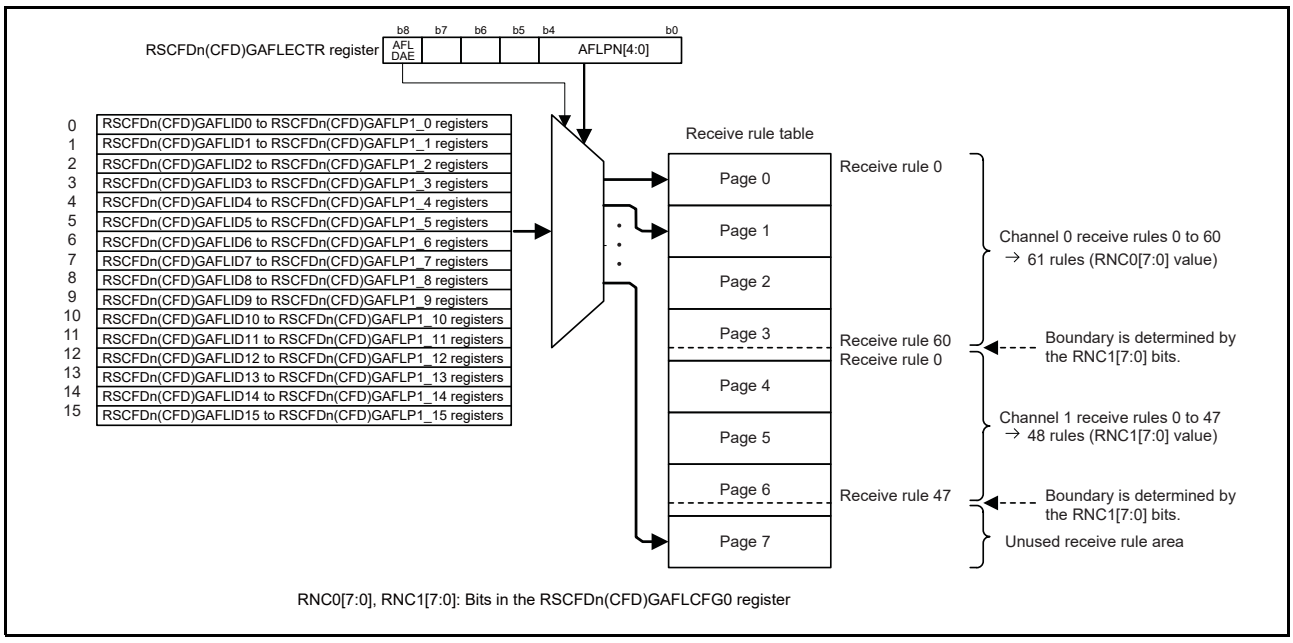


Figure 25.6 Entry of Receive Rules (for Setting Channels 0 and 1)

Note: Receive rules for each channel must be set in contiguous blocks.
Channel 1 rules and channel 0 rules must be set separately.

Each receive rule consists of 16 bytes in the RSCFDn(CFD)GAFLIDj, RSCFDn(CFD)GAFLMj, RSCFDn(CFD)GAFLP0_j, and RSCFDn(CFD)GAFLP1_j registers (j = 0 to 15). The RSCFDn(CFD)GAFLIDj register is used to set GAFLID, GAFLIDE bit, GAFLRTR bit, and the mirror function, the RSCFDn(CFD)GAFLMj register is used to set mask, the RSCFDn(CFD)GAFLP0_j register is used to set label information to be added, DLC value, and storage receive buffer, and the RSCFDn(CFD)GAFLP1_j register is used to set storage FIFO buffer. Up to 16 receive rules can be set per page.

25.7.1.1 Acceptance Filter Processing

In the acceptance filter processing, the ID data, IDE bit, and RTR bit in a received message are compared with the ID data, IDE bit, and RTR bit set in the receive rule of the corresponding channel. When all these bits match, the message passes through the acceptance filter processing. The ID data, IDE bit, and RTR bit in the received message which correspond to the bits set to 0 (bits are not compared) in the RSCFDn(CFD)GAFLMj register are not compared and are regarded as matched.

Check begins with the receive rule of the minimum number for the corresponding channel. When all the bits to be compared in a received message match the bits set in the receive rule or when all the receive rules are compared without any match, filter processing stops. If there is no matching receive rule, the received message is not stored in the receive buffer or FIFO buffer.

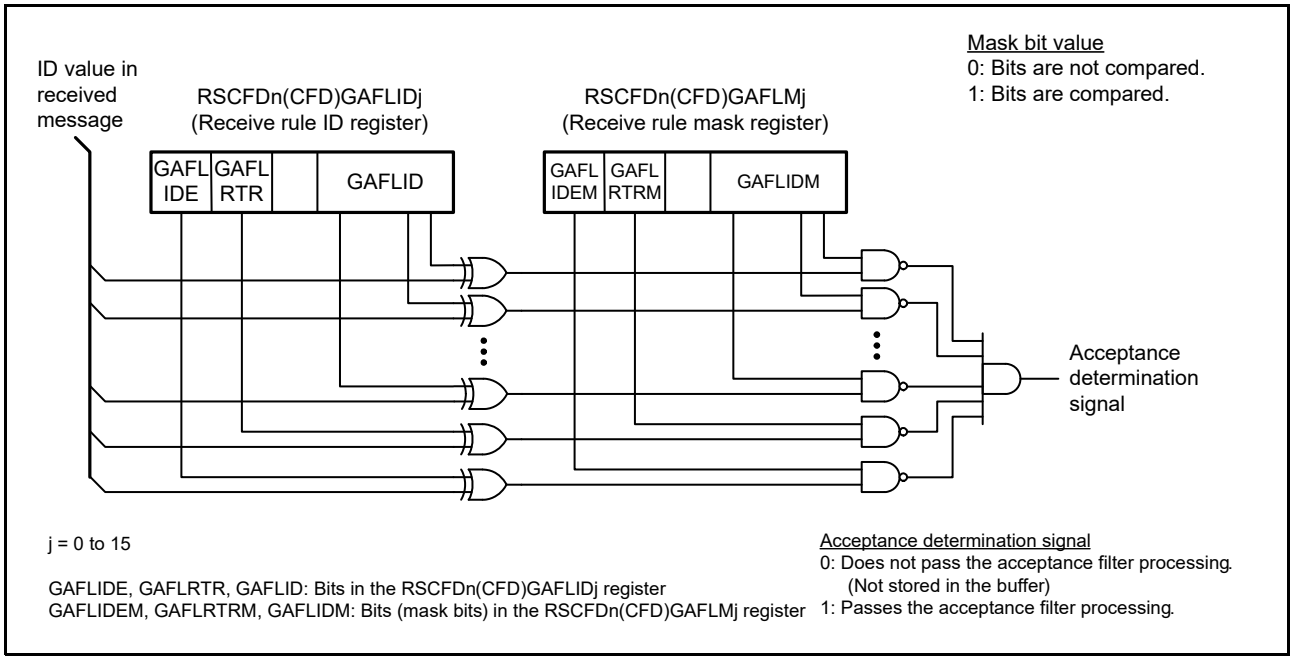


Figure 25.7 Acceptance Filter Function

25.7.1.2 DLC Filter Processing

When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), DLC filter processing is added to messages that passed through the acceptance filter processing. When the DLC value in a message is equal to or larger than the DLC value set in the receive rule, the message passes through the DLC filter processing.

When a message has passed through the DLC filter processing with the DRE bit in the RSCFDn(CFD)GCFG register set to 0 (DLC replacement is disabled), the DLC value in the received message is stored in the buffer. In this case, all the data bytes in the received message are stored in the buffer.

When a message has passed through the DLC filter processing with the DRE bit in the RSCFDn(CFD)GCFG register set to 1 (DLC replacement is enabled), the DLC value in the receive rule is stored in the buffer instead of the DLC value in the received message. In this case, a value of 00_H is stored in each data byte beyond the number of bytes which is indicated by the DLC value in the receive rule.

When the DLC value in the received message is smaller than that in the receive rule, the message does not pass through the DLC filter processing. In this case, the message is not stored in the receive buffer or the FIFO buffer and the DEF flag in the RSCFDn(CFD)GERFL register is set to 1 (a DLC error is present).

25.7.1.3 Routing Processing

Messages that passed through the acceptance filter processing and the DLC filter processing are stored in receive buffers, receive FIFO buffers, or transmit/receive FIFO buffers (set to receive mode or gateway mode). Message storage destination is set by the GAFLRMV and GAFLRMDP[6:0] bits in the RSCFDn(CFD)GAFLP0_j register (j = 0 to 15) and by the RSCFDn(CFD)GAFLP1_j register. Messages that passed through the acceptance filter processing and the DLC filter processing can be stored in up to eight buffers.
In CANFD mode, if the payload length of the received message exceeds the payload storage size of the storage buffer, the CMPOF flag in the RSCFDnCFDGERFL register is set to 1 (payload overflow) and the processing is handled according to the CMPOC bit in the RSCFDnCFDGCFG register. When the CMPOC bit is 0, the received message which exceeds the payload storage size is not stored in the buffer. When the CMPOC bit is 1, the received message is stored in the buffer with payloads exceeding the storage size being discarded.

25.7.1.4 Label Addition Processing

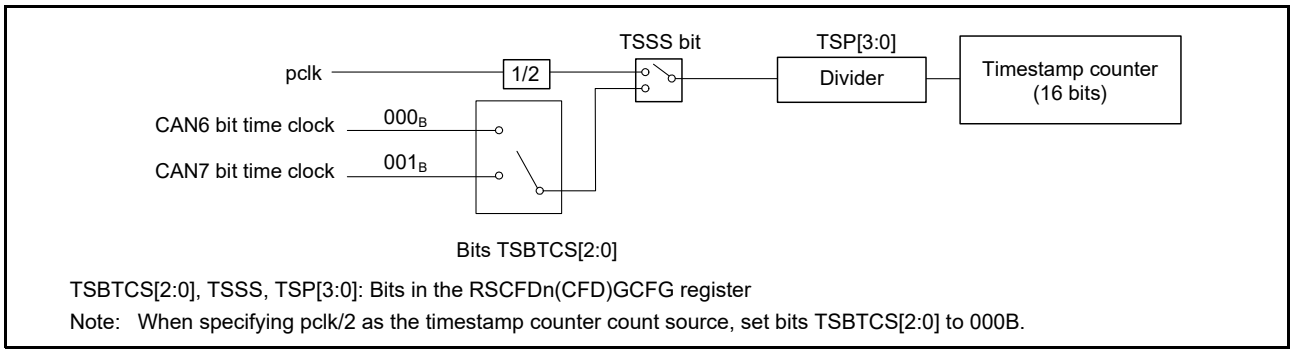
It is possible to add 12-bit label information to messages that passed through the filter processing and store them in buffers. This label information is set in the GAFLPTR[11:0] bits in the RSCFDn(CFD)GAFLP0_j register.

25.7.1.5 Mirror Function Processing

The mirror function allows the CAN node to receive its own transmitted messages. The mirror function is made available by setting the MME bit in the RSCFDn(CFD)GCFG register to 1 (mirror function is enabled).
When the mirror function is in use, receive rules for which the GAFLLB bit in the RSCFDn(CFD)GAFLIDj register is set to 0 are used for data processing when receiving messages transmitted from other CAN nodes. When the CAN node is receiving its own transmitted messages, receive rules for which the GAFLLB bit is set to 1 are used for data processing.

25.7.1.6 Timestamp

The timestamp counter is a 16-bit free-running counter used for recording message receive time and transmission time. The timestamp counter value is fetched at the timing set with the TSCCFG[1:0] bits in the RSCFDn(CFD)GCFG register and is then stored in a receive buffer or a FIFO buffer together with the message ID and data during data reception. The clock source of the timestamp counter is selected by the TSBTCS[2:0] and TSSS bits in the RSCFDn(CFD)GCFG register. In classical CAN mode, either pclk/2 or the CANm bit time clock (m = 0, 1). In CANFD mode, the clock source is selectable from pclk/2 or nominal CANm bit time clock. However, do not select the nominal CANm bit time clock of channels that handle CANFD frames. The timestamp counter count source is obtained by dividing the selected clock source by the TSP[3:0] value in the RSCFDn(CFD)GCFG register.
When the CANm bit time clock or nominal CANm bit time clock is used as a clock source, the timestamp counter stops when the corresponding channel transitions to channel reset mode or channel halt mode. When the pclk/2 is used as a clock source, the timestamp function is not affected by channel mode.
The timestamp counter value is reset to 0000_H by setting the TSRST bit in the RSCFDn(CFD)GCTR register to 1.



TSBTCS[2:0], TSSS, TSP[3:0]: Bits in the RSCFDn(CFD)GCFG register

Note: When specifying pclk/2 as the timestamp counter count source, set bits TSBTCS[2:0] to 000B.

Figure 25.8 Timestamp Function Block Diagram

25.8 Transmission Functions

There are three types of transmission. In classical CAN mode, transmittable payload length is 8 bytes in every transmission type. In CANFD mode, transmittable payload length varies with transmission types.

- Transmission using transmit buffers:
  Each channel has 16 buffers. Transmittable payload length in CANFD mode is 20 bytes. However, when transmit buffer merge mode is used, four buffers out of 16 buffers are allocated as a payload-only storage area and two buffers are able to transmit payloads with a length of more than 20 bytes.
- Transmission using transmit/receive FIFO buffers (transmit mode):
  Each channel has three FIFO buffers. Up to 128 messages can be contained in a single FIFO buffer. Transmittable payload length in CANFD mode is 64 bytes. Each FIFO buffer is used with a link to a transmit buffer. Only the message to be transmitted next in a FIFO buffer becomes the target of transmit priority determination. Messages are transmitted sequentially on a first-in, first-out basis.
- Transmission using transmit queues:
  Up to 16 transmit buffers per channel can be allocated to the transmit queues. Transmittable payload length in CANFD mode is 20 bytes. Transmit buffer ((16 × m) + 15) is used as an access window of a corresponding channel. Transmit buffers are allocated to transmit queues in descending order of buffer number. All messages in transmit queues, which are targets of priority determination, are transmitted in the order of ID number.

Figure 25.9 shows the allocation of transmit queues and transmit/receive FIFO buffer link.

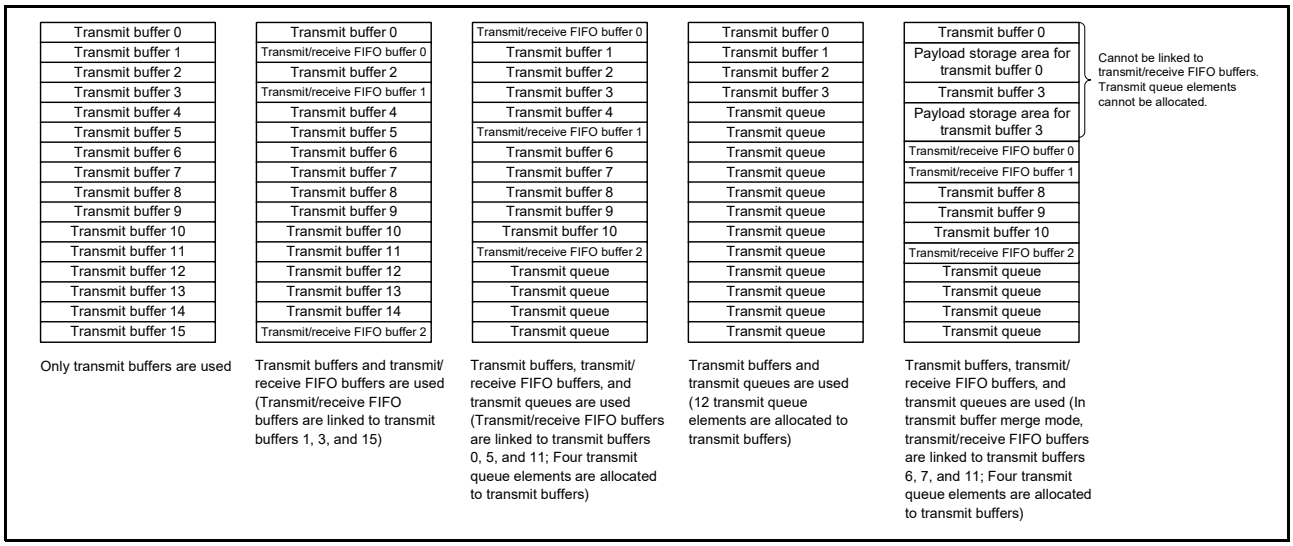


Figure 25.9 Allocation of Transmit Queues and Transmit/Receive FIFO Buffer Links

25.8.1 Transmit Priority Determination

If transmit requests are issued from multiple buffers or from the queue on the same channel, transmit priority is determined using one of the following methods.
The priority is determined by using one of the following methods.

- ID priority (TPRI bit = 0)
- Transmit buffer number priority (TPRI bit = 1)

All CAN channels use the setting of the TPRI bit in the RSCFDn(CFD)GCFG register.
When the TPRI bit is set to 0, messages are transmitted according to the priority of stored message IDs. ID priority conforms to the CAN bus arbitration specification defined in the CAN specifications. All IDs of pending transmit messages are targets of priority determination, regardless of whether they are stored in transmit buffers, transmit/receive FIFO buffers (set to transmit mode or gateway mode), or the transmit queue. If even a single transmit queue is used, select ID priority. When transmit/receive FIFO buffers are used, the oldest message in a FIFO buffer becomes the target of priority determination. When a message is being transmitted from a transmit/receive FIFO buffer, the next message in the FIFO buffer becomes the target of priority determination. When a transmit queue is used, all messages in the transmit queue are targets of priority determination. If the same ID is set for two or more buffers, the buffer with the smaller buffer number takes precedence.
When the TPRI bit is set to 1, the message in the transmit buffer with the minimum buffer number among all buffers with a transmit request is transmitted first. When transmit/receive FIFO buffers are linked to transmit buffers, transmit priority is determined according to linked transmit buffer numbers.
When messages are retransmitted due to an arbitration-lost or an error, transmit priority determination is made again regardless of the TPRI bit.

25.8.2 Transmission Using Transmit Buffers

Setting the transmit request bit (TMTR bit in the RSCFDn(CFD)TMCp register) in a transmit buffer to 1 (transmission is requested) allows transmission of data frames or remote frames.
The transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCFDn(CFD)TMSTSp register (p = 0 to 31). When transmit completes successfully, the TMTRF[1:0] flag is set to 10_B (transmission has been completed (without transmit abort request)) or 11_B (transmission has been completed (with transmit abort request)).

25.8.2.1 Transmit Abort Function

With respect to transmit buffers for which the TMTRM bit in the RSCFDn(CFD)TMSTSp register is set to 1 (a transmit request is present), when the TMTAR bit in the RSCFDn(CFD)TMCp register is set to 1 (transmit abort is requested), the transmit request is canceled. When transmit abort is completed, the TMTRF[1:0] flag in the RSCFDn(CFD)TMSTSp register is set to 01_B (transmit abort has been completed) and the transmit request is canceled (clearing the TMTRM bit to 0).
A message that is being transmitted or a message to be transmitted next according to the transmit priority determination cannot be aborted. However, when an arbitration-lost or an error occurs during transmission of a message for which the TMTAR bit is set to 1, retransmission is not performed.

25.8.2.2 One-Shot Transmission Function (Retransmission Disabling Function)

When the TMOM bit in the RSCFDn(CFD)TMCp register is set to 1 (one-shot transmission is enabled), transmission is performed only once. Even if an arbitration-lost or an error occurs, retransmission is not performed.
The one-shot transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCFDn(CFD)TMSTSp register. When one-shot transmission completes successfully, the TMTRF[1:0] flag is set to 10B or 11B. When an arbitration-lost or an error occurs, the TMTRF[1:0] flag is set to 01B (transmit abort has been completed).

25.8.2.3 Transmit Buffer Merge Mode (Only in CANFD Mode)

Transmit buffers can transmit messages with a payload length of 20 bytes, but can transmit messages with a payload length of up to 64 bytes by merging three transmit buffers in transmit buffer merge mode.
Setting the TMME bit to 1 in the RSCFDnCFDCmFDCFG register enables transmit buffer merge mode. In this mode, six buffers per channel become a merge area and two sets of transmit buffers (16 × m) + 0 to (16 × m) + 2 and transmit buffers (16 × m) + 3 to (16 × m) + 5 are merged. A transmission request is made by the first transmit buffer, and subsequent two buffers are used as a payload storage area. Do not set the transmission request bit (TMTR bit in the RSCFDnCFDTMCp register) and the transmission abort request bit (TMTAR bit in the RSCFDnCFDTMCp register) to 1 for transmit buffers except for the first buffer.
While transmit buffer merge mode is enabled, do not link the transmit/receive FIFO buffer to the merged buffers or allocate it to the transmit queue.

25.8.3 Transmission Using FIFO Buffers

Multiple messages can be stored in a single transmit/receive FIFO buffers, up to the number specified by the FIFO buffer depth, which is set by the CFDC[2:0] bits in the RSCFDn(CFD)CFCCk register (k = 0 to 5). Messages are transmitted sequentially on a first-in, first-out basis.
Each transmit/receive FIFO buffer is linked to a transmit buffer selected by the CFTML[3:0] bits in the RSCFDn(CFD)CFCCk register. When the CFE bit in the RSCFDn(CFD)CFCCk register is set to 1 (transmit/receive FIFO buffers are used), transmit/receive FIFO buffers become targets of transmit priority determination. Priority of only the next transmit message is determined in the FIFO buffer.
When the CFE bit is set to 0 (no transmit/receive FIFO buffer is used), the CFEMP flag is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) at the timing below.

- The transmit/receive FIFO buffer becomes empty immediately if the message in it is not being transmitted or is not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, arbitration-lost or the transition to channel halt mode in the case that a message in it is being transmitted or to be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages cannot be stored in FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

25.8.3.1 Interval Transmission Function

A message transmission interval time can be set to space the transmission of messages from the same FIFO buffer when using a transmit/receive FIFO buffer set to transmit mode or gateway mode.
Immediately after the first message has been transmitted successfully from the FIFO buffer with the CFE bit in the RSCFDn(CFD)CFCCk register set to 1, the interval timer starts counting (after EOF7 of the CAN protocol). After that, when the interval time has passed, the next message is transmitted. The interval timer stops in channel reset mode or by clearing the CFE bit to 0.
The interval time is set by the CFITT[7:0] bits in the RSCFDn(CFD)CFCCk register. When the interval timer is not used, set the CFITT[7:0] bits to 00_H.
Select an interval timer count source using the CFITR and CFITSS bits in the RSCFDn(CFD)CFCCk register. When the CFITR and CFITSS bits are set to 00_B, the count source is obtained by dividing pclk/2 by the value of the ITRCP[15:0] bits. When the CFITR and CFITSS bits are set to 10_B, the count source is obtained by dividing pclk/2 by (the value of the ITRCP[15:0] bits in the RSCFDn(CFD)GCFG register × 10). When the CFITR and CFITSS bits are set to $x1_B$, the CANm bit time clock becomes a count source in classical CAN mode and the nominal CANm bit time clock becomes a count source in CANFD mode. Use this count source only for the channel that does not handle the CANFD frames.
The interval time is calculated by the following equations where M is the value of ITRCP[15:0] and N is the value of CFITT[7:0].

- When CFITR and CFITSS = 00_B:

$$\frac{1}{\text{pclk frequency}} \times 2 \times M \times N$$

- When CFITR and CFITSS = 10_B:

$$\frac{1}{\text{pclk frequency}} \times 2 \times M \times 10 \times N$$

- When CFITR and CFITSS = $x1_B$:

$$\text{Classical CAN mode: } \frac{1}{\text{CANm bit time clock frequency}} \times N$$

$$\text{CANFD mode: } \frac{1}{\text{Nominal CANm bit time clock frequency}} \times N$$

Figure 25.10 shows the interval timer block diagram.

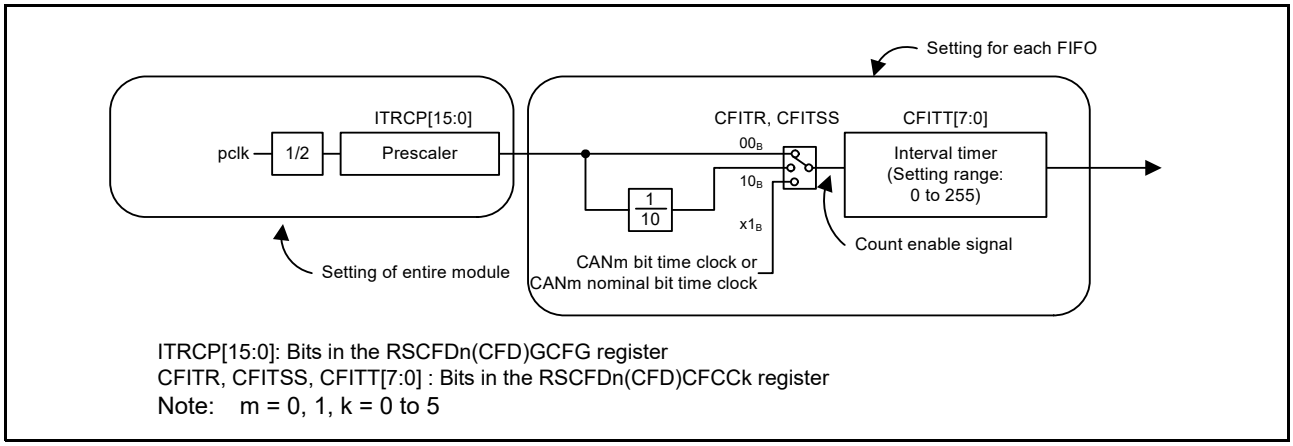


Figure 25.10 Interval Timer Block Diagram

Figure 25.11 shows the interval timer timing diagram.

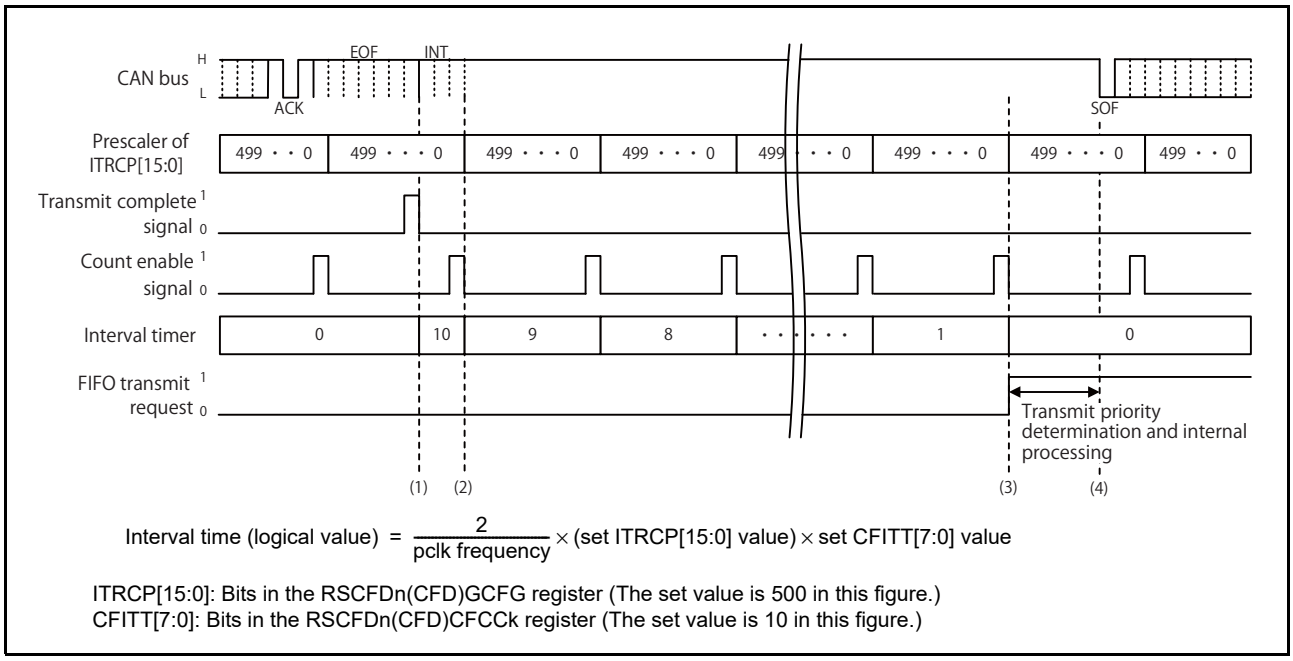


Figure 25.11 Interval Timer Timing Chart

(1) The interval timer starts counting upon completion of transmission. Since the prescaler is not initialized at the time of transmission completion, the first interval time contains an error of up to one count of the interval timer.
(2) The interval timer is decremented by the next count enable signal.
(3) When the interval timer has decreased to 0, the transmit/receive FIFO buffer issues a transmit request.
(4) The transmit/receive FIFO buffer is determined for the next transmission by the priority determination, it starts transmitting data. Transmission starts usually with a delay of three CANm bit time clock cycles or less from the issue of transmit request. If multiple internal processes (such as receive filter processing, message routing, and transmit priority determination) take place in all channels, a delay of up to 644 cycles of the pclk may be generated.

25.8.4 Transmission Using Transmit Queues

Three to sixteen buffers (up to 10 buffers in transmit buffer merge mode) are allocated to a transmit queue for each channel, and transmit buffer ((16 × m) + 15) is used as an access window of a corresponding channel.
All messages in a transmit queue are targets of transmit priority determination and are transmitted in the ID priority order regardless of storage sequence. If two messages having the same ID are stored in a transmit queue, these messages are not always transmitted in the order of their storage in the transmit queue.
Setting the TXQE bit in the RSCFDn(CFD)TXQCCm register to 0 disables transmit queues. When the TXQE bit is set to 0, the TXQEMP flag in the RSCFDn(CFD)TXQSTSm register is set to 1 (the transmit queue contains no messages (transmit queue empty)) at the timing below.

- The transmit queue becomes empty immediately when no message in it is being transmitted or will be transmitted next.
- The transmit queue becomes empty after transmission completion, CAN bus error detection, arbitration-lost, or the transition to channel halt mode when a message in it is being transmitted or will be transmitted next.

When the TXQE bit is cleared to 0, all messages in transmit queues are lost and messages cannot be stored in transmit queues. Confirm that the TXQEMP flag is set to 1 before setting the TXQE bit to 1 again.

25.8.5 Transmit Data Padding (Only in CANFD Mode)

When the payload length indicated by the set DLC value in a transmit message exceeds the payload storage area size of a buffer to be used for transmission, excessive payloads are padded by CC_H.
This processing is performed in the following cases when the transmit buffer merge mode is disabled (TMME bit in the RSCFDnCFDCmFDCFG register is 0).

- Transmit/receive FIFO set to transmission or gateway mode:
  When the payload length of the transmit DLC exceeds the transmit/receive FIFO payload storage area size set by the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register
- Transmit buffer (including transmit queue):
  When the payload length of the transmit DLC exceeds 20 bytes

When the transmit buffer merge mode is enabled, no transmit data is padded in any transmission using a transmit buffer, transmit/receive FIFO buffer, or transmit queue. At this time, do not set a payload length more than the payload storage size of the buffer for transmitting as the DLC value in the transmit message.

25.8.6 Transmit History Function

Information about transmission-completed messages can be stored in the transmit history buffer. Each channel has a single transmit history buffer that can contain 16 sets of transmit history data.
A message transmit source buffer type can be selected by the THLDTE bit in the RSCFDn(CFD)THLCCm register. The THLEN bit in the RSCFDn(CFD)CFIDk register (k = 0 to 5) determines whether transmit history data is stored for each message.
In classical CAN mode, the TMTSCE bit in the RSCANnGCFG register can be used to set whether to include a timestamp value in the transmit history data. In CANFD mode, a timestamp value is always included.
The following information on a transmitted message will be stored in the transmit history buffer after the successful completion of transmission.
Storage of the transmit history data after the successful completion of transmission may take up to 128 cycles of pclk in classical CAN mode or 396 cycles of pclk in CANFD mode.

- Buffer type 001_B: Transmit buffer
  010_B: Transmit/receive FIFO buffer
  100_B: Transmit queue
- Buffer number: Number of source transmit buffer, transmit queue, or transmit/receive FIFO buffer. This number depends on buffer types. See Table 25.178.
- Label data: Label information of the transmit message
- Timestamp: Timestamp value of the transmit message (When the TMTSCE bit is 1 in classical CAN mode)

Table 25.178 Transmit History Data Buffer Numbers

Buffer Type Buffer No.	001_B	010_B	100_B
0000_B	Transmit buffer 16 × m + 0	Buffer numbers of the transmit buffer linked to the transmit/receive FIFO buffer by the CFTML[3:0] bits in the RSCFDn(CFD)CFCCk register (k = 0 to 5)	Buffer numbers of the transmit buffer allocated to the transmit queue that performed transmission
0001_B	Transmit buffer 16 × m + 1		
0010_B	Transmit buffer 16 × m + 2		
0011_B	Transmit buffer 16 × m + 3		
0100_B	Transmit buffer 16 × m + 4		
0101_B	Transmit buffer 16 × m + 5		
0110_B	Transmit buffer 16 × m + 6		
0111_B	Transmit buffer 16 × m + 7		
1000_B	Transmit buffer 16 × m + 8		
1001_B	Transmit buffer 16 × m + 9		
1010_B	Transmit buffer 16 × m + 10		
1011_B	Transmit buffer 16 × m + 11		
1100_B	Transmit buffer 16 × m + 12		
1101_B	Transmit buffer 16 × m + 13		
1110_B	Transmit buffer 16 × m + 14		
1111_B	Transmit buffer 16 × m + 15		

Label data is used to identify each message. Unique label data can be added to each message transmitted from a transmit buffer, transmit queue, or transmit/receive FIFO buffer.
The timestamp value is fetched from the timestamp counter at the SOF (start of frame) timing of the message. For details about the timestamp counter, see section 25.7.1.6, Timestamp.
Transmit history data can be read from the RSCFDn(CFD)THLACCm register. If an attempt is made to store new transmit history data while the buffer is full, the buffer overflows and the new data is discarded.

25.9 Gateway Function

When a transmit/receive FIFO buffer is set to gateway mode, receive messages can be transmitted from an arbitrary channel without CPU intervention.
When the CFM[1:0] bits in the RSCFDn(CFD)CFCCk register are set to 10_B (gateway mode) for the transmit/receive FIFO buffer selected by the RSCFDn(CFD)GAFLP1_j register of a channel being used for transmission, messages that pass through filter processing according to the receive rule are stored in the specified transmit/receive FIFO buffer and are automatically transmitted from the buffer.
Messages stored in a transmit/receive FIFO buffer are transmitted sequentially on a first-in, first-out basis. Only the message to be transmitted next becomes the target of transmit priority determination.
Transmit/receive FIFO buffers in the gateway mode are disabled by setting the CFE bit in the RSCFDn(CFD)CFCCk register to 0 and the CFEMP flag becomes 1 according to the timing below.

- The transmit/receive FIFO buffer becomes empty immediately when the message in it is not being transmitted and will not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, or arbitration-lost when the message in it is being transmitted or will be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages can no longer be stored in transmit/receive FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

25.9.1 CAN-CANFD Gateway (Only in CANFD Mode)

When the gateway function is used in CANFD mode, a frame to be transmitted can be replaced with a classical CAN frame or a CANFD frame.
Setting the GWEN bit in the RSCFDnCFDCmFDCFG register to 1 enables the CAN-CANFD gateway. The FDF and BRS bits in the transmit frame can be selected by the GWFDF and GWBRS bits in the RSCFDnCFDCmFDCFG register.
When the DLC value of the received CAN frame is 1001_B or more and the GWFDF bit is 1 (CANFD frame), the DLC value is replaced with 1000_B.
When the CAN-CANFD gateway is enabled, do not perform routing for the following frames.

- CANFD frames with a payload length of more than 8 bytes
- Remote frames

When the CAN-CANFD gateway is enabled, the following frame should be transmitted in the channel by setting of GWFDF.

- When GWFDF bit is set to 0, only classical CAN frames should be transmitted.
- When GWFDF bit is set to 1, only CANFD frames should be transmitted.

25.10 Test Function

The test function is classified into communication tests and global tests.

- Communication tests: Performed for each channel.
  - Standard test mode
  - Listen-only mode
  - Self-test mode 0 (external loopback mode)
  - Self-test mode 1 (internal loopback mode)
  - Restricted operation mode (only in CANFD mode)
- Global tests: Performed for the entire module
  - RAM test (read/write test)
  - Inter-channel communication test [CRC error test enabled]

25.10.1 Standard Test Mode

CRC tests are enabled in standard test mode. The CRC value calculated by the RS-CANFD module based on the transmit message or receive message is stored in the register. This CRC value is stored in the CRCREG[14:0] bits in the RSCFDn(CFD)CmERFL register when the message is a classical CAN frame (CRC length = 15 bits) or in the CRCREG[20:0] bits in the RSCFDnCFDCmFDCRC register when the message is a CANFD frame (CRC length = 17 or 21 bits). Use the inter-channel communication test function for CRC error tests. For details, see section 25.10.6.1, CRC Error Test.

25.10.2 Listen-Only Mode

Listen-only mode allows reception of data frames and remote frames. Only recessive bits are transmitted on the CAN bus, and the ACK bit, overload flag, and active error flag are not transmitted.
Listen-only mode is available for detecting the communication speed.
Do not make a transmit request from any buffer or queue in listen-only mode.
Figure 25.12 shows the connection when listen-only mode is selected.

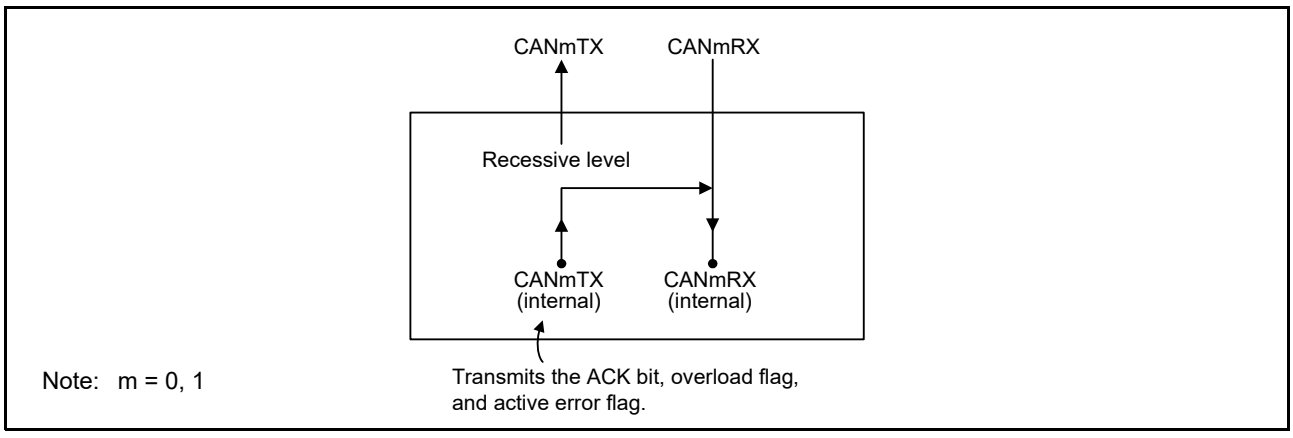


Figure 25.12 Connection when Listen-Only Mode is Selected

25.10.3 Self-Test Mode (Loopback Mode)

In self-test mode, transmitted messages are compared with the receive rule of the own channel and the messages are stored in a buffer if they have passed through the filter processing. Messages transmitted from other CAN nodes are compared only with the receive rule for which the GAFLLB bit in the RSCFDn(CFD)GAFLIDj register (j = 0 to 15) is set to 0 (when a message transmitted from another CAN node is received).
If the mirror function and self-test mode are both enabled, the self-test mode setting takes precedence.

25.10.3.1 Self-Test Mode 0 (External Loopback Mode)

Self-test mode 0 is used to perform a loopback test within a channel including the CAN transceiver.
In self-test mode 0, transmitted messages are handled as messages received through the CAN transceiver and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.
Figure 25.13 shows the connection when self-test mode 0 is selected.

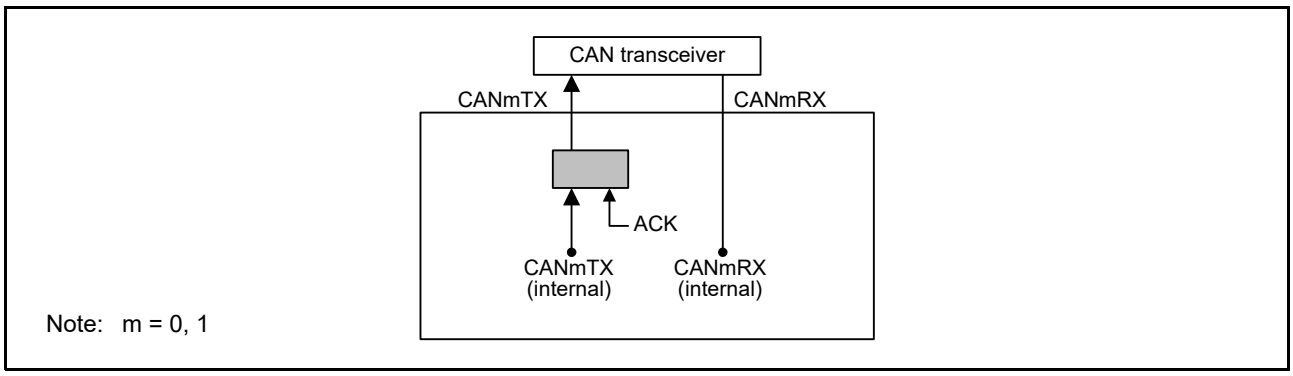


Figure 25.13 Connection when Self-Test Mode 0 is Selected

25.10.3.2 Self-Test Mode 1 (Internal Loopback Mode)

In self-test mode 1, transmitted messages are handled as received messages and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.
In self-test mode 1, internal feedback from the internal CANmTX pin (m = 0, 1) to the internal CANmRX pin is performed. The external CANmRX pin input is isolated. The external CANmTX pin outputs only recessive bits.
Figure 25.14 shows the connection when self-test mode 1 is selected.

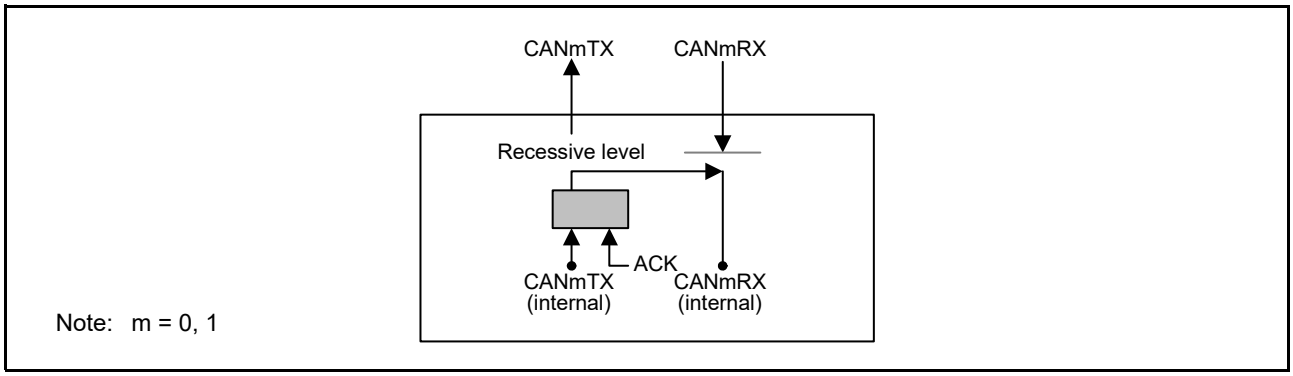


Figure 25.14 Connection when Self-Test Mode 1 is Selected

25.10.4 Restricted Operation Mode (Only in CANFD Mode)

In restricted operation mode, an ACK bit is generated when a valid data frame and a remote frame have been received, but these frames are not transmitted even if an error frame or an overload frame transmit condition is detected. When a condition is detected, operation is suspended until the bus idle state comes for resynchronization with the CAN communication. The receive error counter (REC) and the transmit error counter (TEC) do not change due to an error. A desired transmission request can be made for transmission without restrictions.

25.10.5 RAM Test

The RAM test function allows accesses to all CAN RAM addresses.
When the RAM test function is used, the RAM is divided into pages of 256 bytes each. RAM test page is set by the RTMPS[6:0] bits in the RSCFDn(CFD)GTSTCFG register. Data in the set page can be read from and written to the RSCFDn(CFD)RPGACCr register (r = 0 to 63). The available total RAM size is 5120 bytes (1400_H) in classical CAN mode or 7104 bytes ($1BC0_H$) in CANFD mode. Do not access the RAM area higher than the 192th byte on the last page (the setting of the RTMPS bit is $1B_H$) in CANFD mode.

25.10.6 Inter-Channel Communication Test

The inter-channel communication test function allows communication test by internally connecting CAN channels to each other. During this test, channels are isolated from the external CAN bus.
Before starting data transmission/reception in channel communication mode, make transmission/reception settings for each channel.
Figure 25.15 shows the connection for inter-channel communication test.

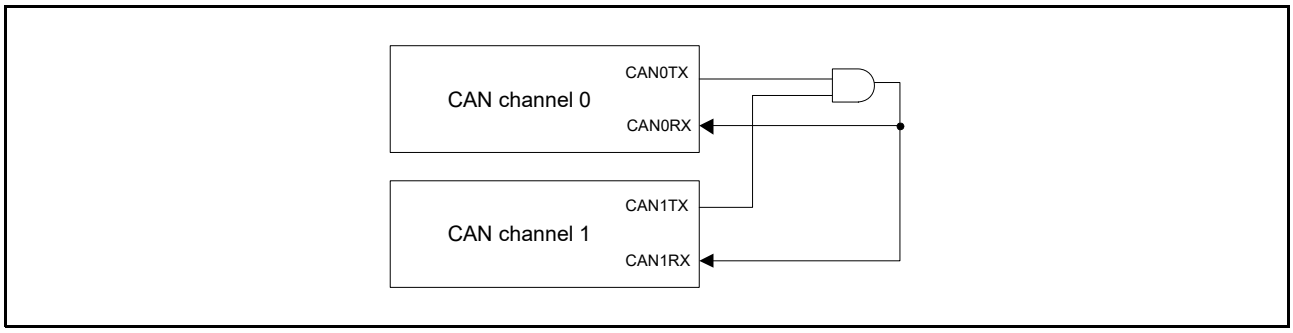


Figure 25.15 Connection for Inter-Channel Communication Test

25.10.6.1 CRC Error Test

A CRC error test is enabled during an inter-channel communication test. The following shows an example of channel 0 CRC error test procedure during a communication test between channel 0 and channel 1.

Preconditions

- Inter-channel communication test is enabled.
- Channel 0 and channel 1 are in standard test mode.

Procedure

1. Make a setting to send a message from the transmit buffer p of channel 1.
2. Set the CRCT bit in the RSCFDn(CFD)C0CTR register to 1 (to enable inversion of the first bit in the received ID field).
3. Set the TMTR bit in the RSCFDn(CFD)TMCp register to 1 (to issue a transmission request to the transmit buffer p of channel 1).
4. Wait for occurrence of a CAN0 error interrupt due to a channel bus error.
5. Read the CRCREG[14:0] bits in the RSCFDn(CFD)CmERFL register or the CRCREG[20:0] bits in the RSCFDnCFDCmFDCRC register of channel 0 and channel 1, and confirm that the CRC values are different on the transmission and the reception side.
6. Confirm that the CERR bit in RSCFDm(CFD)C0ERFL is 1 (CRC error detected).

The CRC error test function generates an incorrect CRC value by inverting the first bit in the received ID field. Therefore, note that not a CRC error but a stuff error (continuous 6-bit data of the same level) is detected when a message in which ID's upper 5-bit value is 10000B or ID's upper 6-bit value is 011111B is received.
The CRC generation circuit of the RS-CANFD module is contained in the protocol controller of each channel. Another CRC calculation test is not necessary during transmission because the same circuit is used for both transmission and reception.

25.11 RS-CANFD Setting Procedure

25.11.1 Initial Settings

The RS-CANFD module initializes the CAN RAM after the MCU is reset. The RAM initialization time is 2530 cycles of the pclk. The GRAMINIT flag in the RSCFDn(CFD)GSTS register is set to 1 (CAN RAM initialization is ongoing) during the RAM initialization and is cleared to 0 (CAN RAM initialization is finished) when the initialization is completed. Make CAN settings after the GRAMINIT flag is cleared to 0. Figure 25.16 shows the CAN setting procedure after the MCU is reset.

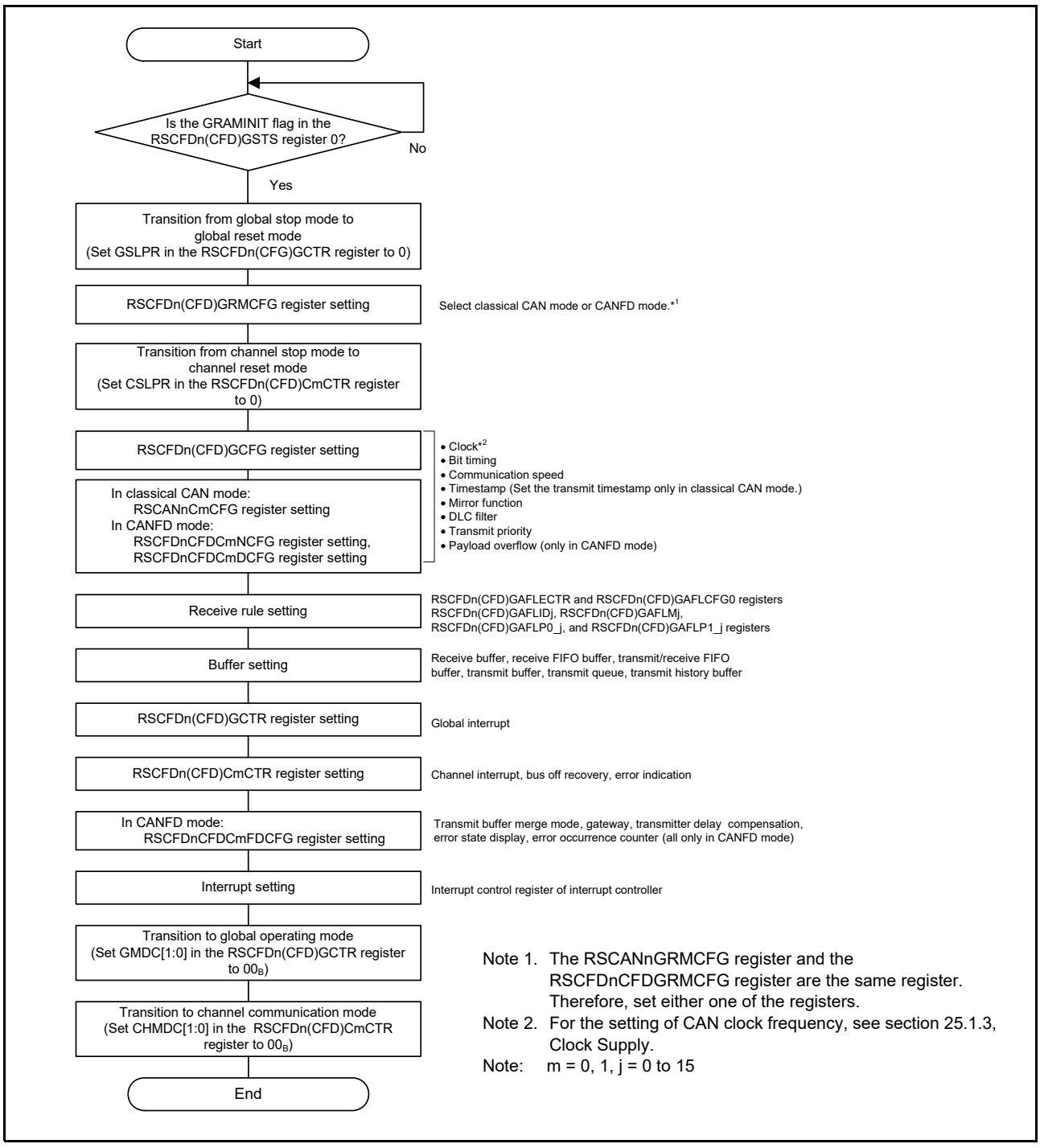


Figure 25.16 CAN Setting Procedure after the MCU is Reset

25.11.1.1 Clock Setting

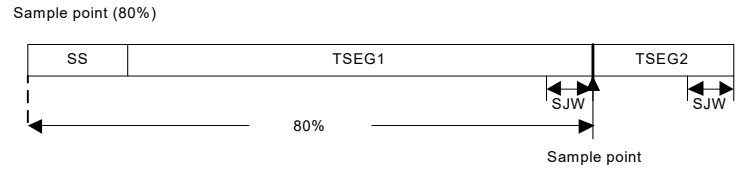
Set the CAN clock (fCAN) as a clock source of the RS-CANFD module. Select the clk_xincan or clkc using the DCS bit in the RSCFDn(CFD)GCFG register.

25.11.1.2 Bit Timing Setting

In the CAN protocol, one bit of a communication frame consists of three segments SS, TSEG1, and TSEG2, of which two segments TSEG1 and TSEG2 can be set by the corresponding registers for each channel. In classical CAN mode, set these two segments in the RSCANnCmCFG register. Two bit rates (nominal bit rate and data bit rate) are provided for CANFD mode. Set the nominal bit rate in the RSCFDnCFDCmNCFG register and set the data bit rate in the RSCFDnCFDCmDCFG register. Sample point timing can be determined by setting these two segments. This timing can be adjusted in units of 1 Time Quantum (hereafter referred to as Tq). A single Tq is the cycle of clock obtained by dividing the clock selected by the DCS bit in the RSCFDn(CFD)GCFG register. Set a division ratio by the BRP[9:0] bits in the RSCANnCmCFG register in classical CAN mode (CANmTq clock), and by the NBRP[9:0] bits in the RSCFDnCFDCmNCFG register and the DBRP[7:0] bits in the RSCFDnCFDCmDCFG register in CANFD mode (CANmTq(N) clock and CANmTq(D) clock). Be sure to specify the same values for both NBRP[9:0] and DBRP[7:0]. To specify different values for the nominal bit rate and the data bit rate, change the values of the RSCFDnCFDCmNCFG.NTSEG1 and NTSEG2 bits and RSCFDnCFDCmDCFG.DTSEG1 and DTSEG2 bits, respectively.

When the TDCE bit is set to 1 (Transmitter delay compensation is enabled) in the RSCFDnCFDCmFDCFG register, set the equal value of 1 or less to the bits NBRP[9:0] and DBRP[7:0].

Figure 25.17 shows the bit timing chart. Table 25.179 shows an example of bit timing setting.



Classical CAN mode:
- SS = 1 Tq fixed
- Set TSEG1 to a range of 4 Tq to 16 Tq
- Set TSEG2 to a range of 2 Tq to 8 Tq
- Set SJW to a range of 1 Tq to 4 Tq
- Set SS + TSEG1 + TSEG2 to a range of 8 Tq to 25 Tq
- TSEG1 > TSEG2 ≥ SJW

CANFD mode:

<Nominal bit rate>
- SS = 1 Tq fixed
- Set TSEG1 to a range of 4 Tq to 128 Tq
- Set TSEG2 to a range of 2 Tq to 32 Tq
- Set SJW to a range of 1 Tq to 32 Tq
- Set SS + TSEG1 + TSEG2 to a range of 8 Tq to 161 Tq
- TSEG1 > TSEG2 ≥ SJW

<Data bit rate>
- SS = 1 Tq fixed
- Set TSEG1 to a range of 2 Tq to 16 Tq
- Set TSEG2 to a range of 2 Tq to 8 Tq
- Set SJW to a range of 1 Tq to 8 Tq
- Set SS + TSEG1 + TSEG2 to a range of 8 Tq to 25 Tq
- TSEG1 ≥ TSEG2 ≥ SJW

- SS (synchronization segment):
  The SS is a segment that performs synchronization by monitoring the edge from recessive to dominant bits in the Interframe Space. Interframe Space consists of Intermission, Suspend Transmission, and Bus Idle. All nodes can start transmission during Bus Idle.
- TSEG1 (time segment 1):
  TSEG1 is a segment that absorbs physical delay on the CAN network. The physical delay on the CAN network is twice the total of the delay on the CAN bus, the delay in the input comparator, and the delay in the output driver.
- TSEG2 (time segment 2):
  TSEG2 is a segment that compensates phase error due to an error in frequency.
- SJW (resynchronization jump width)
  The SJW is a length to extend or reduce the time segment to compensate for an error in phase due to phase error.

Figure 25.17 Bit Timing Chart

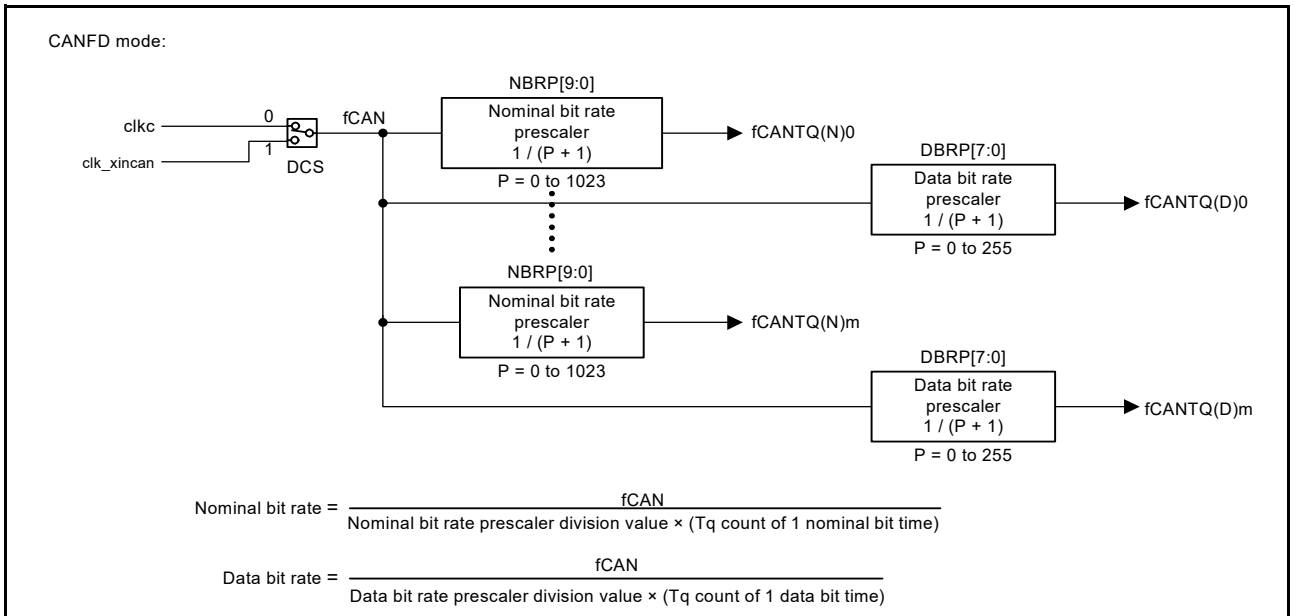
Table 25.179 Example of Bit Timing Setting

1 Bit	Set Value (Tq) SS	TSEG1	TSEG2	SJW	Sample Point (%) Note: See Figure 25.17.
	1	4	3	1	62.50
8 Tq	1	5	2	1	75.00
	1	6	3	1	70.00
10 Tq	1	7	2	1	80.00
	1	10	5	1	68.75
16 Tq	1	11	4	1	75.00
	1	12	7	1	65.00
20 Tq	1	13	6	1	70.00
50 Tq*1	1	39	10	4	80.00

Note 1. Only in CANFD mode

25.11.1.3 Communication Speed Setting

Set the CAN communication speed for each channel using the fCAN, baud rate prescaler division value, and Tq count per bit time. For CANFD mode, set two types of transmission rate (arbitration phase and data phase) for each channel. Figure 25.18 shows the CAN clock control block diagram, and Table 25.180 and Table 25.181 show examples of the communication speed setting.



Note. m = 0, 1
DCS: Bit in the RSCFDn(CFD)GCFG register
BRP[9:0]: Bits in the RSCFDnCmCFG RSCANnCmCFG register
NBRP[9:0]: Bits in the RSCFDnCFDCmNCFG register
DBRP[7:0]: Bits in the RSCFDnCFDCmDCFG register
fCAN: CAN clock
fCANTQm: CANmTq clock
fCANTQ(N)m: CANmTq(N) clock
fCANTQ(D)m: CANmTq(D) clock

Figure 25.18 CAN Clock Control Block Diagram

Table 25.180 Example of Communication Speed Setting (Classical CAN Mode)

Communication Speed \ fCAN	32 MHz	16 MHz	8 MHz
1 Mbps	8 Tq (4) 16 Tq (2)	8 Tq (2) 16 Tq (1)	8 Tq (1)
500 Kbps	8 Tq (8) 16 Tq (4)	8 Tq (4) 16 Tq (2)	8 Tq (2) 16 Tq (1)
250 Kbps	8 Tq (16) 16 Tq (8)	8 Tq (8) 16 Tq (4)	8 Tq (4) 16 Tq (2)
125 Kbps	8 Tq (32) 16 Tq (16)	8 Tq (16) 16 Tq (8)	8 Tq (8) 16 Tq (4)

Table 25.181 Example of Transmission Rate Setting (Nominal Bit Rate and Data Bit Rate in CANFD Mode)

Communication Speed \ fCAN	32 MHz	16 MHz
Nominal bit rate 1 Mbps Data bit rate 4 Mbps	Nominal bit rate 32 Tq (1) Data bit rate 8 Tq (1)	None
Nominal bit rate 500 Kbps Data bit rate 2 Mbps	Nominal bit rate 64 Tq (1) Data bit rate 16 Tq (1)	Nominal bit rate 32 Tq (1) Data bit rate 8 Tq (1)

Note: Values in ( ) are baud rate prescaler division values.

25.11.1.4 Receive Rule Setting

Receive rules can be set using receive rule-related registers.
Up to 16 receive rules can be registered per page. Specify pages 0 to 23 (for 6-channel unit) by the AFLPN[4:0] bits in the RSCFDn(CFD)GAFLECTR register. Set receive rule table write enable/disable using the AFLDAE bit.
Figure 25.19 shows the receive rule setting procedure.

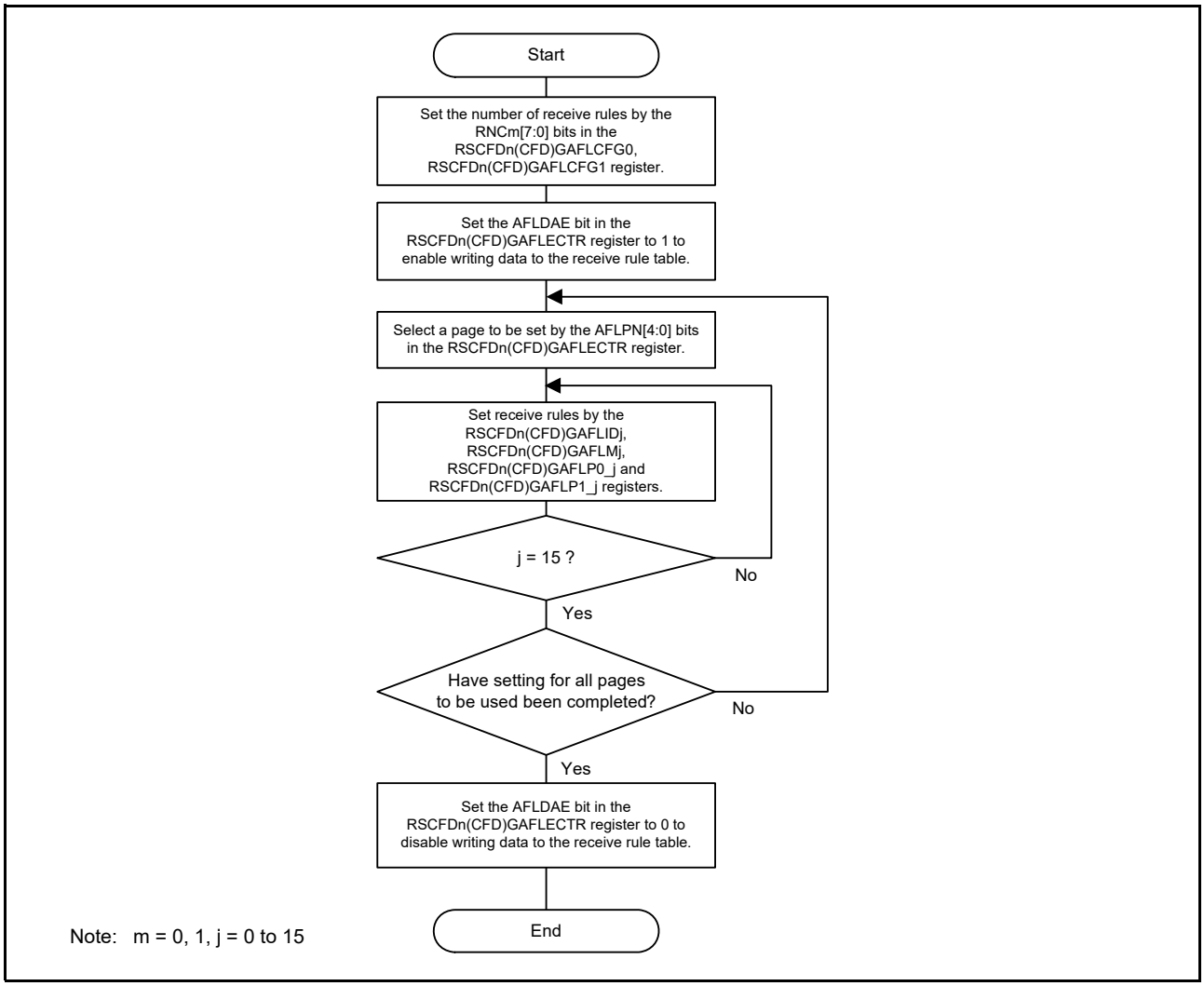


Figure 25.19 Receive Rule Setting Procedure

25.11.1.5 Buffer Setting

Set the number of buffers to be used (number of messages to be stored) and interrupt sources. Also set the payload storage size for CANFD mode. For transmit/receive FIFO buffers that are set to transmit mode, set transmit buffers to be linked.

In classical CAN mode, up to 2048 bytes of the RAM can be used in receive buffers and FIFO buffers. Up to 128 buffers are available, and 16 bytes are used per buffer. Configure the buffers so that the following conditions are met.

Number of receive buffers + total number of depth of receive FIFO buffers x + total number of depth of transmit/receive FIFO buffers k ≤ 128 buffers

In CANFD mode, up to 3584 bytes of the RAM can be used in receive buffers and FIFO buffers. Configure the buffers so that the following conditions are met.

Number of receive buffers × (12 + payload storage size) + total of (number of depth × (12 + payload storage size)) of receive FIFO buffers x + total of (number of depth × (12 + payload storage size)) of transmit/receive FIFO buffers k ≤ 3584 bytes

Figure 25.20 shows the buffer configuration. Figure 25.21 shows the buffer setting procedure.

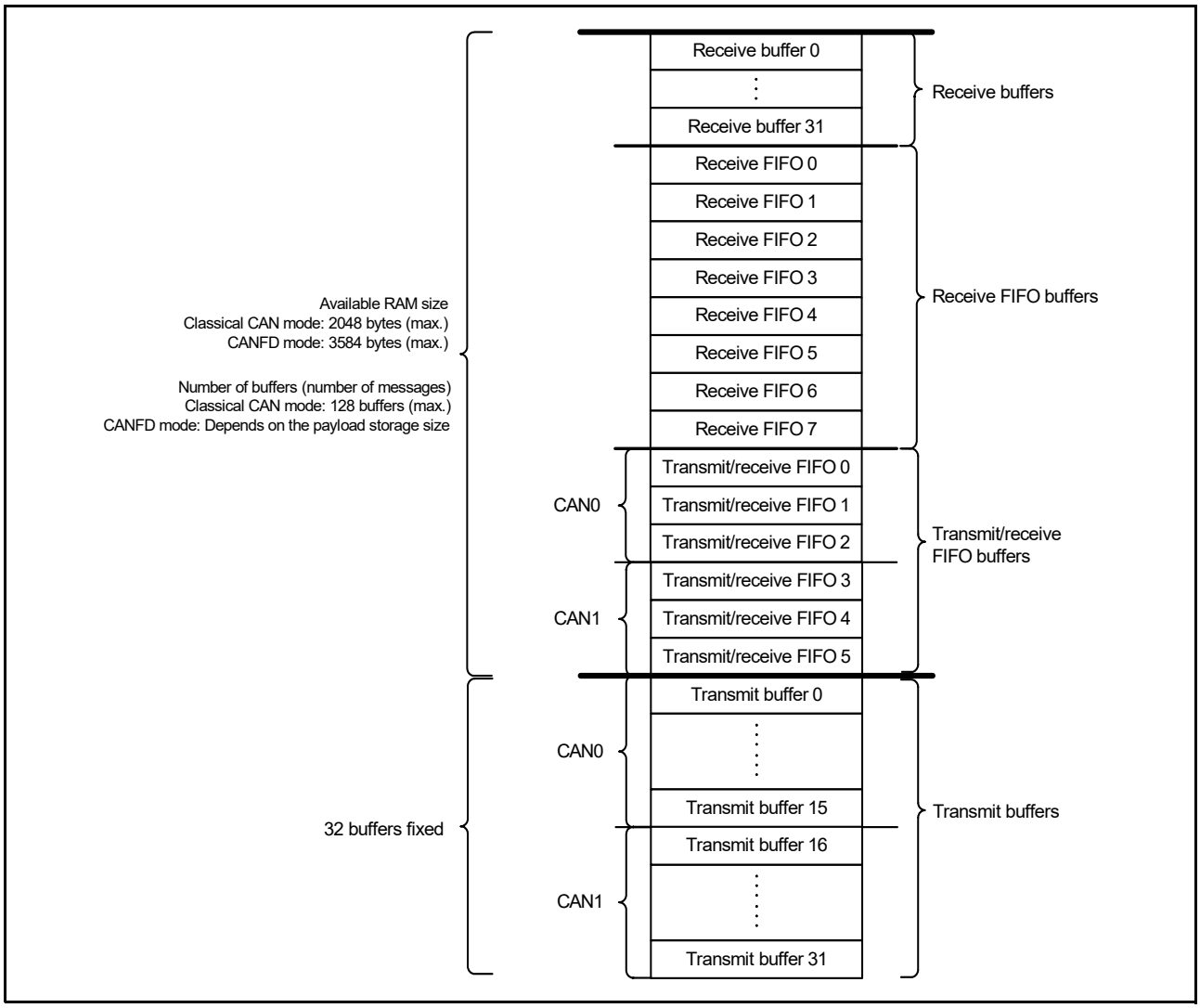
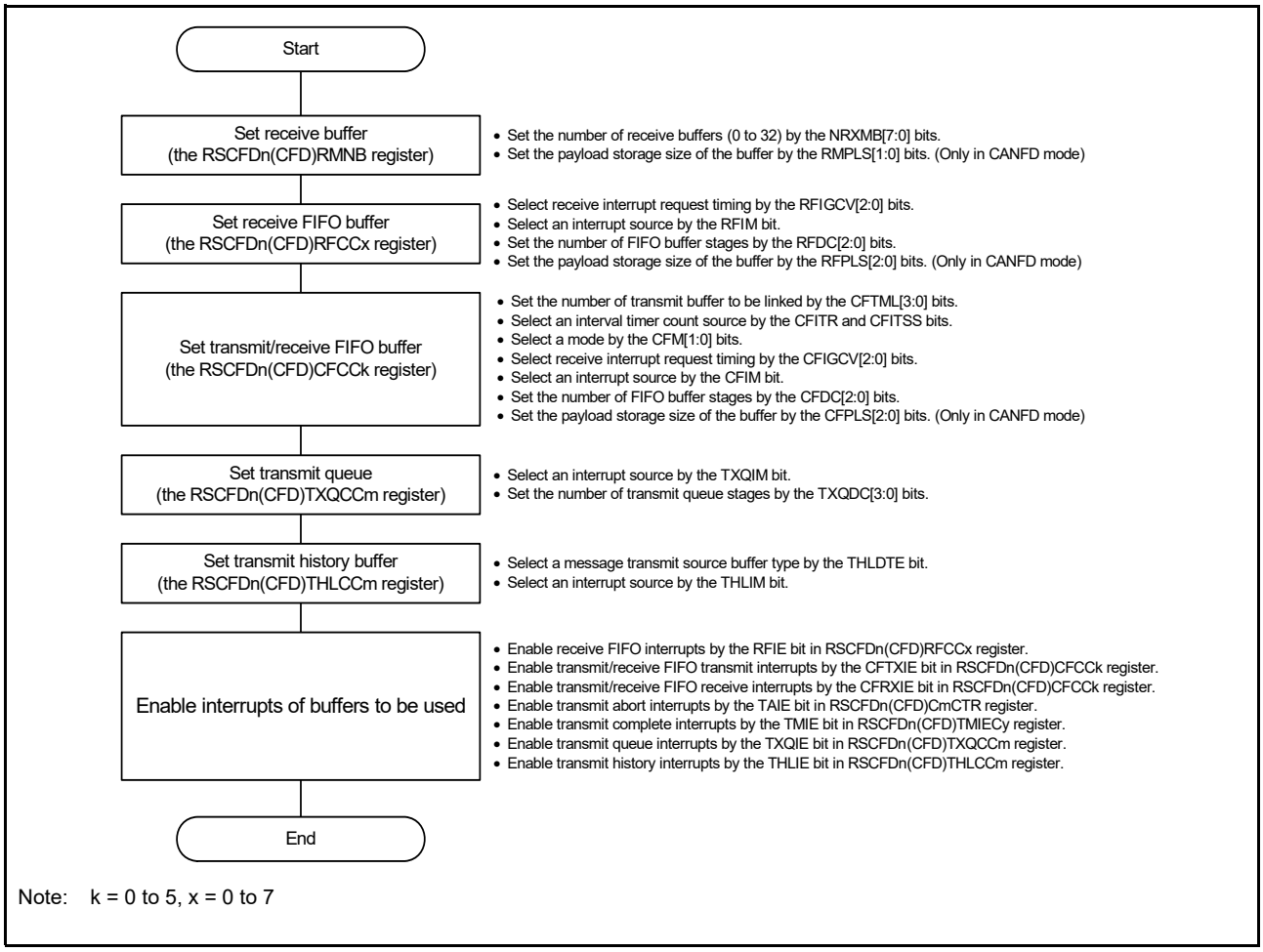


Figure 25.20 Buffer Configuration

Note: Receive buffers, receive FIFO buffers, transmit/receive FIFO buffers, and transmit buffers are located in succession.

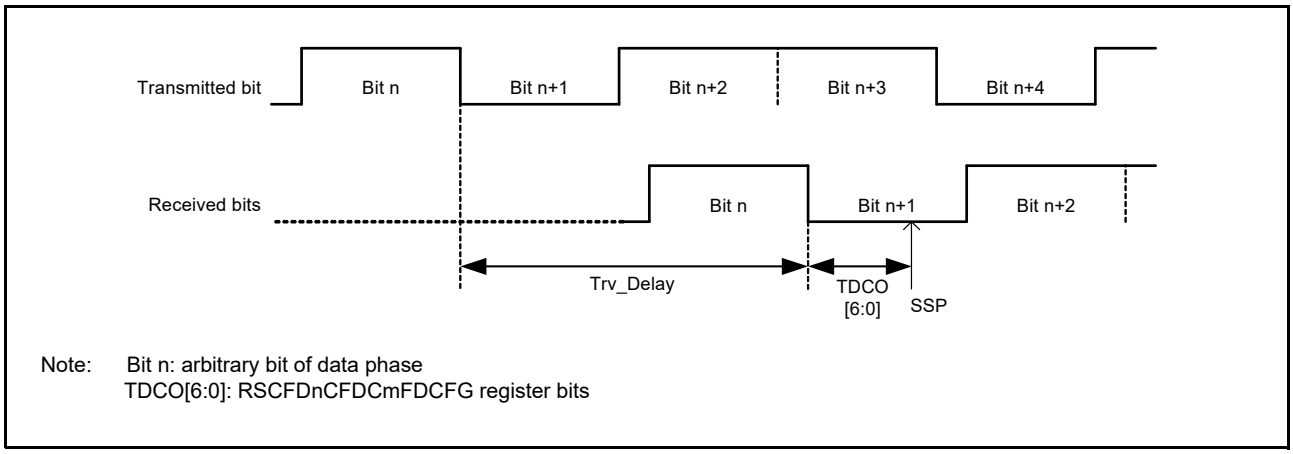


Note: k = 0 to 5, x = 0 to 7

Figure 25.21 Buffer Setting Procedure

25.11.1.6 Transmitter Delay Compensation (Only in CANFD Mode)

A high baud rate is used in CANFD mode. Transmitter delay compensation is provided as a function to accept propagation delay in this case.
To use this function, set the TDCE bit in the RSCFDnCFDCmFDCFG register to 1. Also set the secondary sample point (SSP) timing used in the data phase by the TDCOC bit and TDCO[6:0] bits in the RSCFDnCFDCmFDCFG register.
When the TDCOC bit is 0, the SSP timing equals the total value of the delay measured by the RS-CANFD module and the TDCO[6:0] value. (This value is rounded off to the nearest integer of Tq.) Usually, the TDCO[6:0] value must be equal to SS + TSEG1, the sample point timing.



Note: Bit n: arbitrary bit of data phase
TDCO[6:0]: RSCFDnCFDCmFDCFG register bits

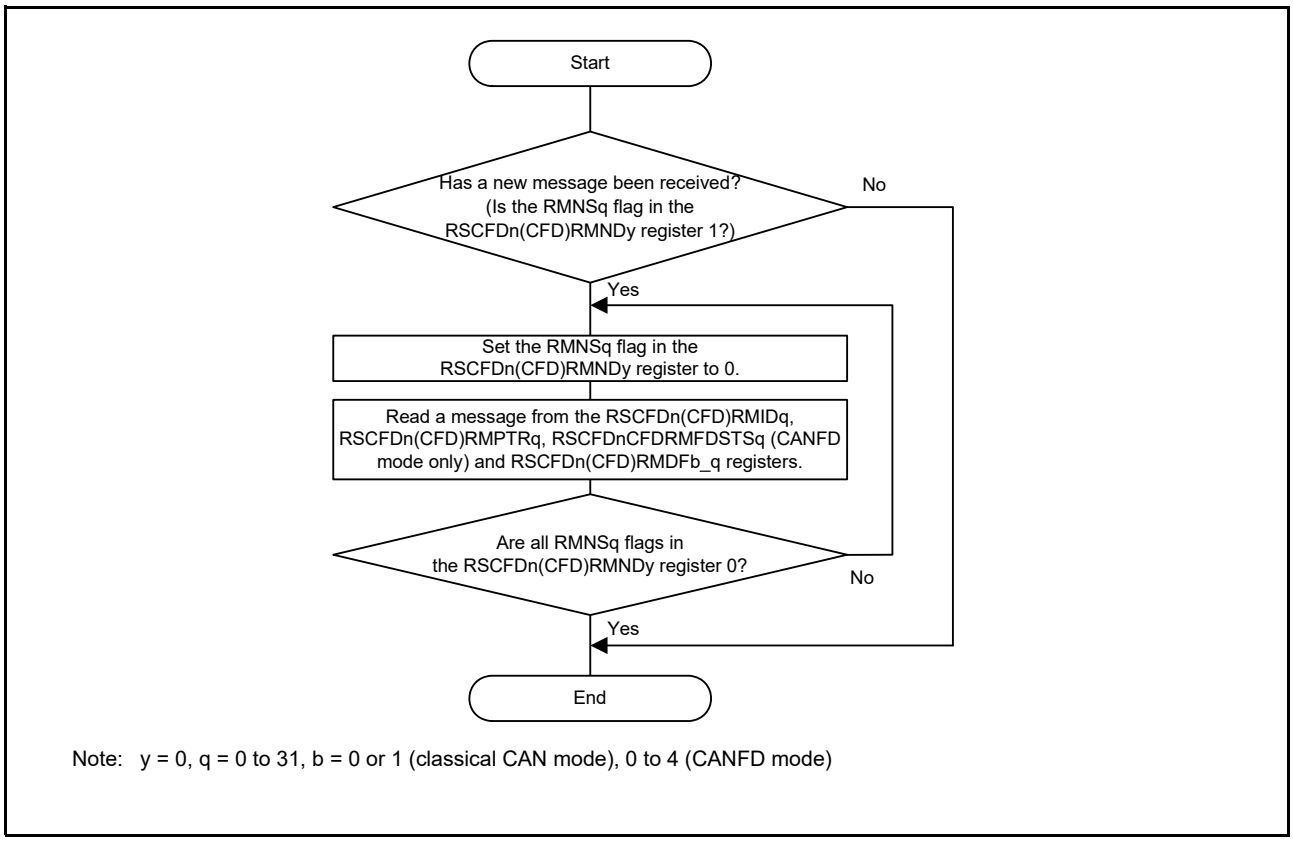
Figure 25.22 SSP Timing

When the TDCOC bit is 1, the SSP timing is determined only by the TDCO[6:0] value. (When the DBRP[7:0] value in the RSCFDnCFDCmDCFG register is larger than 0, the TDCO[6:0] value is also rounded off to the nearest integer of Tq.)
The RS-CANFD module compensates a delay up to (3 CANm bit time - 2 Tq). (Both CANm bit time and Tq are data bit rate values.)
When the TDCE bit in RSCFDnCFDCmFDCFG register is 1 (transmitter delay compensation is enabled), set bits NBRP[9:0] and DBRP[7:0] to the same value that is less than 1.

25.11.2 Reception Procedure

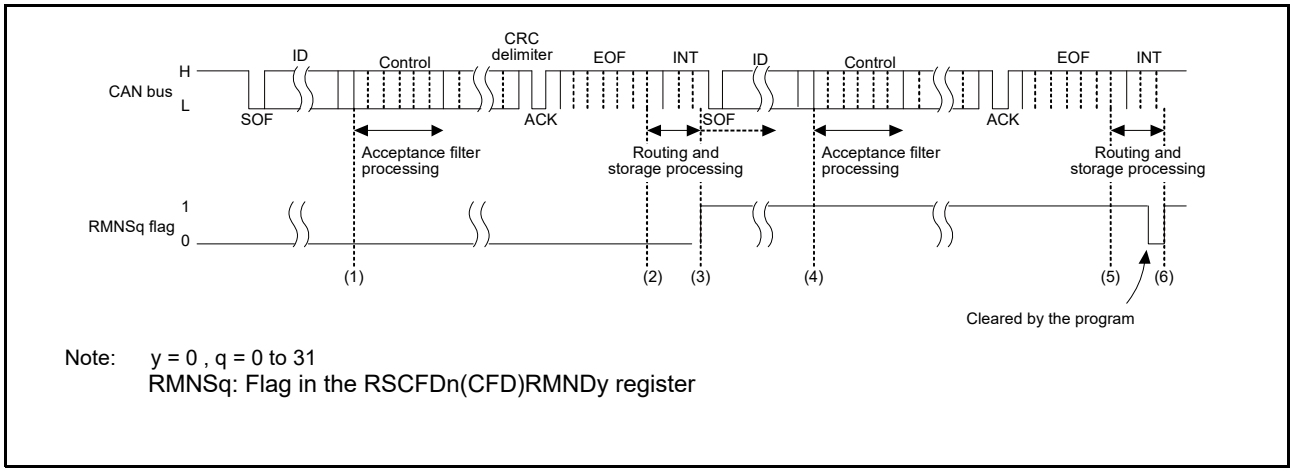
25.11.2.1 Receive Buffer Reading Procedure

When the processing to store received messages in a receive buffer starts, the RMNSq flag in the RSCFDn(CFD)RMNDy register (y = 0, q = 0 to 31) is set to 1 (receive buffer q contains a new message). Messages can be read from registers RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq (only in CANFD mode), and RSCFDn(CFD)RMDFb_q (b = 0 or 1 in classical CAN mode, b = 0 to 4 in CANFD mode). Figure 25.23 shows the receive buffer reading procedure. This procedure ensures the consistency of messages read from registers RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq, and RSCFDn(CFD)RMDFb_q.



Note: y = 0, q = 0 to 31, b = 0 or 1 (classical CAN mode), 0 to 4 (CANFD mode)

Figure 25.23 Receive Buffer Reading Procedure



Note: y = 0 , q = 0 to 31
RMNSq: Flag in the RSCFDn(CFD)RMNDy register

Figure 25.24 Receive Buffer Reception Timing Chart

(1) When the ID field in a message has been received, the acceptance filter processing starts.
(2) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.
(3) When the message has passed through the DLC filter processing, the processing to store the message in the specified receive buffer starts.
When the message storage processing starts, the RMNSq flag in the corresponding RSCFDn(CFD)RMNDy register is set to 1 (the receive buffer contains a new message). If other channels are performing filter processing or transmit priority determination processing, the routing processing and the storage processing may be delayed.
(4) When the ID field of the next message has been received, the acceptance filter processing starts.
(5) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.
(6) When the corresponding RMNSq flag is cleared to 0 (the receive buffer contains no new message), this flag is set to 1 again when the message storage processing starts. Even if the RMNSq flag remains 1, a new message is overwritten to the receive buffer. The RMNSq flag should not be cleared to 0 during storage of messages.

25.11.2.2 FIFO Buffer Reading Procedure

When received messages have been stored in one or more receive FIFO buffers or a transmit/receive FIFO buffer that is set to receive mode or gateway mode, the corresponding message count display counter (RFMC[7:0] bits in the RSCFDn(CFD)RFSTSx register (x = 0 to 7) or CFMC[7:0] bits in the RSCFDn(CFD)CFSTSk register (k = 0 to 17)) is incremented. At this time, when the RFIE bit (receive FIFO interrupt is enabled) in the RSCFDn(CFD)RFCCx register or the CFRXIE bit (transmit/receive FIFO receive interrupt is enabled) in the RSCFDn(CFD)CFCCk register is set to 1, an interrupt request is generated. Received messages can be read from the RSCFDn(CFD)RFIDx, RSCFDn(CFD)RFPTRx, RSCFDnCFDRFFDSTSx (only in CANFD mode), and RSCFDn(CFD)RFDFd_x (d = 0 or 1 in classical CAN mode, d = 0 to 15 in CANFD mode) registers for receive FIFO buffers, or from the RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, RSCFDnCFDCFFDCSTSk (only in CANFD mode), and RSCFDn(CFD)CFDFd_k registers for transmit/receive FIFO buffers. Messages in FIFO buffers can be read sequentially on a first-in, first-out basis.

When the message count display counter value matches the FIFO buffer depth (a value set by the RFDC[2:0] bits in the RSCFDn(CFD)RFCCx register or the CFDC[2:0] bits in the RSCFDn(CFD)CFCCk register), the RFFLL or CFFLL flag is set to 1 (the receive FIFO buffer is full).

When all messages have been read out of the FIFO buffer, the RFEMP flag in the RSCFDn(CFD)RFSTSx register or the CFEMP flag in the RSCFDn(CFD)CFSTSk register is set to 1 (the receive FIFO buffer contains no unread message (buffer empty)).

If the RFE bit or the CFE bit is cleared to 0 (no receive FIFO buffer is used) with the interrupt request flag (RFIF flag in the RSCFDn(CFD)RFSTSx register or CFRXIF flag in the RSCFDn(CFD)CFSTSk register) set to 1 (a receive FIFO interrupt request is present), the interrupt request flag is not automatically cleared to 0. The program must clear the interrupt request flag to 0.

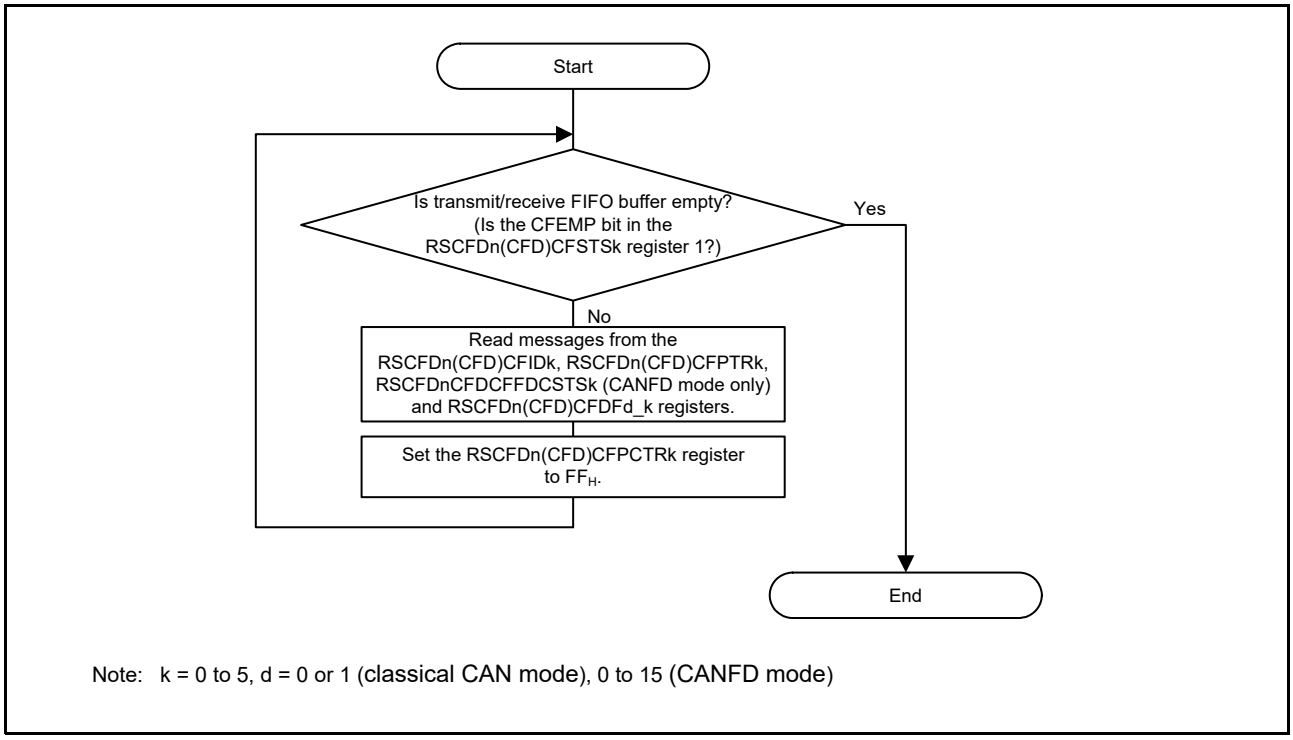


Figure 25.25 Transmit/Receive FIFO Buffer Reading Procedure

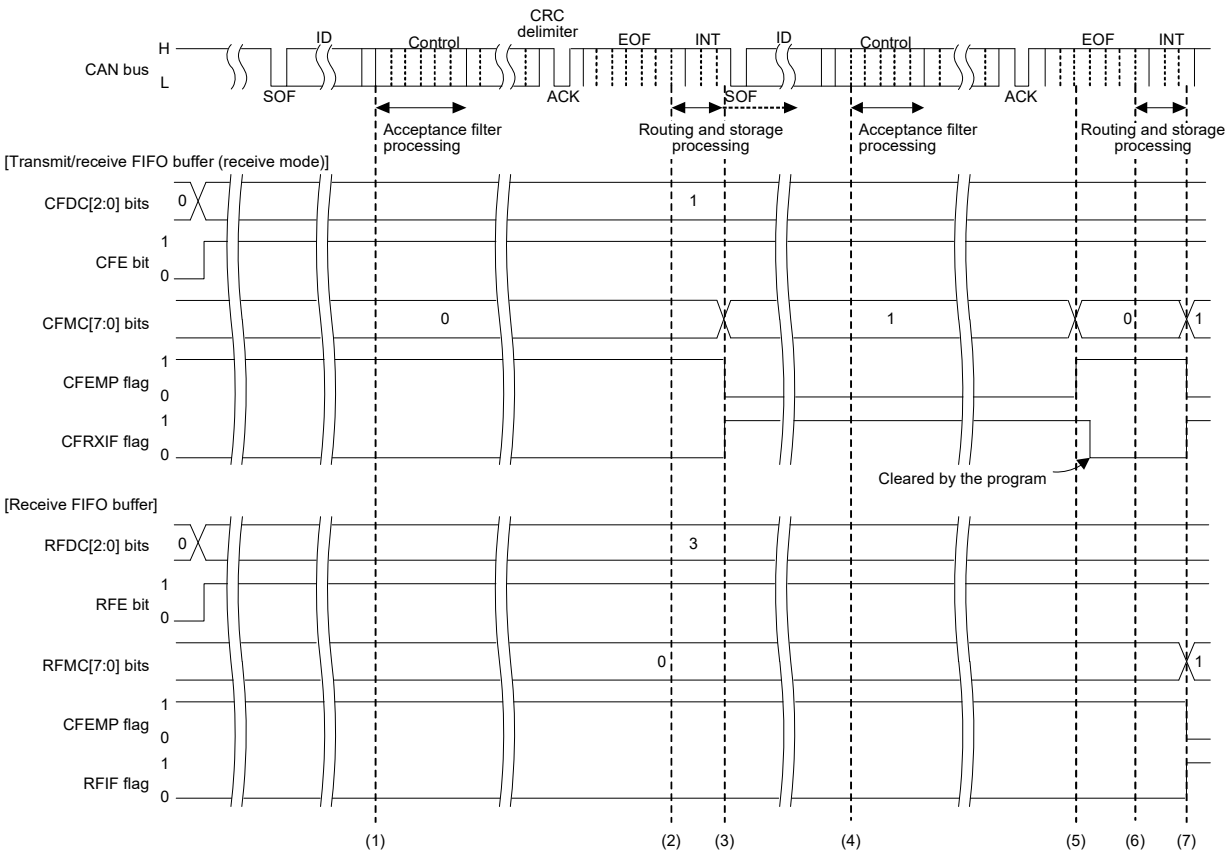
When reading a message in CANFD mode, do not read the RSCFDnCFDRFDFd_x or RSCFDnCFDCFDFd_k register corresponding to the area exceeding the payload storage size specified by the RFPLS[2:0] bits in the RSCFDnCFDRFCCx register or the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

Table 25.182 Payload Storage Area of Receive FIFO Buffer

Set RFPLS[2:0] Value	Payload Storage Size	Corresponding Data Field Registers
000_B	8 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF1_x
001_B	12 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF2_x
010_B	16 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF3_x
011_B	20 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF4_x
100_B	24 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF5_x
101_B	32 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF7_x
110_B	48 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF11_x
111_B	64 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF15_x

Table 25.183 Payload Storage Area of Transmit/Receive FIFO Buffer

Set CFPLS[2:0] Value	Payload Storage Size	Corresponding Data Field Registers
000_B	8 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF1_k
001_B	12 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF2_k
010_B	16 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF3_k
011_B	20 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF4_k
100_B	24 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF5_k
101_B	32 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF7_k
110_B	48 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF11_k
111_B	64 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF15_k



CFDC[2:0], CFE: Bits in the RSCFDn(CFD)CFCCk register
CFMC[7:0], CFEMP, CFRXIF: Flags in the RSCFDn(CFD)CFSTSk register
RFDC[2:0], RFE: Bits in the RSCFDn(CFD)RFCCx register
RFMC[7:0], RFEMP, RFIF: Flags in the RSCFDn(CFD)RFSTSx register

Note: k = 0 to 5, x = 0 to 7

Figure 25.26 FIFO Buffer Reception Timing Chart

(1) When the ID field in a message has been received, the acceptance filter processing starts.

(2) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(3) When the message has passed through the DLC filter processing and the CFE bit in the RSCFDn(CFD)CFCCk register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCFDn(CFD)CFCCk register is 001_B or more, the message is stored in the transmit/receive FIFO buffer that is set to receive mode. The CFMC[7:0] value in the RSCFDn(CFD)CFSTSk register is incremented and becomes 01_H. When the CFIM bit in the RSCFDn(CFD)CFCCk register is set to 1 (a FIFO receive interrupt request is generated each time a message has been received), the CFRXIF flag in the RSCFDn(CFD)CFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). The CFRXIF flag can be reset to 0 by the program.

(4) When the ID field of the next message has been received, the acceptance filter processing starts.

(5) Read received messages from the RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, and RSCFDn(CFD)CFDFd_k registers and write FF_H to the RSCFDn(CFD)CFPCTRk register. This causes the CFMC[7:0] bits in the RSCFDn(CFD)CFSTSk register to be decremented. When CFMC[7:0] becomes 00_H, the CFEMP flag in the RSCFDn(CFD)CFSTSk register becomes 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

(6) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(7) The message is stored in the transmit/receive FIFO buffer set in receive mode when the message has passed through the DLC filter process if the CFE bit is set to 1 (transmit/receive FIFO buffers are used), and the CFDC[2:0] bits are set to 001_B or more. The CFMC[7:0] bit value is incremented by 1 to be 01_H. When the CFIM bit is set to 1 (an interrupt occurs each time a message has been received), the CFRXIF flag is set to 1 (a transmit/receive FIFO receive interrupt request is present).
The message is stored in the receive FIFO buffer if the RFE bit in the RSCFDn(CFD)RFCCx register is set to 1 (receive FIFO buffers are used), and the RFDC[2:0] bits in the RSCFDn(CFD)RFCCx register are set to 001_B or more. The RFMC[7:0] bits in the RSCFDn(CFD)RFSTSx register are set to 01_H by being incremented by 1. When the RFIM bit in the RSCFDn(CFD)RFCCx register is set to 1 (an interrupt occurs each time a message has been received), the RFIF flag in the RSCFDn(CFD)RFSTSx register is set to 1 (a receive FIFO interrupt request is present).

25.11.2.3 FIFO Buffer Reading Procedure by DMA Transfer

In CANFD mode, the following FIFO buffers can be read by DMA transfer.

- All receive FIFO buffers x (x = 0 to 7)
- The first transmit/receive FIFO buffer k allocated to channel m (k = 3 × m, m = 0, 1)

The DMA enable bit (RFDMAEx or CFDMAEm bit in the RSCFDnCFDCDTC register) can be set at any time. However, before setting the DMA enable bit to 1 (to enable DMA transfer requests), set the receive interrupt enable bit (RFIE bit in the RSCFDnCFDRFCCx register or CFRXIE bit in the RSCFDnCFDCFCCk register) of related FIFO buffers to 0 (to disable interrupts). When DMA transfer requests are enabled, do not write a value to the FIFO control register (RSCFDnCFDRFCCx register or RSCFDnCFDCFCCk register).

When an unread message is remaining in a DMA transfer-enabled FIFO buffer, a DMA transfer request trigger is generated. Specify the FIFO access register address for the transfer source address, and adjust the transfer size so that data can be read to the end of the payload storage area with a single trigger. The end of the payload storage area depends on the payload storage size specified by the RFPLS[2:0] bits in the RSCFDnCFDRFCCx register or the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

After the end of the payload stored in the FIFO buffer has been read, the RFMC[7:0] value in the RSCFDnCFDRFSTSx register or the CFMC[7:0] value in the RSCFDnCFDCFSTSk register is automatically decremented. After that, if an unread message is remaining in the FIFO buffer, a trigger is generated again.

When the RFDMAEx or CFDMAEm bit is set to 0 (to disable DMA transfer requests) during DMA transfer, wait until the DMA transfer status (RFDMASTSx or CFDMASTSm bit in the RSCFDnCFDCDTSTS register) is cleared to 0 (DMA transfer disabled), and then start the next processing (enabling DMA transfer again etc.). When disabling DMA transfer, examine how to process a message remaining in the FIFO buffer and a newly arriving message. When the FIFO buffer is enabled, it continues to receive messages.

25.11.3 Transmission Procedure

25.11.3.1 Procedure for Transmission from Transmit Buffers

Figure 25.27 shows the procedure for transmission from transmit buffers.
Figure 25.28 shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmission has been successfully completed. Figure 25.29 shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmit abort has been completed.

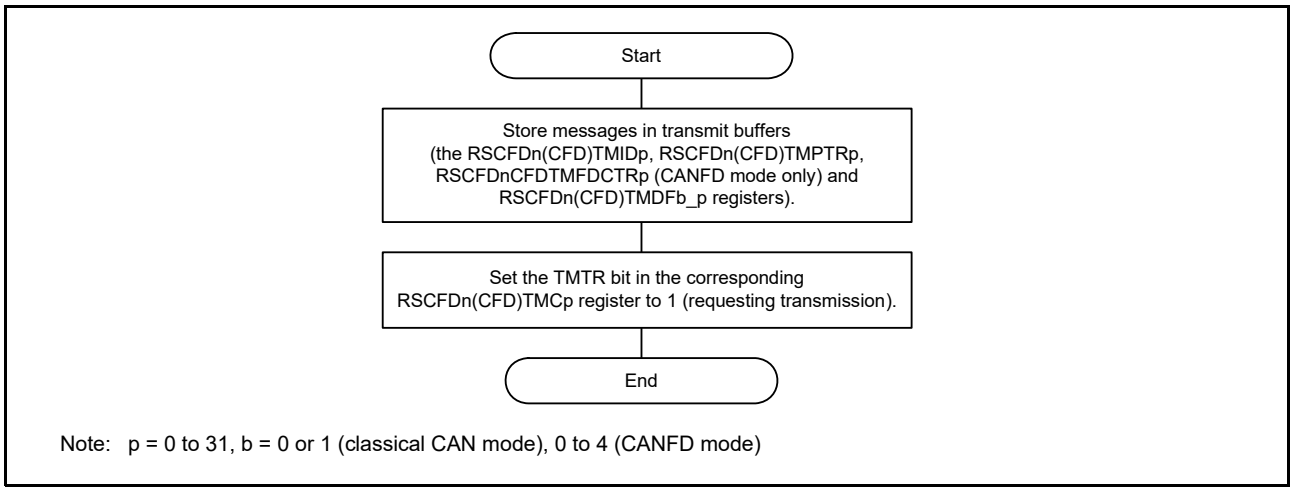
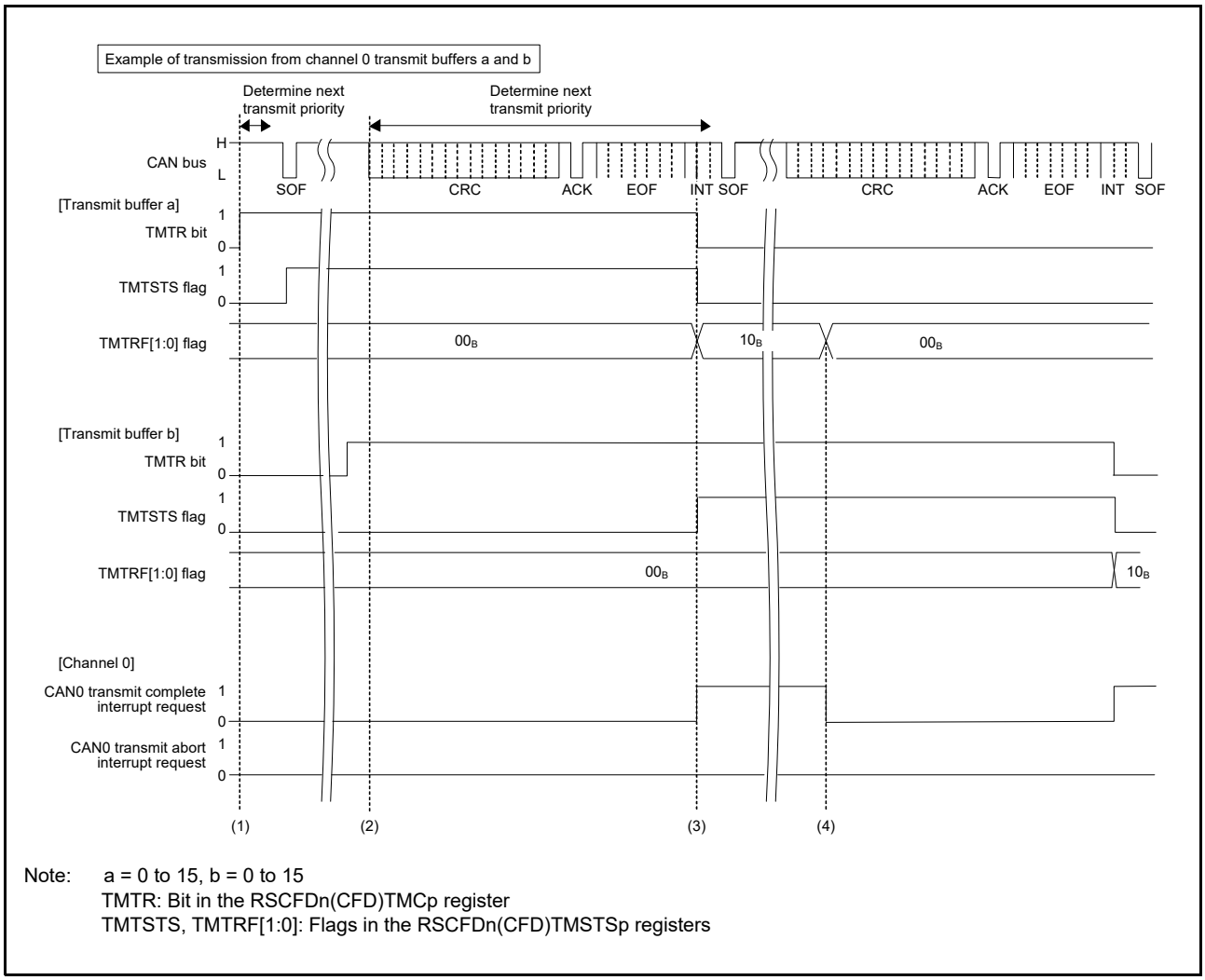


Figure 25.27 Procedure for Transmission from Transmit Buffers

In CANFD mode and transmit buffer merge mode, messages with a payload size of more than 20 bytes can be transmitted from transmit buffers (16 × m) + 0 and transmit buffers (16 × m) + 3. At this time, transmit buffers (16 × m) + 1 to (16 × m) + 2 and transmit buffers (16 × m) + 4 to (16 × m) + 5 are allocated as a payload storage area. Registers RSCFDnCFDTMIDp, RSCFDnCFDTMPTRp, and RSCFDnCFDTMFDCTRp corresponding to these buffers can be used as data field registers that can store 4-byte data bytes (payload) like the RSCFDnCFDTMDFb_p register. Table 25.184 shows message storage registers when transmitting a message with a payload size of more than 20 bytes from transmit buffer 0.

Table 25.184 Message Storage Registers in Transmit Buffer Merge Mode (Example of Transmit Buffer 0)

Transmit Buffer	Offset from Base Address	Symbol	Register Function in Transmit Buffer Merge Mode
Transmit buffer 0	4000_H	RSCFDnCFDTMID0	Transmit buffer 0 ID data, transmit history data store enable bit, RTR bit,and IDE bit
	4004_H	RSCFDnCFDTMPTR0	Transmit buffer 0 label data and DLC data
	4008_H	RSCFDnCFDTMFDCTR0	Transmit buffer 0 ESI bit, BRS bit, and FDF bit
	$400C_H$ to $401C_H$	RSCFDnCFDTMDF0_0 to RSCFDnCFDTMDF4_0	Transmit buffer 0 data bytes 0, 1, 2, and 3 to transmit buffer 0 data bytes 16, 17, 18, and 19
Transmit buffer 1	4020_H	RSCFDnCFDTMID1	Transmit buffer 0 data bytes 20, 21, 22, and 23
	4024_H	RSCFDnCFDTMPTR1	Transmit buffer 0 data bytes 24, 25, 26, and 27
	4028_H	RSCFDnCFDTMFDCTR1	Transmit buffer 0 data bytes 28, 29, 30, and 31
	$402C_H$ to $403C_H$	RSCFDnCFDTMDF0_1 to RSCFDnCFDTMDF4_1	Transmit buffer 0 data bytes 32, 33, 34, and 35 to transmit buffer 0 data bytes 48, 49, 50, and 51
Transmit buffer 2	4040_H	RSCFDnCFDTMID2	Transmit buffer 0 data bytes 52, 53, 54, and 55
	4044_H	RSCFDnCFDTMPTR2	Transmit buffer 0 data bytes 56, 57, 58, and 59
	4048_H	RSCFDnCFDTMFDCTR2	Transmit buffer 0 data bytes 60, 61, 62, and 63
	$404C_H$ to $405C_H$	RSCFDnCFDTMDF0_2 to RSCFDnCFDTMDF4_2	Not used



Note: a = 0 to 15, b = 0 to 15
TMTR: Bit in the RSCFDn(CFD)TMCp register
TMTSTS, TMTRF[1:0]: Flags in the RSCFDn(CFD)TMSTSp registers

Figure 25.28 Transmit Buffer Transmission Timing Chart (Transmission Completed Successfully)

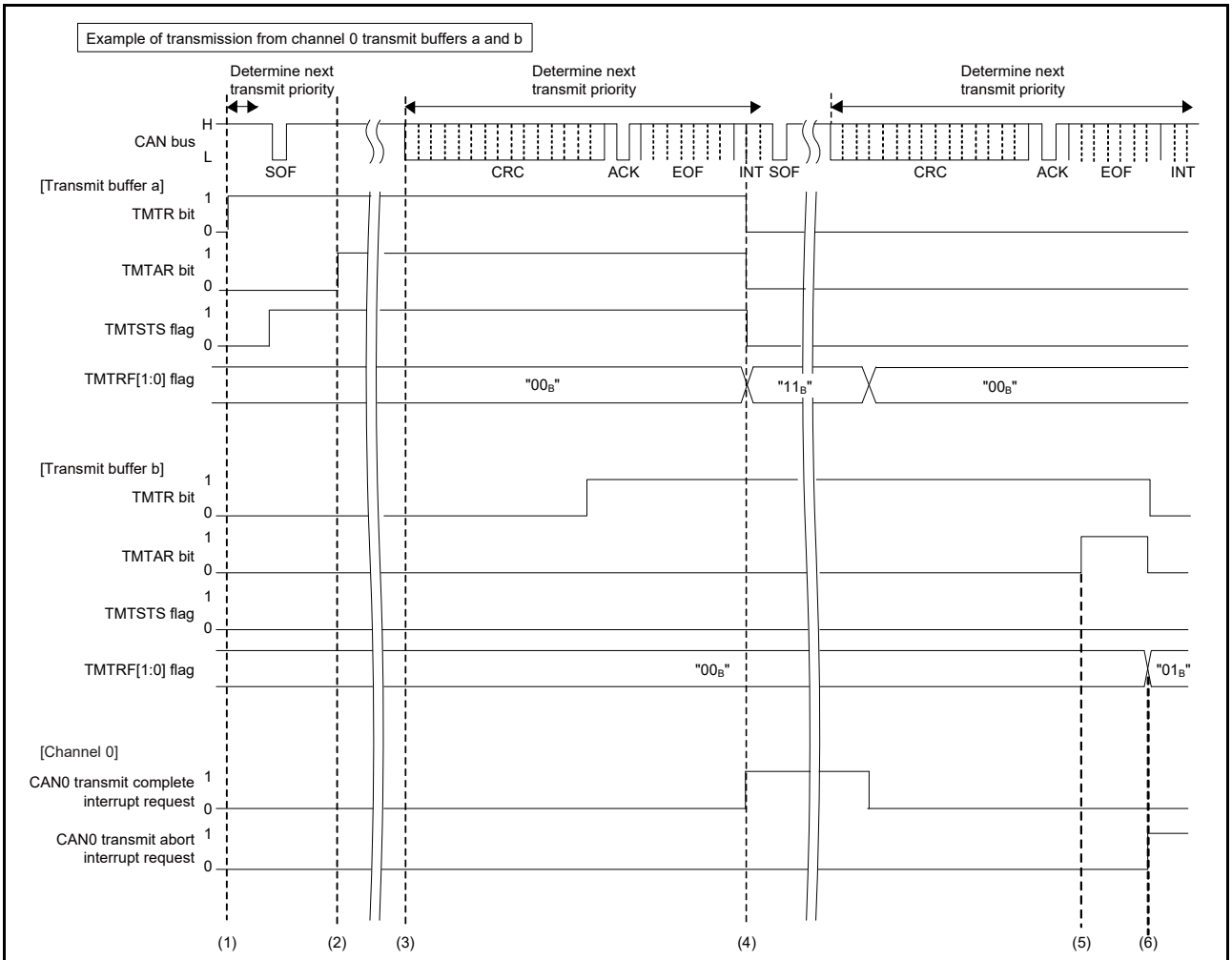
(1) When the TMTR bit in the RSCFDn(CFD)TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCFDn(CFD)TMSTSa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.

(2) When a transmit request from a buffer is present, the priority determination starts at the first bit of CRC field for the next transmission. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.

(3) When transmission completes successfully, the TMTRF[1:0] flag in the RSCFDn(CFD)TMSTSa register is set to 10_B (transmission has been completed (without transmit abort request)) and the TMTSTS flag and the TMTR bit in the RSCFDn(CFD)TMCa register are cleared to 0. When the TMIEa bit in the RSCFDn(CFD)TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmit complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00_B (transmission is in progress or no transmit request is present).

(4) Before starting the next transmission, set the TMTRF[1:0] flag to 00_B. Write the next message to the transmit buffer, and then set the TMTR bit to 1 (transmission is requested). The TMTR bit can be set to 1 only when the TMTRF[1:0] flag value is 00_B.

If an arbitration-lost has occurred after transmission is started, the TMTSTS flag is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC field to search the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration loss, the priority determination processing is reexecuted during transmission of an error frame.



Note: a = 0 to 15, b = 0 to 15
TMTR, TMTAR: Bits in the RSCFDn(CFD)TMCp register
TMTSTS, TMTRF[1:0]: Flags in the RSCFDn(CFD)TMSTSp register

Figure 25.29 Transmit Buffer Transmission Timing Chart (Transmit Abort Completed)

(1) When the TMTR bit in the RSCFDn(CFD)TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCFDn(CFD)TMSTSa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.

(2) When it is determined that the transmit buffer is used for the next transmission or transmission is in progress, message transmission is not aborted unless an error or arbitration loss occurs even if the TMTAR bit is set to 1 (transmit abort is requested).

(3) The priority determination starts at the first bit of the CRC field for the next transmission. In this timing chart, buffer b is not selected as the next transmit buffer. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.

(4) When transmission completes successfully, the TMTRF[1:0] flag in the RSCFDn(CFD)TMSTSa register is set to 11_B (transmission has been completed (with transmit abort request)) and the TMTSTS flag and the TMTR bit in the RSCFDn(CFD)TMCa register are cleared to 0. When the TMIEa value in the RSCFDn(CFD)TMIEC0 register is 1

(transmit buffer interrupt is enabled), a CAN0 transmit complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00_B (transmission is in progress or no transmit request is present).

(5) While another CAN node is transmitting data on the CAN bus (TMTSTS flag = 0), if the TMTAR bit is set to 1 while the corresponding channel is determining transmit priority, the TMTR bit cannot be cleared to 0.

(6) After the internal processing time has passed, the transmission is terminated and the TMTRF[1:0] flag is set to 01_B. When the transmit buffer is not transmitting data and is not selected as the next transmit buffer and priority determination is not being made, an abort request is immediately accepted and the TMTRF[1:0] flag is set to 01_B. At this time, the TMTR and TMTAR bits are cleared to 0. When transmit abort is completed with the TAIE bit in the RSCFDn(CFD)CmCTR register set to 1 (transmit abort interrupt is enabled), an interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00_B.

If an arbitration loss has occurred after the CAN channel started transmission, the TMTSTS bit is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC field to find the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration loss, the priority determination processing is reexecuted during transmission of an error frame.

25.11.3.2 Procedure for Transmission from Transmit/Receive FIFO Buffers

Figure 25.30 shows the procedure for transmission from transmit/receive FIFO buffers.
Figure 25.31 shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmission has been successfully completed. Figure 25.32 shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmit abort has been completed.

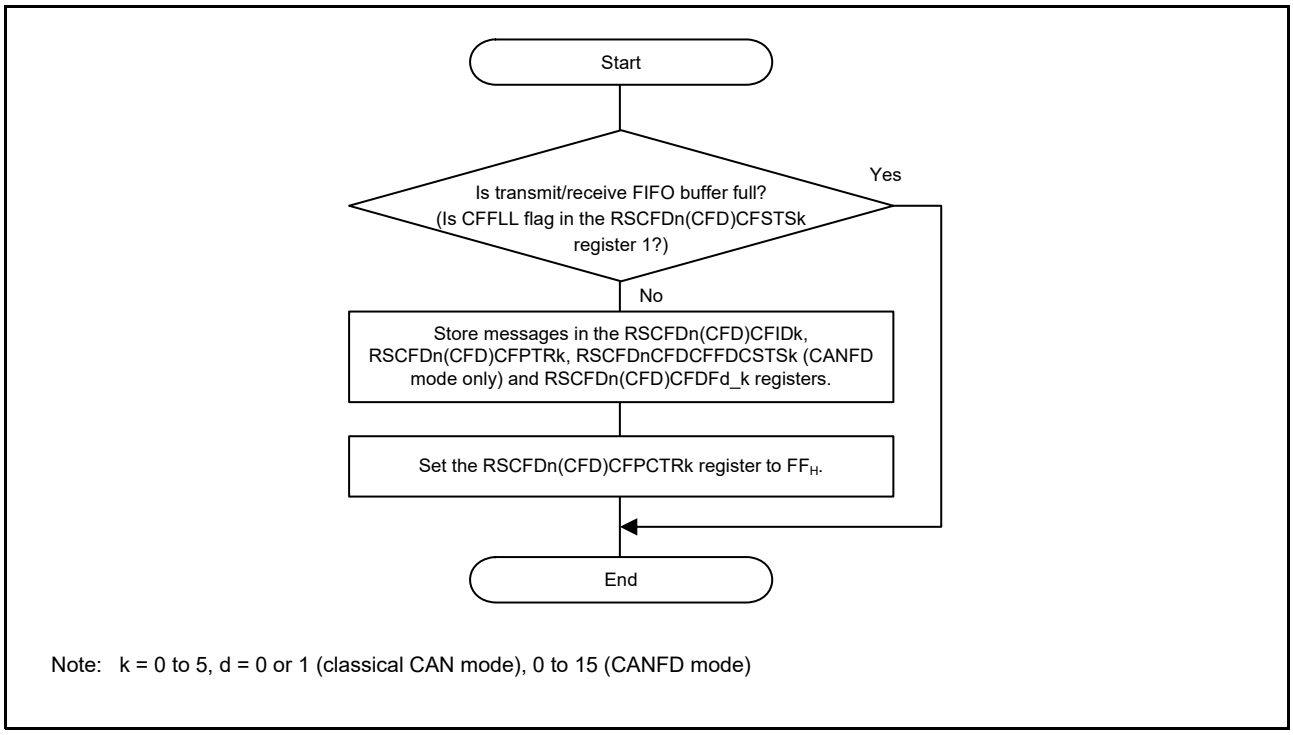
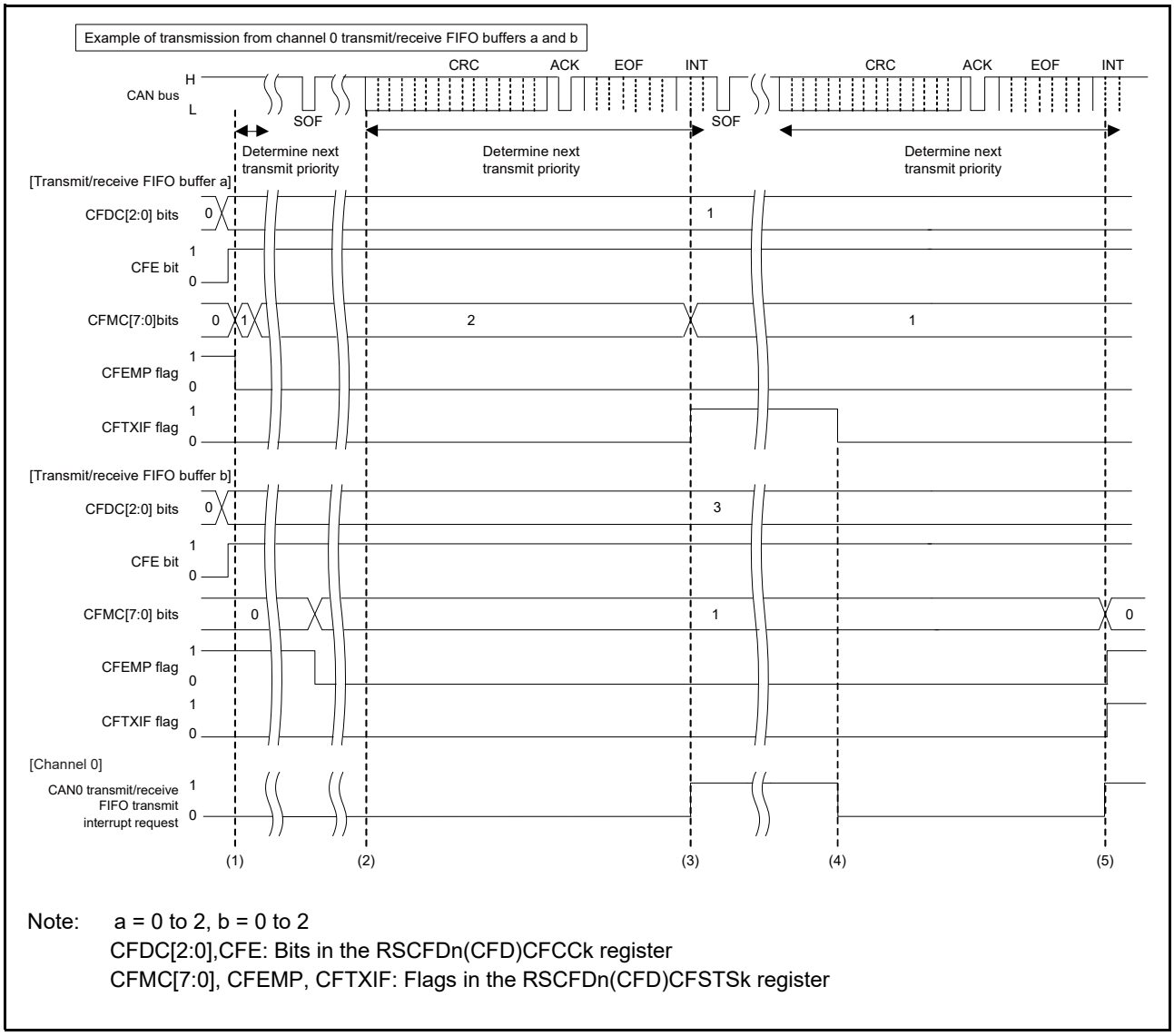


Figure 25.30 Procedure for Transmission from Transmit/Receive FIFO Buffers

When storing a message, do not write a value to the RSCFDnCFDCFDFd_k register corresponding to the area exceeding the payload storage size specified by the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

Table 25.185 Payload Storage Area of Transmit/Receive FIFO Buffer

Set CFPLS[2:0] Value	Payload Storage Size	Corresponding Data Field Registers
000_B	8 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF1_k
001_B	12 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF2_k
010_B	16 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF3_k
011_B	20 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF4_k
100_B	24 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF5_k
101_B	32 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF7_k
110_B	48 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF11_k
111_B	64 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF15_k



Note: a = 0 to 2, b = 0 to 2
CFDC[2:0],CFE: Bits in the RSCFDn(CFD)CFCCk register
CFMC[7:0], CFEMP, CFTXIF: Flags in the RSCFDn(CFD)CFSTSk register

Figure 25.31 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Completed Successfully)

(1) While the CAN bus is idle, when the CFE bit in the RSCFDn(CFD)CFCCa register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCFDn(CFD)CFCCa register is 001_B (4 messages) or more and the CFMC[7:0] value in the RSCFDn(CFD)CFSTSa register is 01_H or more, the priority determination processing starts to determine the highest-priority transmit message. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.

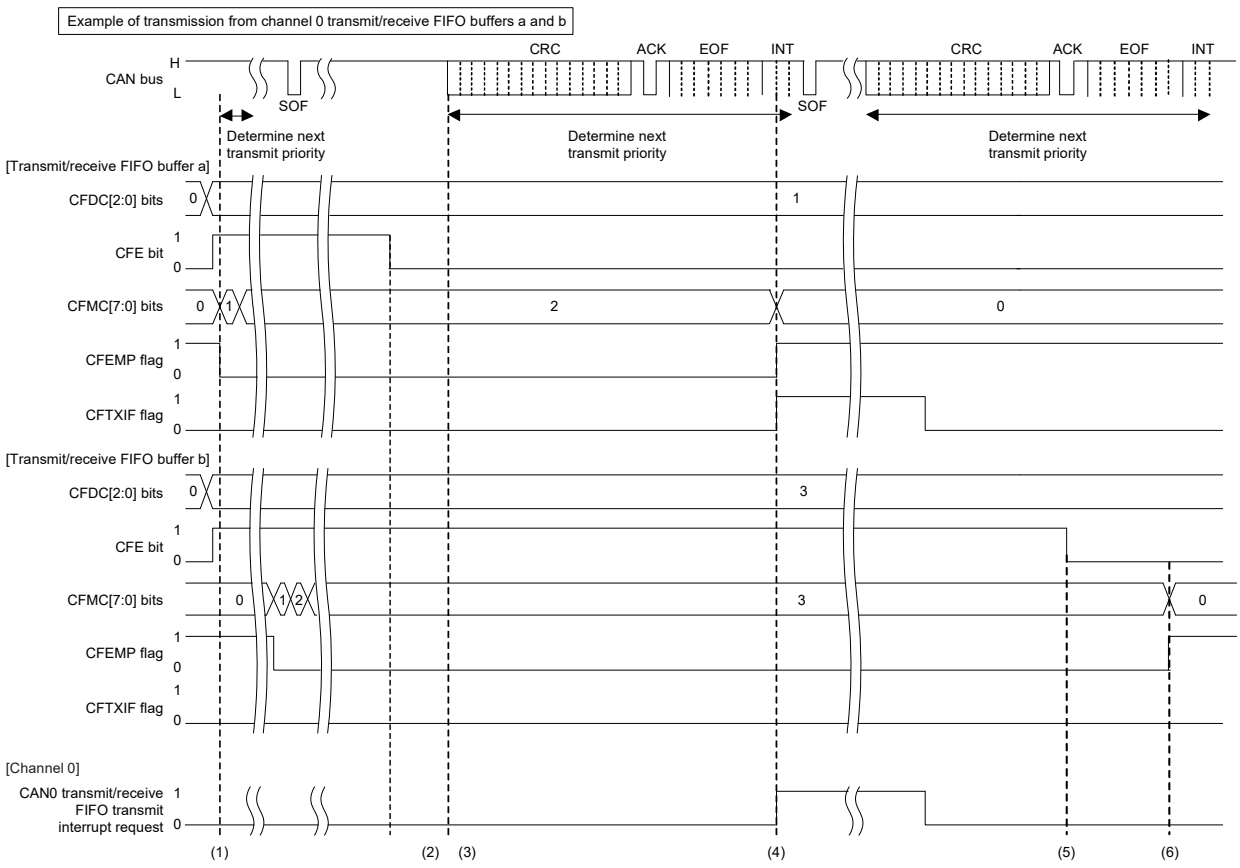
(2) When a transmit request from a buffer is present, the priority determination starts at the first bit of the CRC field for the next transmission. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.

(3) When transmission completes successfully, the CFMC[7:0] value in the RSCFDn(CFD)CFSTSa register is decremented. Setting the CFIM bit in the RSCFDn(CFD)CFCCa register to 1 (a FIFO transmit interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCFDn(CFD)CFSTSk register to 1 (a transmit/receive FIFO transmit interrupt request is present).

(4) The program can clear the CFTXIF flag.

(5) Message transmission from transmit/receive FIFO buffer b of channel 0 has been completed and the CFMC[7:0] value in the RSCFDn(CFD)CFSTSb register is decremented. The CFMC[7:0] bits are cleared to 00_H and therefore the CFEMP flag in the RSCFDn(CFD)CFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

Transmission is continued until the CFEMP flag is set to 1. It is possible to continuously store transmit messages in FIFO buffers until the CFFLL flag in the RSCFDn(CFD)CFSTSa and RSCFDn(CFD)CFSTSb register is set to 1 (the transmit/receive FIFO buffer is full).



*a = 0 to 2, b = 0 to 2
CFDC[2:0], CFE: Bits in the RSCFDn(CFD)CFCCk register
CFMC[7:0], CFEMP, CFTXIF: Flags in the RSCFDn(CFD)CFSTSk register

Figure 25.32 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmit Abort Completed)

(1) While the CAN bus is idle, when the CFE bit in the RSCFDn(CFD)CFCCa register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCFDn(CFD)CFCCa register is 001_B (4 messages) or more and the CFMC[7:0] value in the RSCFDn(CFD)CFSTSa register is 01_H or more, the priority determination processing starts to determine the highest-priority transmit message. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.

(2) When transmission is in progress or it is determined that the transmit/receive FIFO buffer is used for the next transmission, message transmission is not aborted unless an error or arbitration loss occurs even if the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

(3) When a transmit request from a buffer is present, the priority determination starts at the first bit of the CRC field for the next transmission. In this figure, transmit/receive FIFO buffer b is not selected as a buffer for the next transmission. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.

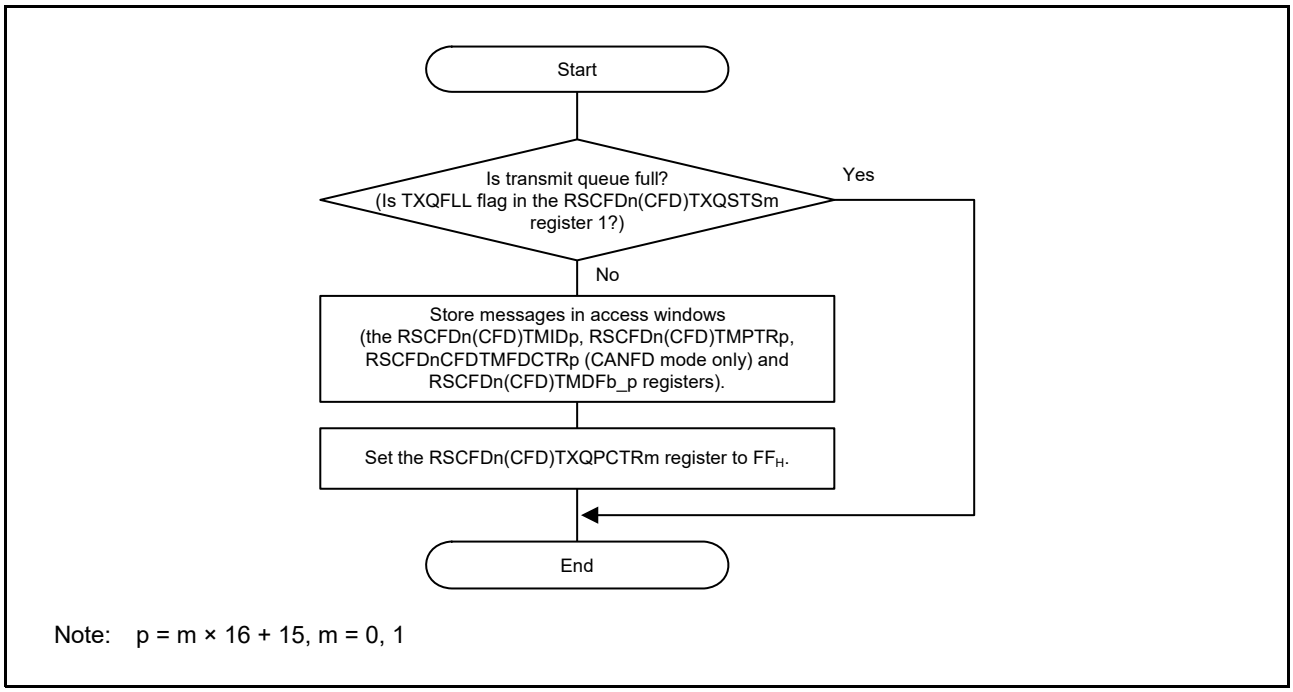
(4) When transmit completes successfully, the CFMC[7:0] value is cleared to 00_H. Setting the CFIM bit to 1 (a FIFO transmit interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCFDn(CFD)CFSTSa register to 1 (a transmit/receive FIFO transmit interrupt request is present). The program can clear the CFTXIF flag.

(5) If another CAN node on the CAN bus is transmitting data (not from transmit/receive FIFO buffer b), transmit/receive FIFO buffers cannot be disabled immediately even if the CFE bit in the RSCFDn(CFD)CFCCb register is cleared to 0 (no transmit/receive FIFO buffer is used) during transmit priority determination. (The CFEMP flag in the RSCFDn(CFD)CFSTSb register is not set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) immediately.)

(6) After the internal processing time has passed, transmit/receive FIFO buffers are disabled and the CFMC[7:0] bits in the RSCFDn(CFD)CFSTSb register are cleared to 00_H and the CFEMP flag is set to 1. When the transmit/receive FIFO buffer is not transmitting data and is not selected as the next transmit buffer and priority determination is not in progress, the transmit/receive FIFO buffer is immediately disabled. (The CFMC[7:0] bits are cleared to 00_H and the CFEMP flag is set to 1.)

25.11.3.3 Procedure for Transmission from the Transmit Queue

Figure 25.33 shows the procedure for transmission from the transmit queue.



Note: p = m × 16 + 15, m = 0, 1

Figure 25.33 Procedure for Transmission from the Transmit Queue

25.11.3.4 Transmit History Buffer Reading Procedure

Transmit history data can be read from the RSCFDn(CFD)THLACCm register. The next data can be accessed by writing FF_H to the corresponding RSCFDn(CFD)THLPCTRm register (m = 0, 1) after reading a set of data. Figure 25.34 shows the transmit history buffer reading procedure.

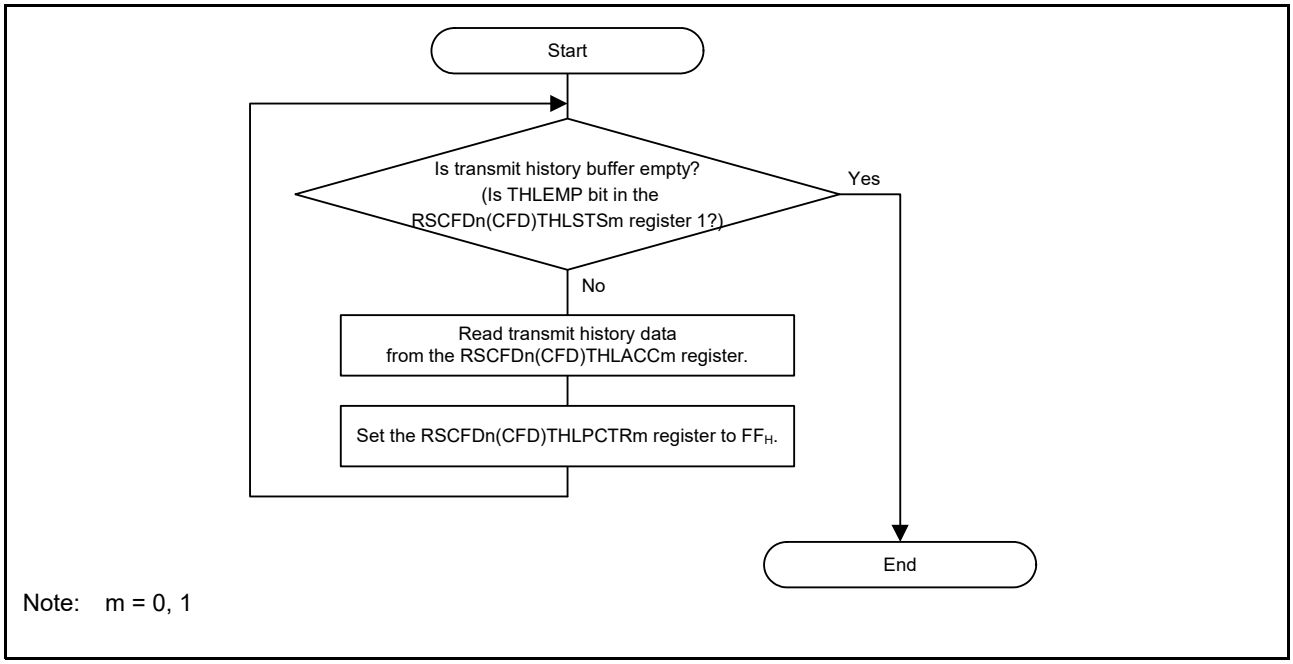


Figure 25.34 Transmit History Buffer Reading Procedure

25.11.4 Test Settings

25.11.4.1 Self-Test Mode Setting Procedure

Self-test mode allows communication test on a channel basis by enabling a CAN node to receive its own transmitted messages.

Figure 25.35 shows the self-test mode setting procedure.

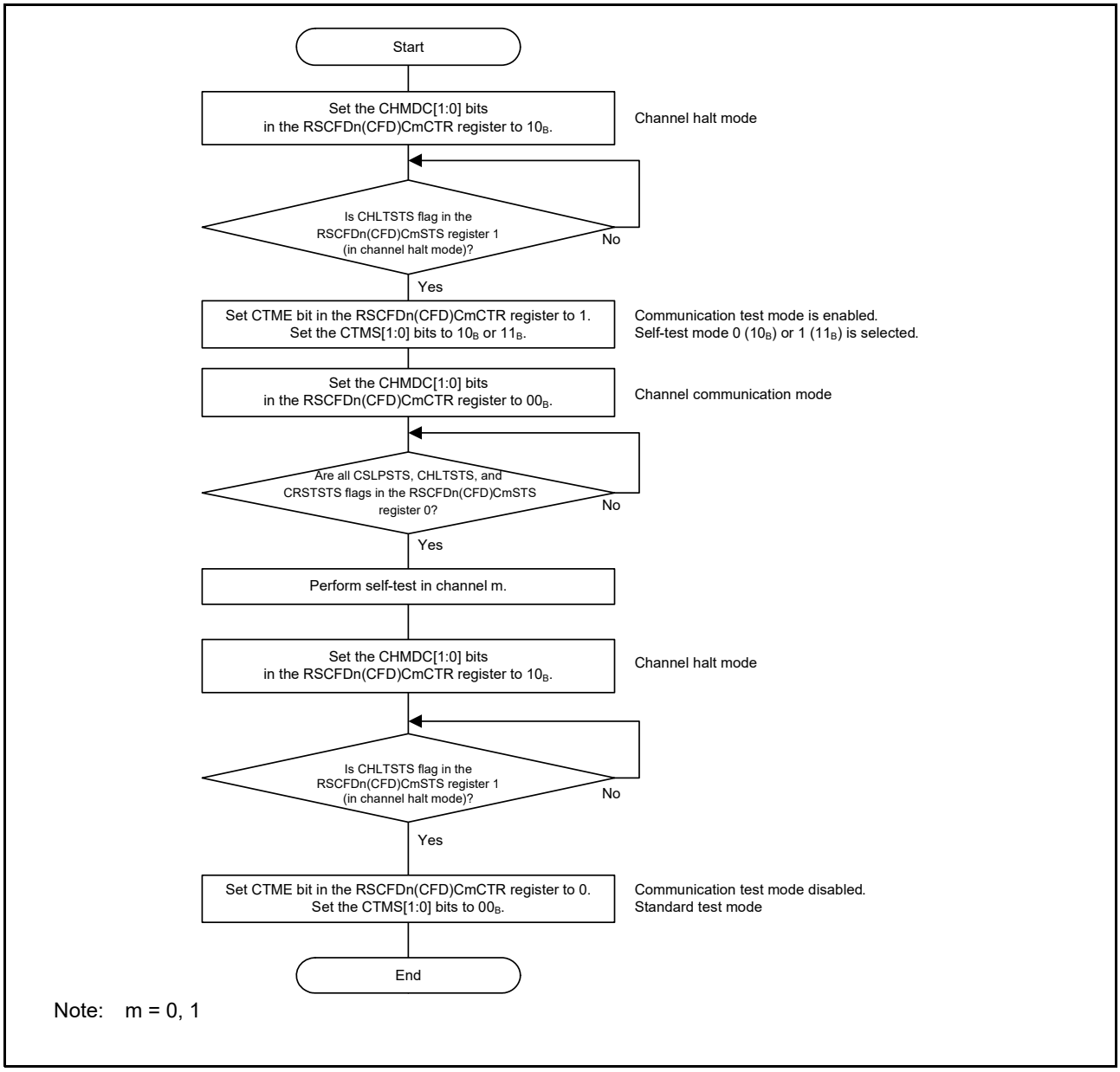


Figure 25.35 Self-Test Mode Setting Procedure

25.11.4.2 Procedure for Releasing the Protection

Since the global test function in Table 25.186 is protected, write the protection release data 1 and release data 2 in succession to the LOCK[15:0] bits in the RSCFDn(CFD)GLOCKK register, then set the target test bit to 1.

Table 25.186 Protection Release Data for Test Function

Test Function	Protection Release Data 1	Protection Release Data 2	Target Bit
RAM test	7575_H	$8A8A_H$	RTME bit in the RSCFDn(CFD)GTSTCTR register

If an incorrect value is written to the LOCK[15:0] bits, restart from writing the protection release data 1. Figure 25.36 shows the procedure for releasing the protection.

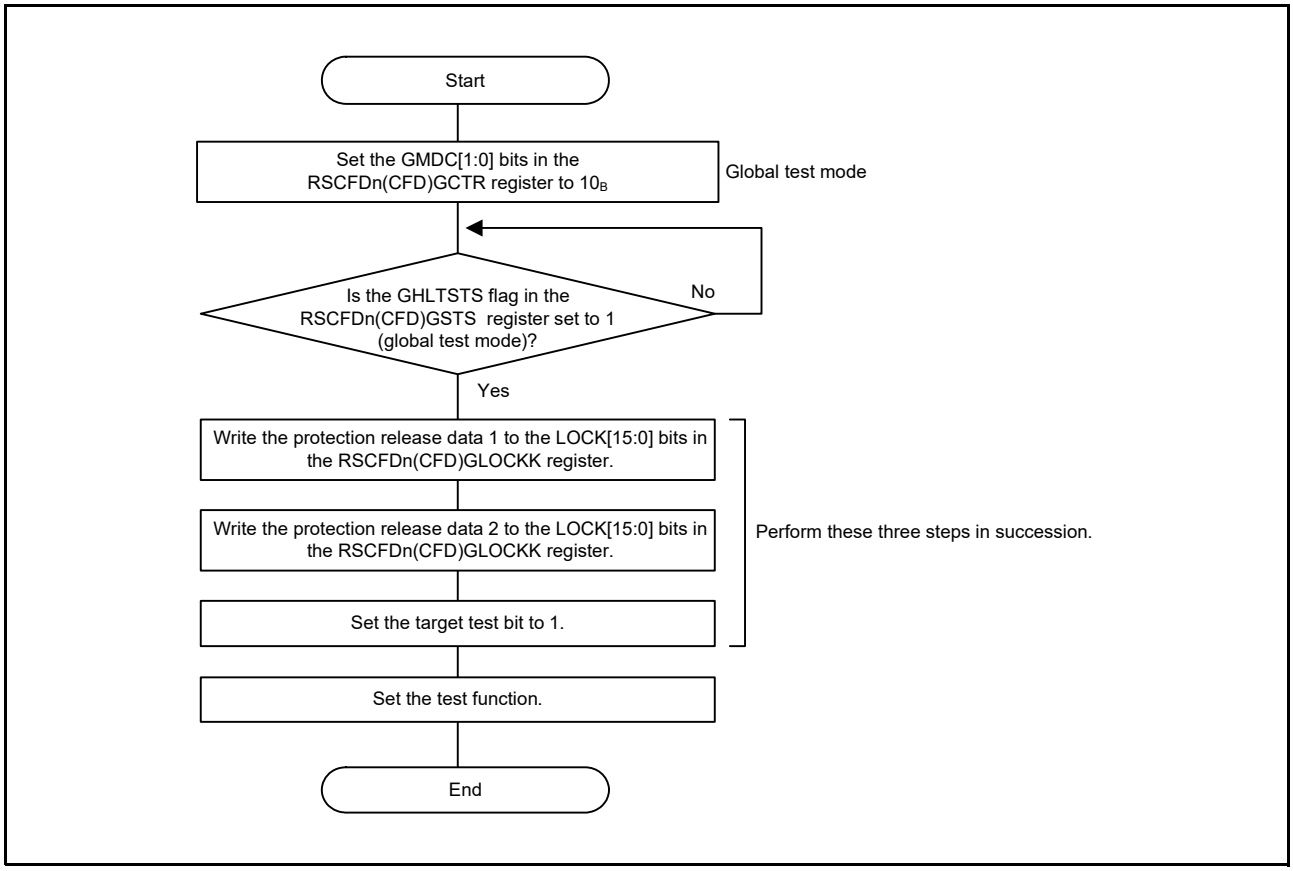
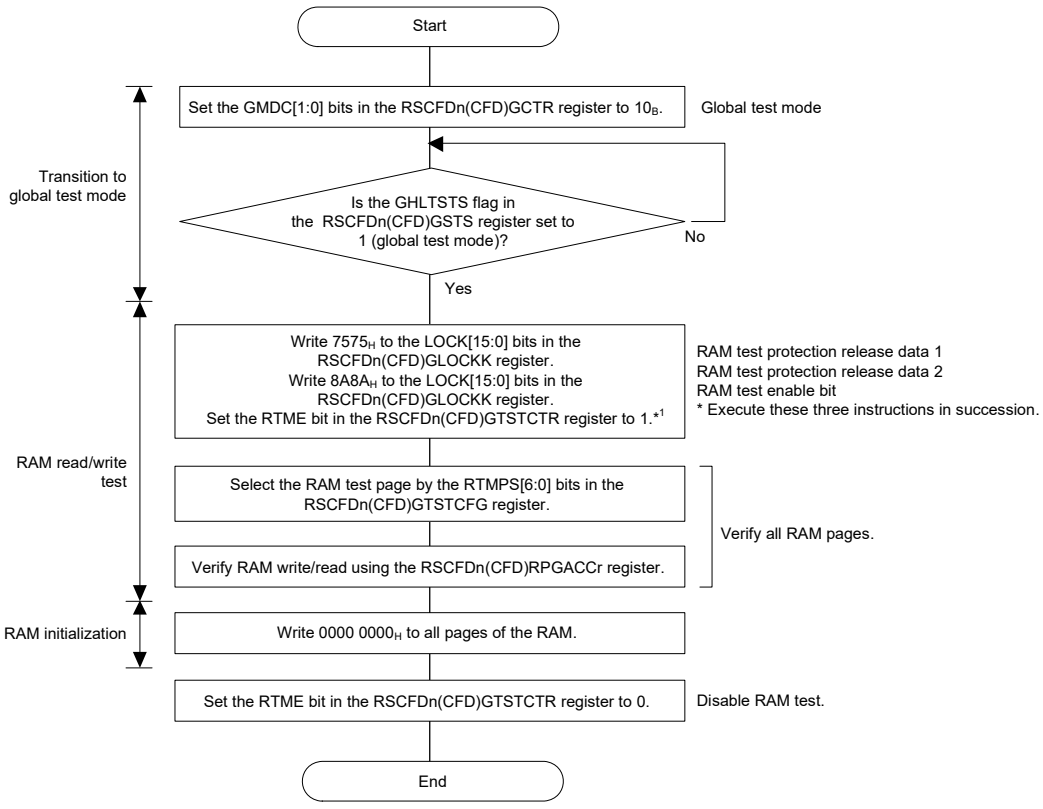


Figure 25.36 Protection Release Procedure

25.11.4.3 RAM Test Setting Procedure

RAM tests include CAN RAM read/write test. The read/write test verifies that data written to the RAM is read correctly. Before closing the RAM test, write 0000 0000_H to all pages of the CAN RAM.
Figure 25.37 shows the RAM test setting procedure.



Note 1. Before transitioning to the RAM test, perform the following.
- Cancel transmission request.
- Disable of each FIFO and transmission queue
- Clear flag of receive buffer

Note: r = 0 to 63

Figure 25.37 RAM Test Setting Procedure

25.11.4.4 Inter-Channel Communication Test Setting Procedure

Communication testing can be performed by transmitting and receiving data between different channels.
Figure 25.38 shows the inter-channel communication test setting procedure.

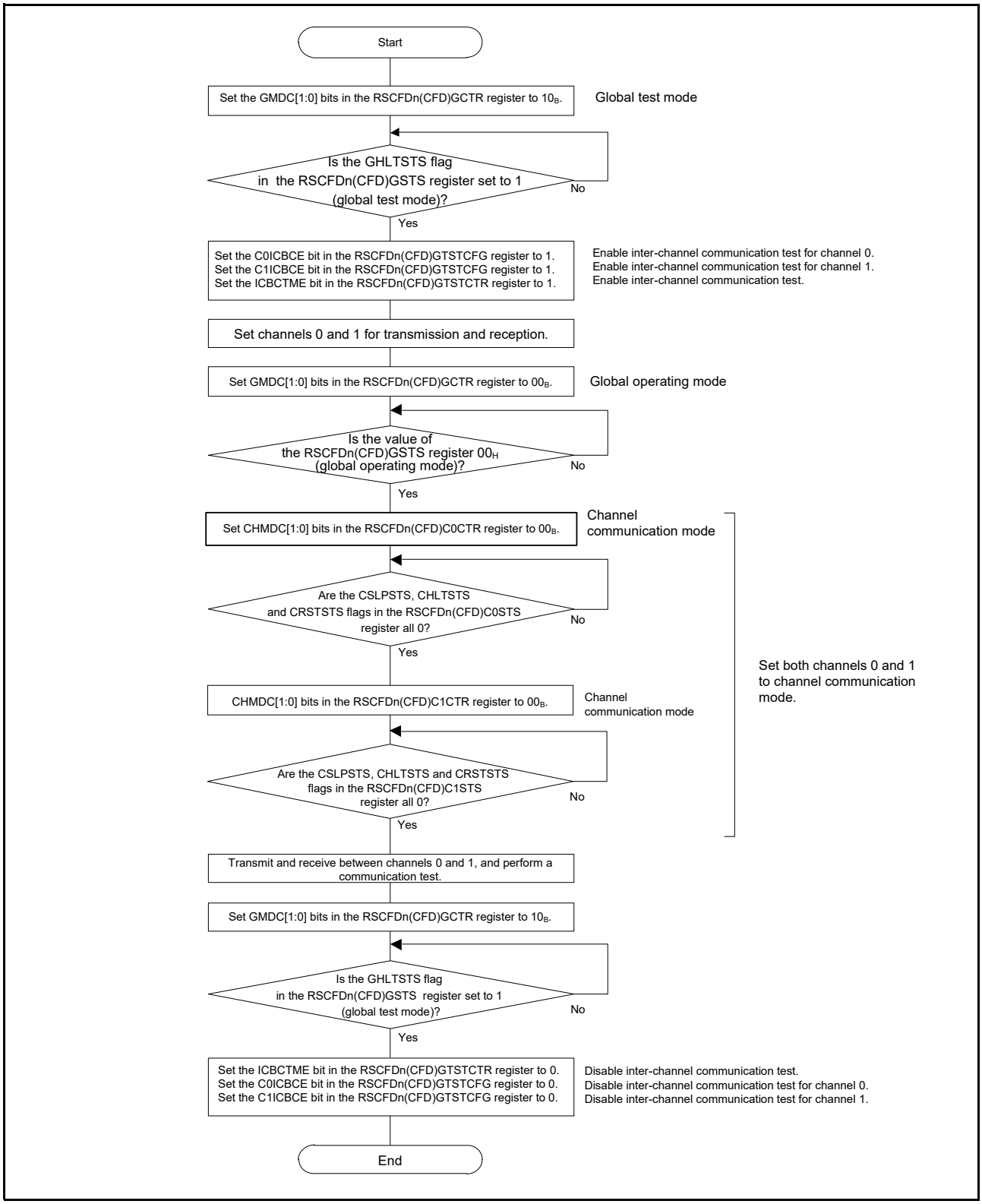


Figure 25.38 Inter-Channel Communication Test Setting Procedure (Example of Communication Test between Channel 0 and Channel 1)

25.12 Notes on the RS-CANFD Module

- When changing the interface mode without applying a power-on reset, write the value after reset to all registers and bits that are not allocated to the register map after change and then modify the RSCFDnCFDGRMCFG register.
- When changing a global mode, check the GSLPSTS, GHLTSTS, and GRSTSTS flags in the RSCFDn(CFD)GSTS register for transitions. When changing a channel mode, check the CSLPSTS, CHLTSTS, and CRSTSTS flags in the RSCFDn(CFD)CmSTS register (m = 0, 1) for transitions.
- When only classical CAN frames are used in CANFD mode, set the RSCFDnCFDCmDCFG register to the value equal to the set RSCFDnCFDCmNCFG register value.
- The acceptance filter processing checks receive rules sequentially in ascending order from the minimum rule number. If the same ID, IDE bit, or RTR bit value is set for multiple receive rules, the minimum number of receive rule is used for the acceptance filter processing. If the message does not pass through the subsequent DLC filter processing, the data processing is terminated without returning to the acceptance filter processing and the message is not stored in the buffer.
- When linking transmit buffers to transmit/receive FIFO buffers or allocating transmit buffers to transmit queues, set the control register (RSCFDn(CFD)TMCp) of the corresponding transmit buffer to 00_H. The status register (RSCFDn(CFD)TMSTSp) of the corresponding transmit buffer should not be used. Flags in other status registers (registers RSCFDn(CFD)TMTRSTS0 to RSCFDn(CFD)TMTRSTS2, RSCFDn(CFD)TMTARSTS0 to RSCFDn(CFD)TMTARSTS2, RSCFDn(CFD)TMTCSTS0 to RSCFDn(CFD)TMTCSTS2, and RSCFDn(CFD)TMTASTS0 to RSCFDn(CFD)TMTASTS2), which correspond to transmit buffers linked to transmit/receive FIFO buffers or allocated to transmit queues remain unchanged. Set the enable bit in the corresponding interrupt enable register (registers RSCFDn(CFD)TMIEC0 to RSCFDn(CFD)TMIEC2) to 0 (transmit buffer interrupt is disabled).
- When using transmit buffer merge mode (in CANFD mode), write 00H to the control register (RSCFDn(CFD)TMCp) of the transmit buffer corresponding to the transmit buffer allocated as a payload storage area. Set the enable bit of corresponding interrupt enable registers (RSCFDn(CFD)TMIEC0 to RSCFDn(CFD)TMIEC2) to 0 (to disable interrupts).
- Transmit buffers that are linked to transmit/receive FIFO buffers must not bet allocated to transmit queues. Do not allocate a transmit buffer allocated as a payload storage area in transmit buffer merge mode (in CANFD mode) to the transmit queue either.
- Only a single transmit/receive FIFO buffer can be linked to a transmit buffer. Do not link two or more transmit/ receive FIFO buffers to transmit buffers of the same number.
- When the CANm bit time clock is selected as a timestamp counter clock source, the timestamp counter stops when the corresponding channel has transitioned to channel reset mode or channel halt mode.
- In case of an attempt to store a new received message when the receive FIFO buffer and the transmit/receive FIFO buffer are full, the new message is discarded. If you wish to store a new transmit message in the transmit/receive FIFO buffer or the transmit queue, check that the transmit/receive FIFO buffer or the transmit queue is not full.
- In the case of registers that access the RAM, the value after reset shown in section 25.3, Registers (Classical CAN Mode) and section 25.4, Registers (CANFD Mode) indicate the values cleared by initialization of the CAN RAM. Values before clear are undefined. The following registers apply.
  - Receive rule (RSCFDn(CFD)GAFLIDj, RSCFDn(CFD)GAFLMj, RSCFDn(CFD)GAFLP0_j, RSCFDn(CFD)GAFLP1_j registers)
  - Receive buffers (RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq, RSCFDn(CFD)RMDFb_q registers)
  - Receive FIFO buffer access registers (RSCFDn(CFD)RFIDx, RSCFDn(CFD)RFPTRx, RSCFDnCFDRFFDSTSx, and RSCFDn(CFD)RFDFd_x registers)
  - Transmit/receive FIFO buffer access registers (RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDn(CFD)CFDFd_k registers)

  - Transmit buffers (RSCFDn(CFD)TMIDp, RSCFDn(CFD)TMPTRp, RSCFDnCFDTMFDCTRp, and RSCFDn(CFD)TMDFb_p registers)
  - Transmit history access register (RSCFDn(CFD)THLACCm registers)
  - RAM test page access register (RSCFDn(CFD)RPGACCr registers)
- The values of unused receive buffers (RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq, and RSCFDn(CFD)RMDFb_q registers), receive FIFO buffer access registers (RSCFDn(CFD)RFIDx, RSCFDn(CFD)RFPTRx, RSCFDnCFDRFFDSTSx, and RSCFDn(CFD)RFDFd_x registers) and transmit/receive FIFO buffer access registers (RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDn(CFD)CFDFd_k registers) are undefined when the RS-CANFD module transitions to global operation mode or global test mode after exiting from global reset mode.

26. Renesas SPDIF Interface

26.1 Overview

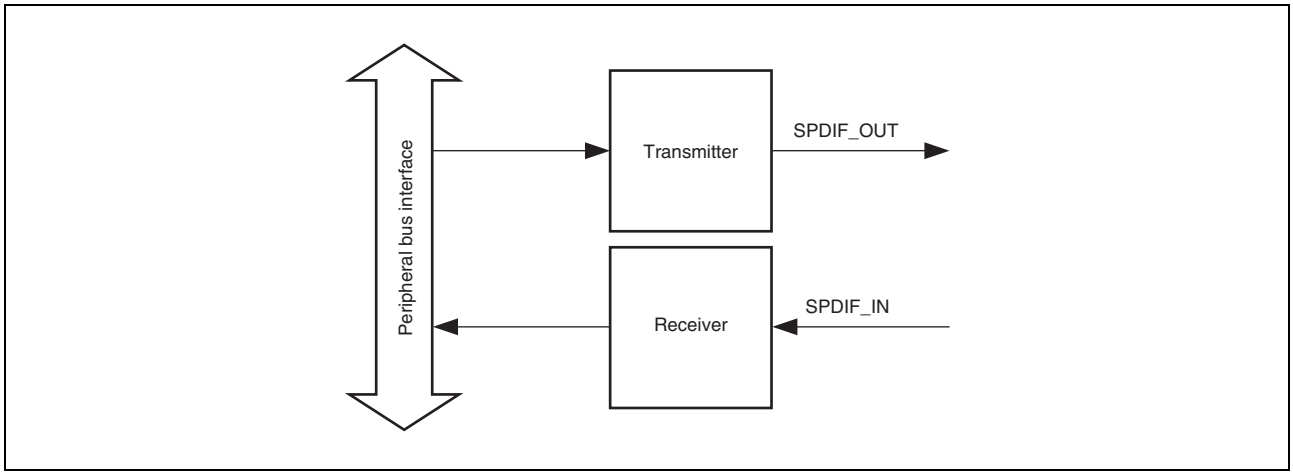


Figure 26.1 Overview Block Diagram

26.2 Features

- Supports the IEC 60958 standard (stereo and consumer use modes only).
- Supports sampling frequencies of 32 kHz, 44.1 kHz, and 48 kHz.
- Supports audio word sizes of 16 to 24 bits per sample.
- Biphase mark encoding.
- Double buffered data.
- Parity encoded serial data.
- Simultaneous transmit and receive
- Receiver autodetects IEC 61937 compressed mode data

26.3 Functional Block Diagram

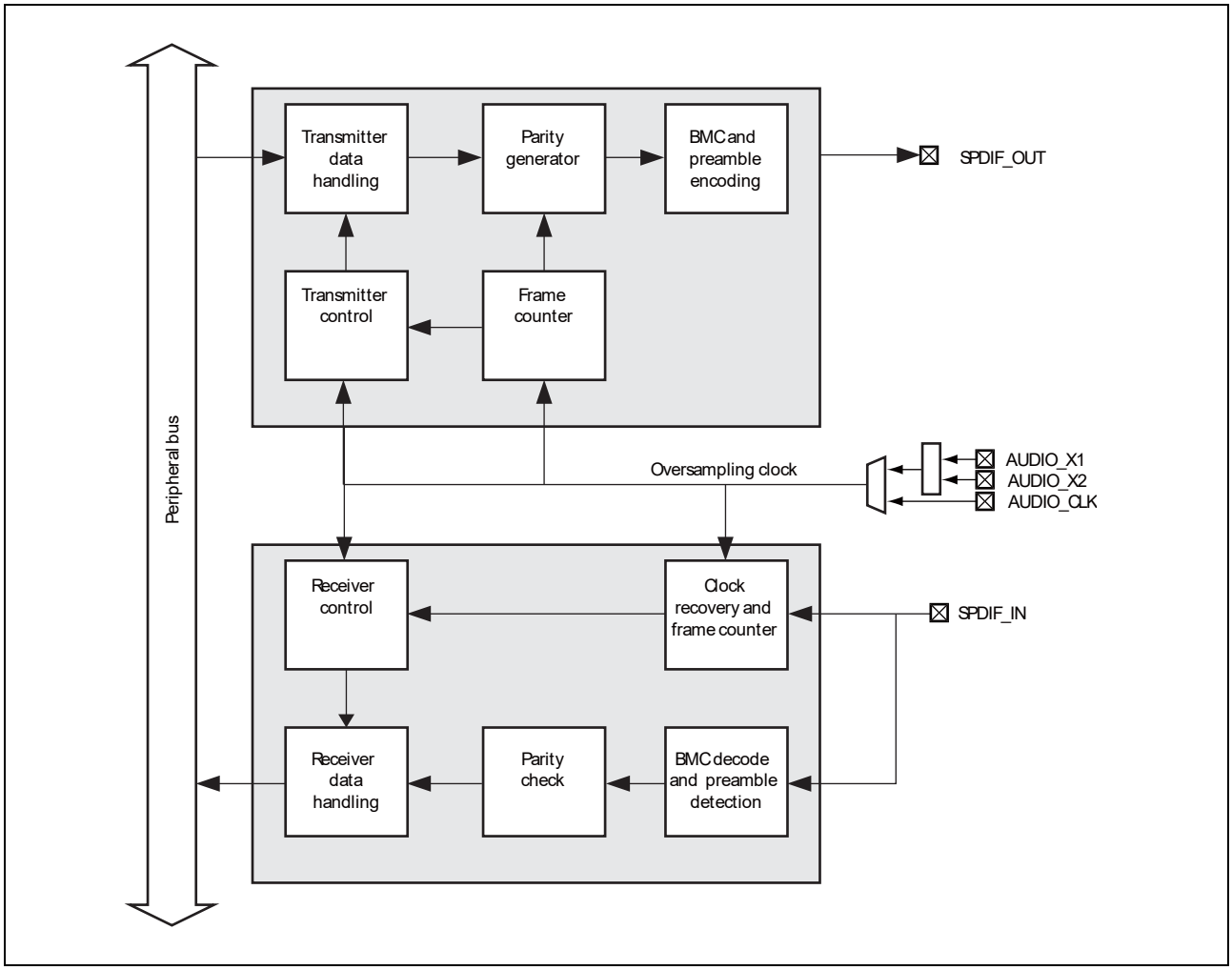


Figure 26.2 Functional Block Diagram

26.4 Input/Output Pins

Table 26.1 shows the pin configuration.

Table 26.1 Pin Configuration

Channel	Pin Name	I/O	Description
0	SPDIF_OUT	Output	Transmitter biphase-mark encoded SPDIF bitstream
1	SPDIF_IN	Input	Receiver biphase-mark encoded SPDIF bitstream
0, 1 (Common)	AUDIO_CLK	Input	External clock for audio
	AUDIO_X1	Input	Crystal resonator/external clock for audio
	AUDIO_X2	Output	

26.5 Renesas SPDIF (IEC60958) Frame Format

The Renesas SPDIF frame consists of two subframes (for channels 1 and 2), each of which contains a 4-bit preamble, audio data of up to 24 bits, a V flag, a user bit, a channel status bit, and an even parity bit. Figure 26.3 shows the subframe format. According to this format, the Renesas SPDIF performs biphase-mark modulation (channel coding) that will make the transmission line's DC component a minimum value.

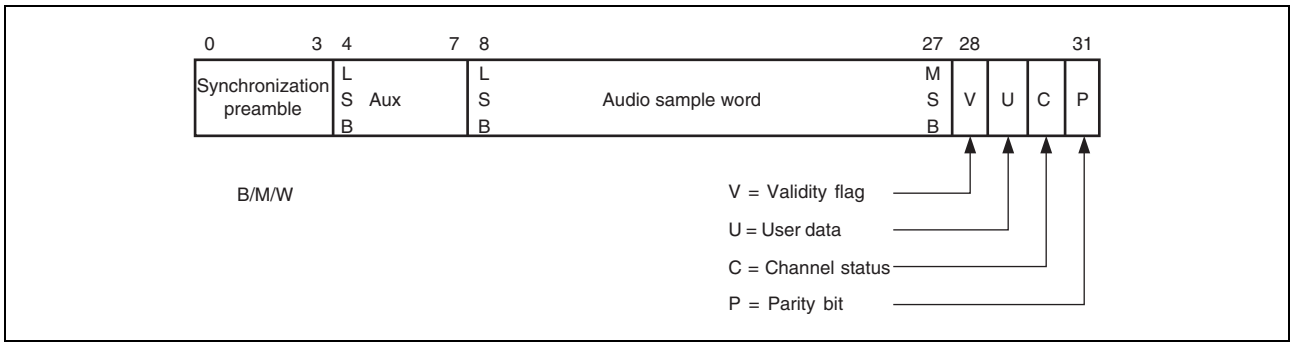


Figure 26.3 Subframe Format

Figure 26.4 shows the block format, which consists of 192 continuous frames. One block begins at the starting frame (preamble B) and ends at the 192nd frame (frame 191), and the preamble is used to identify all subframes. Each block has a total of 384 subframes, which are classified into three categories: subframe 0 indicating the beginning of a new block, subframe 1 (usually the channel 1), and subframe 2 (usually the channel 2). Usually, the music data sent and received by the SPDIF is continuous so that continuous blocks appear.

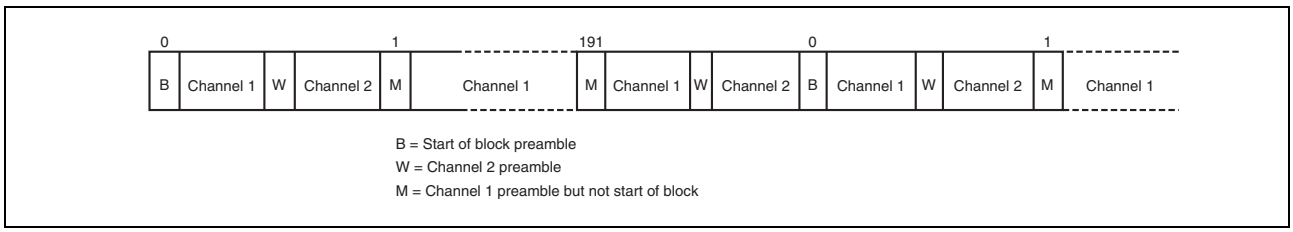


Figure 26.4 Block Format

Table 26.2 shows the binary values of the Renesas SPDIF preambles. The polarity of these preambles differs depending on the status of the preceding symbol (parity bit).

Table 26.2 Binary Preamble Values

Preamble	Preceding Symbol's Status = 0	Preceding Symbol's Status = 1
B	11101000	00010111
M	11100010	00011101
W	11100100	00011011

Note: As shown in Figure 26.3, the even parity bit at time slot 31 of a subframe determines the type of a preamble for one cycle of transmission. Usually, therefore, any one is selected from the set states that are sent through the Renesas SPDIF. However, IEC60958 requires decoding both types in view of connection with the preamble polarity reversed; the Renesas SPDIF has preambles decoded according to Table 26.2.

Channel status information is encoded at the rate of one bit per subframe, making the channel status information per block have a total of 192 bits for each of subframes 1 and 2. For the format of the channel status, refer to the IEC 60958 standard.

26.6 Register

Table 26.3 shows the register configuration.

Table 26.3 Register Configuration

Channel	Register Name	Abbreviation	Address	Access Size
0 (Transmit)	Transmitter channel 1 audio register	TLCA	H'E804 F000	32
	Transmitter channel 2 audio register	TRCA	H'E804 F004	32
	Transmitter channel 1 status register	TLCS	H'E804 F008	32
	Transmitter channel 2 status register	TRCS	H'E804 F00C	32
	Transmitter user data register	TUI	H'E804 F010	32
1 (Receive)	Receiver channel 1 audio register	RLCA	H'E804 F014	32
	Receiver channel 2 audio register	RRCA	H'E804 F018	32
	Receiver channel 1 status register	RLCS	H'E804 F01C	32
	Receiver channel 2 status register	RRCS	H'E804 F020	32
	Receiver user data register	RUI	H'E804 F024	32
0, 1 (Common)	Control register	CTRL	H'E804 F028	32
	Status register	STAT	H'E804 F02C	32
0, 1 (Common)	Transmitter DMA audio data register	TDAD	H'E804 F030	32
	Receiver DMA audio data register	RDAD	H'E804 F034	32

Note: All registers are longword registers and must be accessed as such.
A register diagram containing a 0 indicates that the write value should always be 0 (if the register is writable) and that the read value should always be 0 (if readable).

26.7 Register Descriptions

Legend:

Initial Value:	Register value after reset
—:	Undefined value
R/W:	Readable/writable register. The write value can be read.
R:	Read only register. The write value should always be 0.
R/WC0:	Readable/writable register. Writing 0 initializes the bit, but writing 1 is ignored.
R/WC1:	Readable/writable register. Writing 1 initializes the bit, but writing 0 is ignored.
W:	Write only register. Reading is prohibited. If this bit is reserved, the write value should always be 0.
—/W:	Write only, Read value undefined

26.7.1 Control Register (CTRL)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CKS	—	PB	RASS		TASS		RDE	TDE	NCSI	AOS	RME	TME
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	REIE	TEIE	UBOI	UBUI	CREI	PAEI	PREI	CSEI	ABOI	ABUI	RUII	TUII	RCSI	RCBI	TCSI	TCBI
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved
28	CKS	0	R/W	Oversampling clock select Selects oversampling clock supply source. 0: AUDIO_X1 1: AUDIO CLK
27	—	0	R	Reserved
26	PB	0	R/W	Pass Back Passes transmitter SPDIF output into SPDIF receiver in SPDIF module. 0: Pass Back disabled 1: Pass Back enabled
25, 24	RASS	All 0	R/W	Receiver Audio Sample Bit Size These bits Indicate the receiver audio sample bit size (16, 20, or 24 bits), for data alignment purposes. 00: 16-bit sample 01: 20-bit sample 10: 24-bit sample 11: Reserved
23, 22	TASS	All 0	R/W	Transmitter Audio Sample Bit Size These bits Indicate the transmitter audio sample bit size (16, 20, or 24 bits), for data alignment purposes. 00: 16-bit sample 01: 20-bit sample 10: 24-bit sample 11: Reserved
21	RDE	0	R/W	Receiver DMA Enable Enables DMA requests for the receiver. 0: Receiver DMA disabled 1: Receiver DMA enabled
20	TDE	0	R/W	Transmitter DMA Enable Enables the DMA requests for the transmitter. 0: Transmitter DMA disabled 1: Transmitter DMA enabled
19	NCSI	0	R/W	New Channel Status Information Set this bit to 1 when new channel status information to be corrected is in the transmitter. 0: New channel status information has not been in transmitter 1: New channel status information has been in transmitter
18	AOS	0	R/W	Audio Only Samples Clear this bit to 0 when audio channel 1 and channel 2 registers contain user information. When this bit is set to 1, all user bits are cleared to 0. 0: User information present 1: User information not present
17	RME	0	R/W	Receiver Module Enable Enables the receiver module. 0: Receiver module disabled 1: Receiver module enabled

Bit	Bit Name	Initial Value	R/W	Description
16	TME	0	R/W	Transmitter Module Enable Enables the transmitter module. 0: Transmitter module disabled 1: Transmitter module enabled
15	REIE	0	R/W	Receiver Error Interrupt Enable Enables the receiver error interrupts. 0: Receiver error interrupt disabled 1: Receiver error interrupt enabled
14	TEIE	0	R/W	Transmitter Error Interrupt Enable Enables the transmitter error interrupts. 0: Transmitter error interrupt disabled 1: Transmitter error interrupt enabled
13	UBOI	0	R/W	User Buffer Overrun Interrupt Enable Enables the user buffer overrun interrupts. 0: User buffer overrun interrupt disabled 1: User buffer overrun interrupt enabled
12	UBUI	0	R/W	User Buffer Underrun Interrupt Enable Enables the user buffer underrun interrupts. 0: User buffer underrun interrupt disabled 1: User buffer underrun interrupt enabled
11	CREI	0	R/W	Clock Recovery Error Interrupt Enable Enables the clock recovery error interrupts. 0: Clock recovery error interrupt disabled 1: Clock recovery error interrupt enabled
10	PAEI	0	R/W	Parity Error Interrupt Enable Enables the parity check error interrupts. 0: Parity check error interrupt disabled 1: Parity check error interrupt enabled
9	PREI	0	R/W	Preamble Error Interrupt Enable Enables the preamble check error interrupts. 0: Preamble error interrupt disabled 1: Preamble error interrupt enabled
8	CSEI	0	R/W	Channel Status Error Interrupt Enable Enables the channel status error interrupts. 0: Channel status error interrupt disabled 1: Channel status error interrupt enabled
7	ABOI	0	R/W	Audio Buffer Overrun Interrupt Enable Enables the receiver audio buffer overrun interrupts. 0: Audio buffer overrun interrupt disabled 1: Audio buffer overrun interrupt enabled
6	ABUI	0	R/W	Audio Buffer Underrun Interrupt Enable Enables the transmitter audio buffer underrun interrupts. 0: Audio buffer underrun interrupt disabled 1: Audio buffer underrun interrupt enabled
5	RUII	0	R/W	Receiver User Information Interrupt Enable Enables the receiver user information register full interrupts. 0: Receiver user information interrupt disabled 1: Receiver user information interrupt enabled
4	TUII	0	R/W	Transmitter User Information Interrupt Enable Enables the transmitter user information register empty interrupts. 0: Transmitter user information interrupt disabled 1: Transmitter user information interrupt enabled
3	RCSI	0	R/W	Receiver Channel Status Interrupt Enable Enables the receiver channel status register full interrupts. 0: Receiver channel status interrupt disabled 1: Receiver channel status interrupt enabled
2	RCBI	0	R/W	Receiver Channel Buffer Interrupt Enable Enables the receiver audio channel buffer full interrupts. 0: Receiver audio channel interrupt disabled 1: Receiver audio channel interrupt enabled

Bit	Bit Name	Initial Value	R/W	Description
1	TCSI	0	R/W	Transmitter Channel Status Interrupt Enable Enables the transmitter channel status register empty interrupts. 0: Transmitter channel status interrupt disabled 1: Transmitter channel status interrupt enabled
0	TCBI	0	R/W	Transmitter Channel Buffer Interrupt Enable Enables the transmitter audio channel buffer empty interrupts. 0: Transmitter audio channel interrupt disabled 1: Transmitter audio channel interrupt enabled

26.7.2 Status Register (STAT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RIS	TIS	UBO	UBU	CE	PARE	PREE	CSE	ABO	ABU	RUIR	TUIR	CSRX	CBRX	CSTX	CBTX
Initial value:	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved
16	CMD	0	R	Compressed Mode Data Sets if the data being received is compressed mode data (When bit 1 = 1 in the V flag and channel status). 0: Data is not in compressed mode 1: Data is in compressed mode
15	RIS	1	R	Receiver Idle State Sets if the receiver is in the idle state. 0: Receiver is not in idle state 1: Receiver in idle state
14	TIS	1	R	Transmitter Idle State Sets if the transmitter is in the idle state. 0: Transmitter is not in idle state 1: Transmitter is in idle state
13	UBO	0	R/WC0	User Buffer Overrun* Sets if the receiver user buffer overruns. This bit is cleared by writing 0 to the register. If bit REIE and bit UBOI in the control register are set this causes an interrupt. 0: User buffer has not overrun 1: User buffer has overrun
12	UBU	0	R/WC0	User Buffer Underrun* Sets if the transmitter user buffer underrun. This bit is cleared by writing 0. If bits TEIE and UBUI in the control register are set this causes an interrupt. 0: User buffer has not underrun 1: User buffer has underrun
11	CE	0	R/WC0	Clock Error* Sets when the clock recovery falls out of synchronization. This bit is cleared by writing 0. If bits REIE and CREI in the control register are set this causes an interrupt. 0: Clock recovery stable 1: Clock recovery error
10	PARE	0	R/WC0	Parity Error* Sets when the parity checker produces a fail result. This bit is cleared by writing 0. If bits REIE and PAEI in the control register are set this causes an interrupt. 0: Parity check correct 1: Parity error
9	PREE	0	R/WC0	Preamble Error* Sets when the start of word preamble fails to appear in the correct place. This bit is cleared by writing 0. If bits REIE and PREI in the control register are set this causes an interrupt. Note: Only set after a start of block preamble has occurred. 0: Preamble is in the correct place 1: Preamble error

Bit	Bit Name	Initial Value	R/W	Description
8	CSE	0	R/WC0	Channel Status Error*1 Sets when the channel status information is written before the 32nd frame of the current block. This bit is cleared by writing 0. If bits TEIE and CSEI in the control register are set this causes an interrupt. 0: Channel status correct 1: Channel status error
7	ABO	0	R/WC0	Audio Buffer Overrun*1 Indicates that the receiver audio buffer is full in both the first and second stages and that data has been overwritten. This bit is cleared by writing 0. If bits REIE and ABOI in the control register are set then this causes an interrupt. 0: Receiver audio buffer has not overrun 1: Receiver audio buffer has overrun
6	ABU	0	R/WC0	Audio Buffer Underrun*1 Indicates that the transmitter audio buffer is empty in both the first and second stages and that the last data transmission has been repeated. This bit is cleared by writing 0. If bits TEIE and ABUI in the control register are set then this causes an interrupt. 0: Transmitter audio buffer has not underrun 1: Transmitter audio buffer has underrun
5	RUIR	0	R	Receiver User Information Register Status Indicates the status of the receiver user information register. This bit is cleared by reading from the receiver user register. If bit RUII in the control register is set then this causes an interrupt. 0: Receiver user information register is empty 1: Receiver user information register is full
4	TUIR	0	R	Transmitter User Information Register Status Indicates the status of the transmitter user information register. This bit is cleared by writing to the transmitter user register. If bit TUII in the control register is set then this causes an interrupt. 0: Transmitter user information register is full 1: Transmitter user information register is empty
3	CSRX	0	R	Channel 1 and Channel 2 Status for Receiver Indicates the status of the receiver channel status registers. This bit is cleared by reading from the receiver channel status registers. If bit RCSI in the control register is set this causes an interrupt. 0: Receiver channel status registers are empty 1: Receiver channel status registers are full
2	CBRX	0	R	Channel 1 and Channel 2 Buffers for Receiver Indicates the status of the receiver audio channel registers. This bit is cleared by reading from the receiver audio channel registers. If bit RCBI in the control register is set this causes an interrupt. 0: Receiver audio channel registers are empty 1: Receiver audio channel registers are full
1	CSTX	0	R	Channel 1 and Channel 2 Status for Transmitter Indicates the status of the transmitter channel status registers. This bit is cleared by writing to the transmitter channel status registers. If bit TCSI in the control register is set this causes an interrupt. 0: Transmitter channel status register is full 1: Transmitter channel status register is empty
0	CBTX	0	R	Channel 1 and Channel 2 Buffers for Transmitter Indicates the status of the transmitter audio channel registers. This bit is cleared by writing to the transmitter audio channel registers. If bit TCBI in the control register is set this causes an interrupt. 0: Transmitter audio channel registers are full 1: Transmitter audio channel registers are empty

Note 1. When an error bit is detected during DMA transfer, DMA transfer settings must be made again. In this case, the Renesas SPDIF's module enable bit (either the RME or TME bit) and the DMA enable bit (either the RDE or TDE bit) must be disabled and the error status must be cleared before making DMA transfer settings again. Then the module enable bit should be set and DMA transfer can be started again.

26.7.3 Transmitter Channel 1 Audio Register (TLCA)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	Audio PCM Data							
Initial value:	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Audio PCM Data															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	W	Reserved
23 to 0	Audio PCM Data	All 0	W	Audio PCM Data LSB aligned PCM encoded audio data.

26.7.4 Transmitter Channel 2 Audio Register (TRCA)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	Audio PCM Data							
Initial value:	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Audio PCM Data															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	W	Reserved
23 to 0	Audio PCM Data	All 0	W	Audio PCM Data LSB aligned PCM encoded audio data.

26.7.5 Transmitter DMA Audio Data Register (TDAD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	Audio PCM Data							
Initial value:	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Audio PCM Data															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	W	Reserved
23 to 0	Audio PCM Data	All 0	W	Audio PCM Data LSB aligned PCM encoded audio data.

26.7.6 Transmitter User Data Register (TUI)

U-bit data in subframes is written in to this register. Because U-bit data is transmitted in a sequence of subframes 1 and 2, you need to update the data on a 16-frame basis. For the contents of the user bytes refer to the appropriate standard for the device in use. The user bits to be transmitted are set in sequence starting at the LSB.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	User Byte 4								User Byte 3							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	User Byte 2								User Byte 1							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	User Byte 4	All 0	W	U-bit information is stored here.
23 to 16	User Byte 3	All 0	W	
15 to 8	User Byte 2	All 0	W	
7 to 0	User Byte 1	All 0	W	

26.7.7 Transmitter Channel 1 Status Register (TLCS)

The 30-bit register stores the channel status information to be transmitted. For each channel, channel status information per frame consists of 192 bits. Because necessary data covers only the 30 bits that are set in the following register, zeros continue to be sent after the transmission of the first 30 bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CLAC[1:0]		FS[3:0]				CHNO[3:0]				SRCNO[3:0]			
Initial value:	—	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATCD[7:0]								—	—	CTL[4:0]					—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	W	Reserved
29, 28	CLAC[1:0]	All 0	W	Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Reserved
27 to 24	FS[3:0]	All 0	W	Sample Frequency (FS) 0000: 44.1 kHz 0010: 48 kHz 0011: 32 kHz
23 to 20	CHNO[3:0]	All 0	W	Channel Number 0000: Don't care 0001: A (left channel) 0010: B (right channel) 0011: C
19 to 16	SRCNO[3:0]	All 0	W	Source Number 0000: Don't care 0001: 1 0010: 2 0011: 3
15 to 8	CATCD[7:0]	All 0	W	Category Code (Example) 00000000: 2-channel general format 00000001: 2-channel compact disc (IEC 908) 00000010: 2-channel PCM encoder/decoder 00000011: 2-channel digital audio tape recorder
7, 6	—	All 0	W	Reserved The write value should always be 0.
5 to 1	CTL[4:0]	All 0	W	Control The control bits are copied from the source (see IEC60958 standard).
0	—	0	W	Reserved The write value should always be 0.

26.7.8 Transmitter Channel 2 Status Register (TRCS)

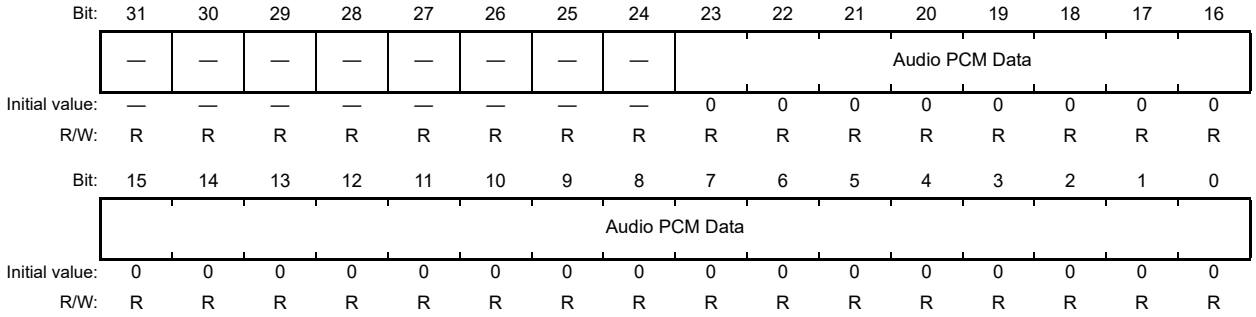
The 30-bit register stores the channel status information to be transmitted. For each channel, channel status information per frame consists of 192 bits. Because necessary data covers only the 30 bits that are set in the following register, zeros continue to be sent after the transmission of the first 30 bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CLAC[1:0]		FS[3:0]				CHNO[3:0]				SRCNO[3:0]			
Initial value:	—	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATCD[7:0]								—	—	CTL[4:0]					—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

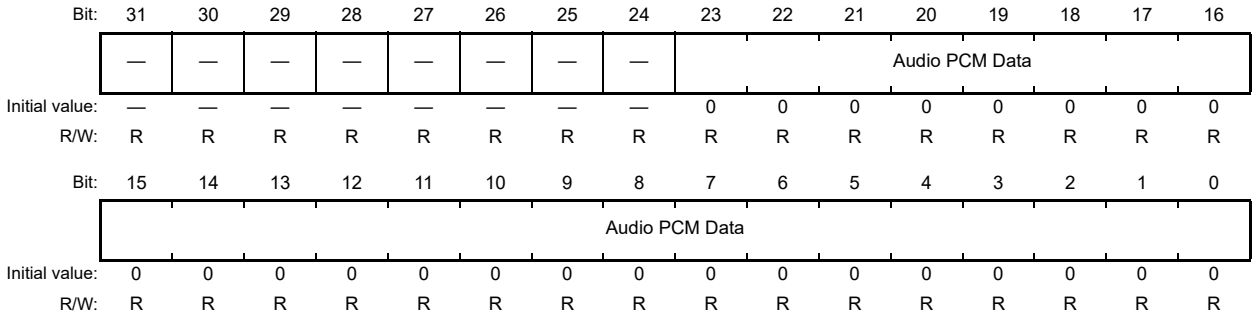
Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	W	Reserved
29, 28	CLAC[1:0]	All 0	W	Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Reserved
27 to 24	FS[3:0]	All 0	W	Sample Frequency (FS) 0000: 44.1 kHz 0010: 48 kHz 0011: 32 kHz
23 to 20	CHNO[3:0]	All 0	W	Channel Number 0000: Don't care 0001: A (left channel) 0010: B (right channel) 0011: C
19 to 16	SRCNO[3:0]	All 0	W	Source Number 0000: Don't care 0001: 1 0010: 2 0011: 3
15 to 8	CATCD[7:0]	All 0	W	Category Code (Example) 00000000: 2-channel general format 00000001: 2-channel compact disc (IEC 908) 00000010: 2-channel PCM encoder/decoder 00000011: 2-channel digital audio tape recorder
7, 6	—	All 0	W	Reserved The write value should always be 0.
5 to 1	CTL[4:0]	All 0	W	Control The control bits are copied from the source (see IEC60958 standard).
0	—	0	W	Reserved The write value should always be 0.

26.7.9 Receiver Channel 1 Audio Register (RLCA)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	R	Reserved
23 to 0	Audio PCM Data	All 0	R	Audio PCM Data LSB aligned PCM encoded audio data.

26.7.10 Receiver Channel 2 Audio Register (RRCA)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	R	Reserved
23 to 0	Audio PCM Data	All 0	R	Audio PCM Data LSB aligned PCM encoded audio data.

26.7.11 Receiver DMA Audio Data (RDAD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	Audio PCM Data							
Initial value:	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Audio PCM Data															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	R	Reserved
23 to 0	Audio PCM Data	All 0	R	Audio PCM Data LSB aligned PCM encoded audio data.

26.7.12 Receiver User Data Register (RUI)

The register stores the U-bit data received through the Renesas SPDIF. Because U-bit data is stored in a sequence of subframes 1 and 2 starting at the LSB, you need to read the data on a 16-frame basis. For the contents of the user bytes refer to the appropriate standard for the device in use.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	User Byte 4								User Byte 3							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	User Byte 2								User Byte 1							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	User Byte 4	All 0	R	U-bit information is stored here.
23 to 16	User Byte 3	All 0	R	
15 to 8	User Byte 2	All 0	R	
7 to 0	User Byte 1	All 0	R	

26.7.13 Receiver Channel 1 Status Register (RLCS)

The channel status is stored starting at the register's LSB in a way that subframe 1 received from the beginning of the block is stored. For the contents of the channel status register, refer to the IEC-60958 standard.

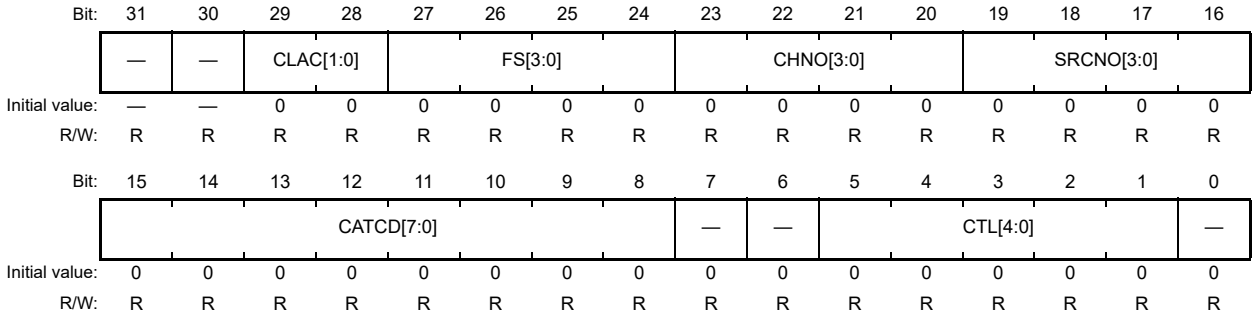
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CLAC[1:0]		FS[3:0]				CHNO[3:0]				SRCNO[3:0]			
Initial value:	—	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATCD[7:0]								—	—	CTL[4:0]					—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	R	Reserved
29, 28	CLAC[1:0]	All 0	R	Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Reserved
27 to 24	FS[3:0]	All 0	R	Sample Frequency (fS) 0000: 44.1 kHz 0010: 48 kHz 0011: 32 kHz
23 to 20	CHNO[3:0]	All 0	R	Channel Number 0000: Don't care 0001: A (left channel) 0010: B (right channel) 0011: C
19 to 16	SRCNO[3:0]	All 0	R	Source Number 0000: Don't care 0001: 1 0010: 2 0011: 3
15 to 8	CATCD[7:0]	All 0	R	Category Code (Example) 00000000: 2-channel general format 00000001: 2-channel compact disc (IEC 908) 00000010: 2-channel PCM encoder/decoder 00000011: 2-channel digital audio tape recorder
7, 6	—	All 0	R	Reserved
5 to 1	CTL[4:0]	All 0	R	Control The control bits are copied from the source (see IEC60958 standard).
0	—	0	R	Reserved

26.7.14 Receiver Channel 2 Status Register (RRCS)

The channel status is stored starting at the register's LSB in a way that subframe 2 received from the beginning of the block is stored. For the contents of the channel status register, refer to the IEC-60958 standard.



Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	R	Reserved
29, 28	CLAC[1:0]	All 0	R	Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Reserved
27 to 24	FS[3:0]	All 0	R	Sample Frequency (fS) 0000: 44.1 kHz 0010: 48 kHz 0011: 32 kHz
23 to 20	CHNO[3:0]	All 0	R	Channel Number 0000: Don't care 0001: A (left channel) 0010: B (left channel) 0011: C
19 to 16	SRCNO[3:0]	All 0	R	Source Number 0000: Don't care 0001: 1 0010: 2 0011: 3
15 to 8	CATCD[7:0]	All 0	R	Category Code (Example) 00000000: 2-channel general format 00000001: 2-channel compact disc (IEC 908) 00000010: 2-channel PCM encoder/decoder 00000011: 2-channel digital audio tape recorder
7, 6	—	All 0	R	Reserved
5 to 1	CTL[4:0]	All 0	R	Control The control bits are copied from the source (see IEC60958 standard).
0	—	0	R	Reserved

26.8 Functional Description—Transmitter

26.8.1 Transmitter Module

The transmitter module transmits PCM data and auxiliary information after encoding it according to the method of biphase-mark modulation that complies with the IEC60958 standard (SPDIF).
The clock for the transmitter module is an oversampling clock supplied from the outside. This clock usually selects a value that serves as an oversample at a frequency eight times larger than the clock frequency required for biphase-mark encoding. In this case, the clock frequency required to transmit 32 time slots in a subframe is 512 times as large as the sample frequency for audio data.
Audio data and channel status information are first written into the module's channel 1 and then into channel 2.
Generally, the channel status need to be written only when the information changes. The SPDIF module requests that the channel status be written in 30 frames -- when all the current channel status data have been transmitted. You need to write somewhere between frame 31 and the beginning of the next block of 192 frames.
The audio data is stored in a double buffer arrangement. To make sure that the first stage buffer is empty, you can send an interrupt request or poll the status register. DMA transfers send channel 1 audio data on the first request and channel 2 data on the second.
The channel status information is stored in the 30-bit registers of channels 1 and 2. For each channel, the channel status information per frame consists of 192 bits. Because necessary data covers only 30 bits, zeros continue to be sent after the transmission of the first 30 bits until the block is completed.
User data forms a 32-bit double buffer arrangement. You can make sure that the first stage buffer is empty by either sending an interrupt request or polling the status register. Usually, information about the user data will become insufficient with the length of data between blocks. Transmission takes place in a sequence of channels 1 and 2. For the user data within a block, 384 bits are transmitted before the next block is continuously transmitted.
The audio data handled by the Renesas SPDIF module is a linear PCM, making it possible to set up to 24 bits. For this reason, the V flag indicating that audio data is a linear PCM remains to be 0. The V flag involves no register-based setting. An even parity is created for each 32 bits of serial output data (excluding the preamble).

Note: When transmitter user buffer underrun occurs, the current data in the buffer data of SPDIF is transmitted until the next data is filled.

26.8.2 Transmitter Module Initialization

The device defaults to an idle state when it comes out of reset, or can be put into an idle state when 0 is written to the TME bit in the CTRL register. When the transmitter module is idle, it has the following settings:

- The transmitter idle status bit (TIS) is set to 1, all other status bits are cleared to 0.
- Preamble generation is invalid.
- Synchronization between channels 1 and 2 is set to 0 (0 for channel 1, 1 for channel 2).
- Both word_count and frame_count are set to 0.
- The output from the biphase-mark encoder is set to 0.

Channel status, user and audio data registers will retain its value prior to putting the module into idle. To exit the idle state the user must write 1 to the TME bit in the CTRL register.

26.8.3 Initial Settings for Transmitter Module

When the TME bit is set to 1, the TUIR and CSTX bits are set to 1. After that, if data is written in the order of 1) TUI and 2) TLCS and TRCS, a channel status error will occur. To avoid this, be sure to write data in the order of 1) TLCS and TRCS and 2) TUI.
Before writing the first audio data (write access to TLCA or TRCA by the CPU or write access to TDAD by the DMA transfer) after setting the TME bit to 1, be sure to check that the CSTX and TUIR bits are cleared by writing to TLCS, TRCS, and TUI.

26.8.4 Transmitter Module Data Transfer

Once the transmitter module has left the idle state, it is ready for data transfer. Data transfer timing can be achieved in three ways. Either the transfer is done by interrupts, DMA requests or by polling the status register. There is a shared interrupt line (for both transmit and receive) and a single transmitter DMA request line.
Figure 26.5 shows a data transfer with an interrupt for the transmitter.

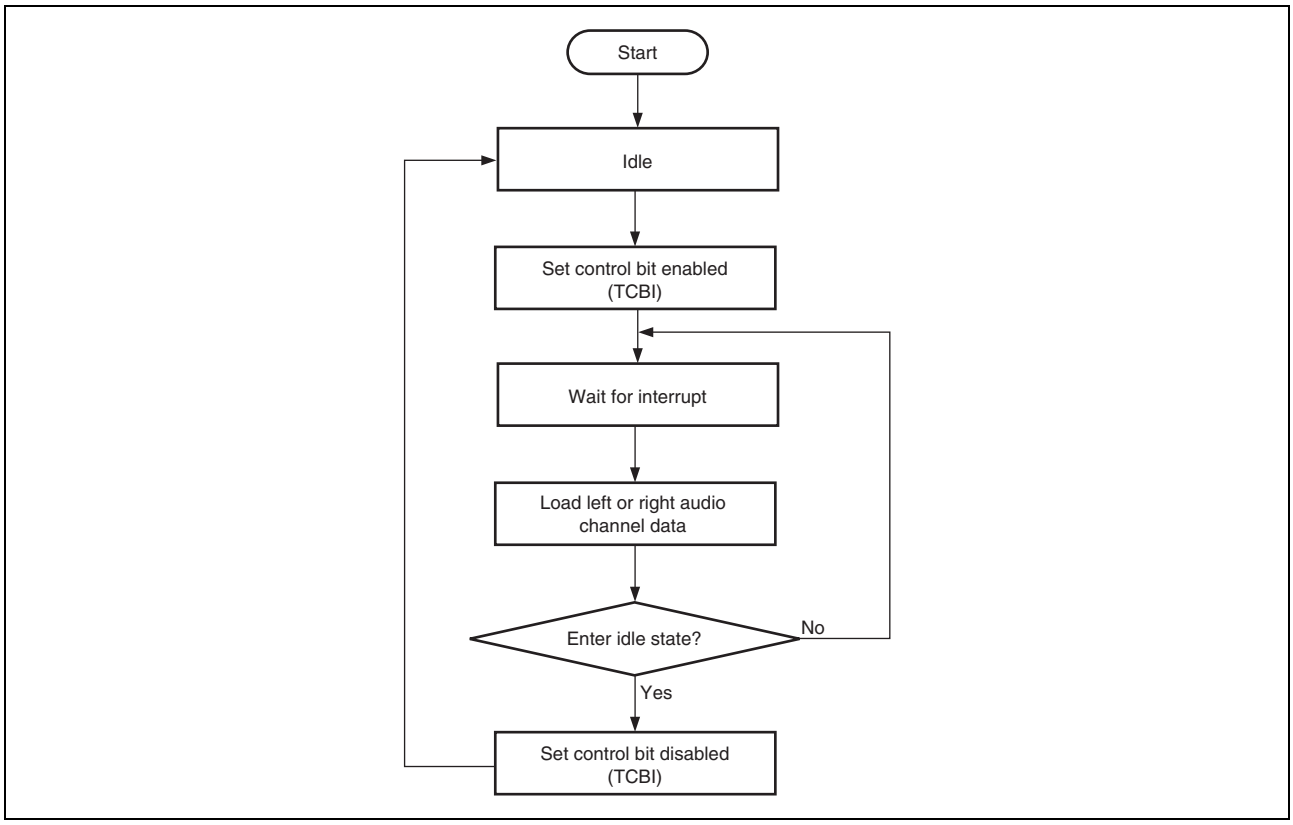


Figure 26.5 Transmitter Data Transfer Flow Diagram - Interrupt Driven

Figure 26.6 shows a data transfer with a DMA transfer for the transmitter.

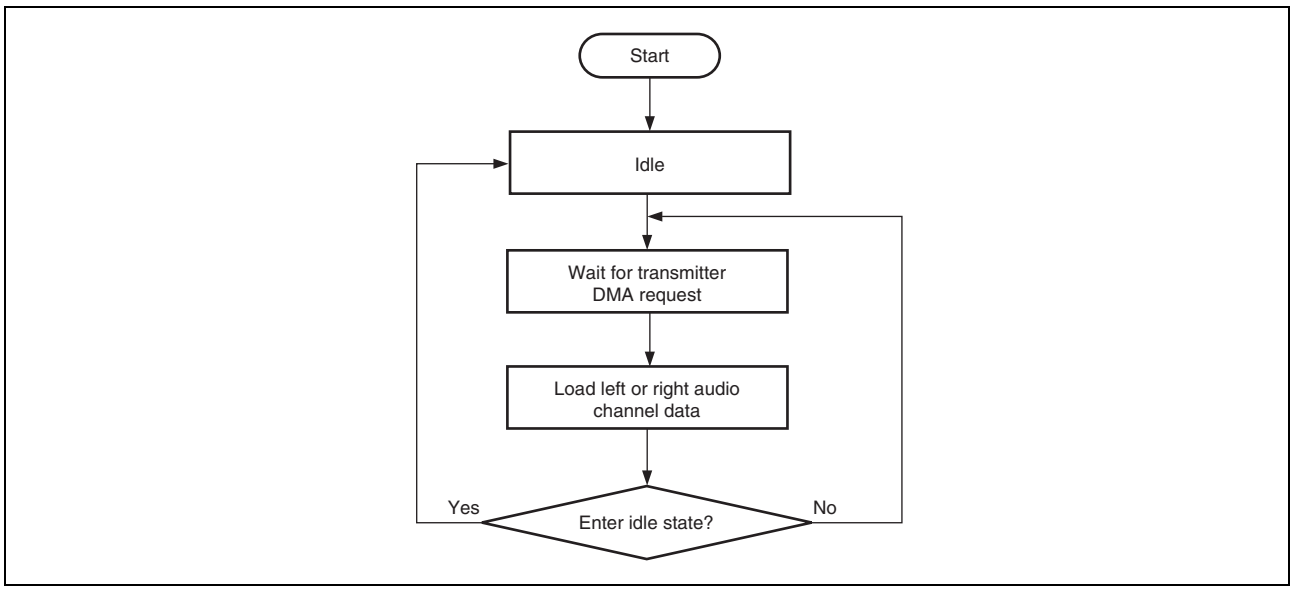


Figure 26.6 Transmitter Data Transfer Flow Diagram—DMA Request Driven

Channel status information is required to be updated when the information has changed. Because the updating needs to be done before the transmission of the next block, the channel status to be updated should be written after 30 frames have been sent; this is indicated either by an interrupt or by polling the status bit. If channel status is written before 30 frames have been sent (while current information is being sent) then an interrupt indicates that the channel status error bit (CSE) in the status register has been set.

Note: 30 frames contains all the valid information in a single channel status block.

26.9 Functional Description—Receiver

26.9.1 Receiver Module

The receiver module demodulates data and clock signals from the input encoded according to the IEC60958 standard. The encoded data, shown in linear PCM format, is stored into the audio data register. The register also stores the channel status and user information being received simultaneously as auxiliary information.
The main clock for the receiver module is an oversampling clock supplied from the outside. The module operates at a frequency four times as large as the oversampling clock.

Note: The oversampling clock is the same for the transmitter and receiver.

Clock recovery is performed using a pulse width counter and averaging filters to produce a sampling pulse in the middle of each bit in the datastream. A clock error status bit indicates clock synchronization loss. Synchronization is achieved when a preamble occurs on the data stream for the first time. Continuous adjustment prevents jitter and/or clock drift from affecting clock recovery, provided that they fall within the clock recovery specifications.
Once the clock recovery is successful the biphase-mark decoder initiates its preamble detection. The decoder searches for the start of block preamble (see Table 26.2). A preamble error status bit indicates that following preambles have not appeared at the correct time, such failures are most likely caused by transmission loss or interference.
Even parity checking is performed on the decoded data. A discrepancy will result in the parity error status bit being set.
The SPDIF module acquires user data and channel status information in addition to audio data. The audio is stored in a double buffer arrangement. Either an interrupt request because of a full buffer or polling of the status bit will indicate when the data is ready to be read. DMA transfers receive channel 1 audio data on the first request and channel 2 data on the second.
Channel status is stored in a 30-bit register. Channel status information is received at 1-bit per subframe. Therefore the registers will not be full until a total of 30 frames for each channel have been received. New channel status is compared with the current data to see if it has changed and is only read by the processor if it has. User data, which is also received at the same time, is stored into the register on a subframe basis, so that the reception is completed when 16 frames are reached.

Note 1. Channel status data requests do not support DMA.
Note 2. When receiver user buffer overrun occurs, the current data in the buffer data of SPDIF is overwritten by the next incoming data from SPDIF interface.

26.9.2 Receiver Module Initialization

The device defaults to an idle state when it comes out of reset, or can be put into an idle state by writing 0 to bit RME in the CTRL register. Whilst idle the module has the following settings:

- The receiver idle status bit is set to 1, all other status bits are cleared to 0.
- Synchronization between channels 1 and 2 is set to 0 (0 for channel 1, 1 for channel 2).
- Both Word_count and frame_count are set to 0.

Channel status registers, user data registers and audio data registers will retain its value prior to putting the module into idle. To exit the idle state the user must write 1 to the bit RME in the CTRL register.

26.9.3 Receiver Module Data Transfer

Once the module has left the idle state it is ready for data transfer. Data transfer timing can be achieved in three ways. The transfer can be done by interrupts, or by polling the status register, or by DMA. There is a shared interrupt line (transmit and receive) and a single receiver DMA request line. Data transfer for the receiver can be interrupted by error signals caused by:

1. Clock recovery failure.
2. Transmission loss or interference – indicated by a preamble error.
3. Parity check failure.

Transmission loss or interference can cause the start of subframe or start of block preamble to be misplaced or not present.
Parity check failure occurs when the parity bit is incorrect, this can be caused by any of the above.

- Clock Recovery Deviation
  The receive margin for clock recovery is based on the following equation:

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5)\, F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100\%$$

where M = receive margin
N = oversampling rate
L = frame length = 33
D = duty cycle = 0.6
F = oversampling clock deviation = Level II accuracy = 1000 in $10e^{-6}$

Figure 26.7 indicates what the receive margin M represents

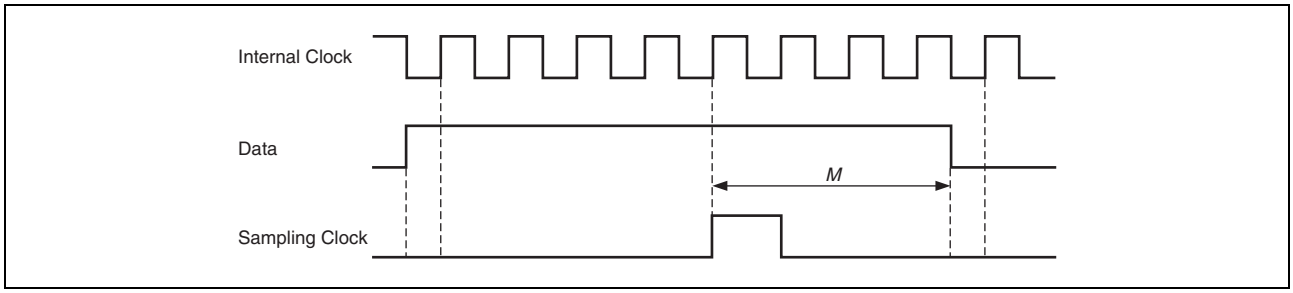


Figure 26.7 Receive Margin

Introducing jitter into the equation gives the following inequality.

$$j \leq \left| \left(0.5 - \frac{1}{2N}\right) - (L - 0.5)\,F - \frac{|D - 0.5|}{N}\,(1 + F) \right| \times 100\%$$

J = clock jitter
Eight times oversampling produces a receive margin = 39.25%
Four times oversampling produces a receive margin = 31.75%
Two times oversampling produces a receive margin = 16.75%

The fastest sample frequency is 48 kHz. This requires a clock speed of 128 × 48 kHz = 6.144 MHz. The worst case jitter in one cycle is specified at 40 ns = 24.5% of the period. This means that an oversampling rate of 4 or more will satisfy the inequality and therefore be sufficient for clock recovery.

Figure 26.8 illustrates the receiver data transfer using interrupts.

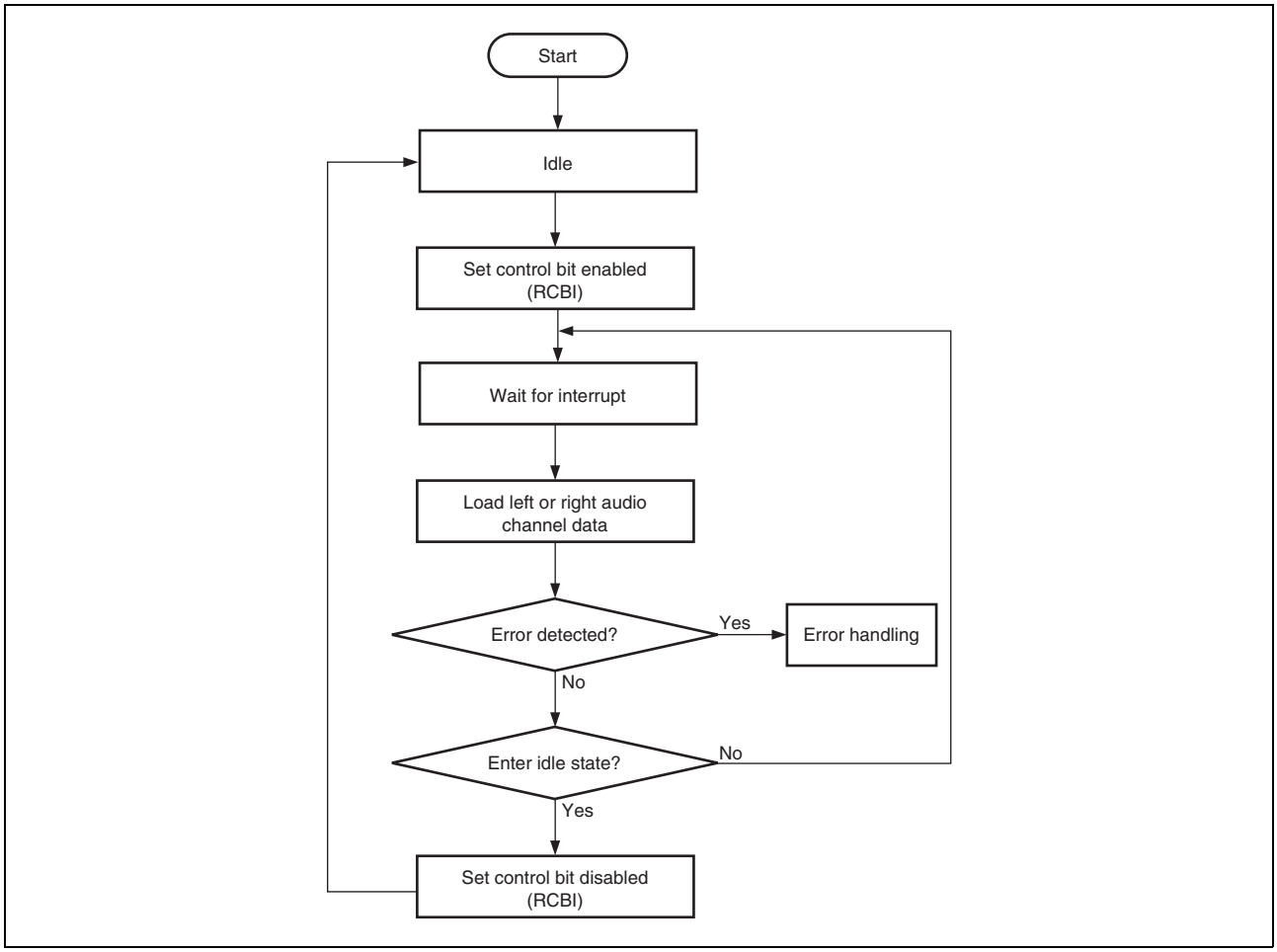


Figure 26.8 Receiver Data Transfer Flow Diagram - Interrupt Driven

Interrupts to indicate that the channel status information register is full occur after frame 30 has been received and only if the information has changed. When the first four bytes have been stored an interrupt occurs.

26.10 Disabling the Module

26.10.1 Transmitter and Receiver Idle

The transmitter or receiver modules can be disabled by writing 0 to the idle bit in the control register (TME for the transmitter and RME for the receiver). The idle state can be detected by polling the idle bit in the status register (TIS and RIS).

26.11 Compressed Mode Data

Compressed mode data is defined in the IEC 61937 specification. This module only detects compressed mode data. This is done by checking the parity flag (V flag) and bit 1 in the channel status data. If both are one then the data is in compressed mode. This is indicated by the setting of the CMD bit in the status register.

Note: Only the receiver detects compressed mode data since the information is not relevant to the transmitter.

26.12 References

IEC60958 Digital Audio Interface
IEC61937 Compressed Mode Digital Audio Interface

26.13 Usage Notes

26.13.1 Clearing TUIR

After TUI is written to, the TUIR bit is cleared only after transmission of a maximum of one frame is completed. When using a transmitter user information interrupt to write data to TUI, check that the TUIR bit is cleared before terminating the interrupt handling routine so that the interrupt is not unexpectedly accepted again.

26.13.2 Frequency of Clock Input for Audio

The frequency of the clock input to the AUDIO_X1 and AUDIO_X2 or AUDIO_CLK must be lower than the P1φ frequency.

27. Ethernet Controller (ETHERC)

27.1 Overview

This MCU has a two-channel Ethernet controller (ETHERC) compliant with the Ethernet or IEEE802.3 Media Access Control (MAC) layer protocol. Each ETHERC channel has one channel of the MAC layer interface, and connecting the MCU to the physical layer LSI (PHY-LSI) allows transmission and reception of frames compliant with the Ethernet/ IEEE802.3 standard. The ETHERC is connected to the Ethernet controller direct memory access controller (EDMAC), so data can be transferred without using the CPU.
Table 27.1 lists the ETHERC specifications. Figure 27.1 shows the ETHERC configuration. Table 27.2 lists the ETHERC I/O pins.
Figure 27.2 and Figure 27.3 show examples of connecting the MCU to the PHY-LSI.

Table 27.1 ETHERC Specifications

Item	Description
Number of channels	Two channels
Protocol	Flow control compliant with IEEE802.3x
Data transmission/reception	Frames compliant with the Ethernet/IEEE802.3 standard can be transmitted and received.
Bit rate	Supports 10 Mbps and 100 Mbps
Operation modes	Supports full-duplex and half-duplex modes
Interfaces	Media Independent Interface (MII), Reduced Media Independent Interface (RMII), compliant with the IEEE802.3u standard
Functions	Magic Packet™ *1 detection, Wake-On-LAN (WOL) signal output

Note 1. Magic Packet is a trademark of Advanced Micro Devices, Inc.

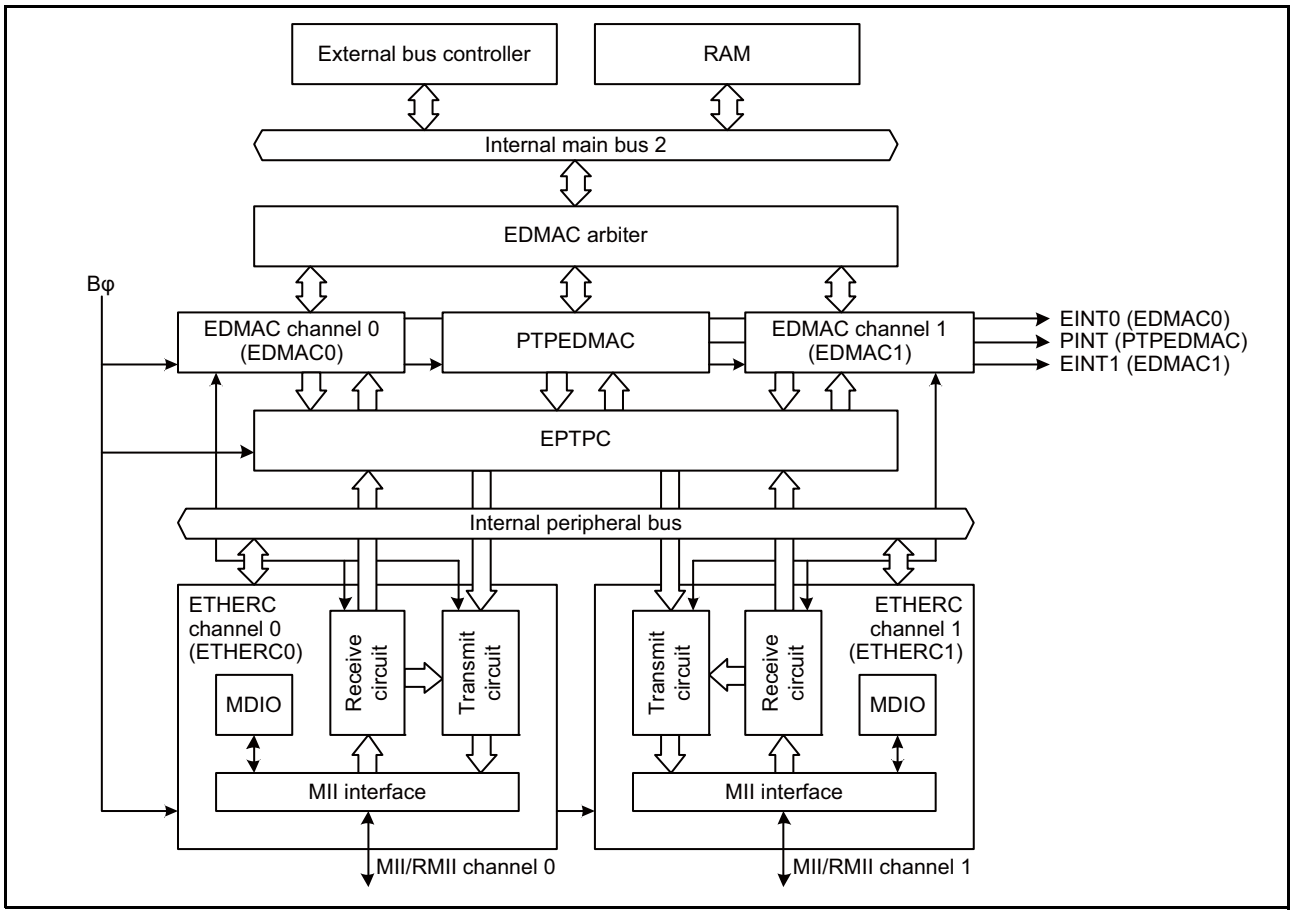


Figure 27.1 ETHERC Configuration

Table 27.2 ETHERC I/O Pins (n = 0, 1)

Operating Mode	Pin Name	I/O	Description
MII	ETn_TX_CLK *1	Input	Transmit clock Timing reference signal for outputting the ETn_TX_EN, ETn_ETXD3 to ETn_ETXD0, and ETn_TX_ER signals.
	ETn_RX_CLK *1	Input	Receive clock Timing reference signal for inputting the ETn_RX_DV, ETn_ERXD3 to ETn_ERXD0, and ETn_RX_ER signals
	ETn_TX_EN *1	Output	Transmit data valid This signal indicates that valid transmit data has been output on pins ETn_ETXD3 to ETn_ETXD0.
	ETn_ETXD3 to ETn_ETXD0 *1	Output	4-bit transmit data
	ETn_TX_ER *1	Output	This signal notifies the PHY-LSI that an error occurred during transmission.
	ETn_RX_DV *1	Input	Receive data valid This signal indicates that valid receive data is on pins ETn_ERXD3 to ETn_ERXD0.
	ETn_ERXD3 to ETn_ERXD0 *1	Input	4-bit receive data
	ETn_RX_ER *1	Input	Receive error This signal indicates that there is an error in a frame that is being transferred from the PHY-LSI to the ETHERC.
	ETn_CRS *1	Input	Carrier sense
	ETn_COL *1	Input	Collision detection signal
	ETn_MDC *1	Output	Management data clock Reference clock signal for transfer of information on the ETn_MDIO pin.
	ETn_MDIO *1	I/O	Management data I/O Bidirectional data signal for exchanging management data with the PHY-LSI.
	ETn_LINKSTA	Input	Link status input from the PHY-LSI
	ETn_EXOUT/ ETn_SCLKIN	Input/ Output	General output pin/STCA clock input
	ETn_WOL	Output	Wake-On-LAN. This signal indicates that a Magic Packet has been received.

Table 27.2 ETHERC I/O Pins (n = 0, 1)

Operating Mode	Pin Name	I/O	Description
RMII	REF50CKn*2	Input	Reference clock Timing reference signal for pins RMIIn_TXD_EN, RMIIn_TXD1 to RMIIn_TXD0, RMIIn_CRS_DV, RMIIn_RXD1 to RMIIn_RXD0, and RMIIn_RX_ER.
	RMIIn_TXD_EN *2	Output	Transmit data valid This signal indicates that valid transmit data has been output on pins RMIIn_TXD1 and RMIIn_TXD0.
	RMIIn_TXD1 to RMIIn_TXD0 *2	Output	2-bit transmit data
	RMIIn_CRS_DV *2	Input	Carrier sense/receive data valid This signal indicates that valid receive data is on pins RMIIn_RXD1 and RMIIn_RXD0.
	RMIIn_RXD1 to RMIIn_RXD0*2	Input	2-bit receive data
	RMIIn_RX_ER*2	Input	Receive error This signal indicates that there is an error in a frame that is being transferred from the PHY-LSI to the ETHERC.
	ETn_MDC *2	Output	Management data clock Reference clock signal for transfer of information on the ETn_MDIO pin
	ETn_MDIO*2	Input/ Output	Management data I/O Bidirectional data signal for exchanging management data with the PHYLSI.
	ETn_LINKSTA	Input	Link status input from the PHY-LSI.
	ETn_EXOUT/ ETn_SCLKIN	Input/ Output	General output pin/STCA clock input
	ETn_WOL	Output	Wake-On-LAN This signal indicates that a Magic Packet has been received.

Note 1. MII signal compliant with IEEE802.3u
Note 2. RMII signal compliant with IEEE802.3u

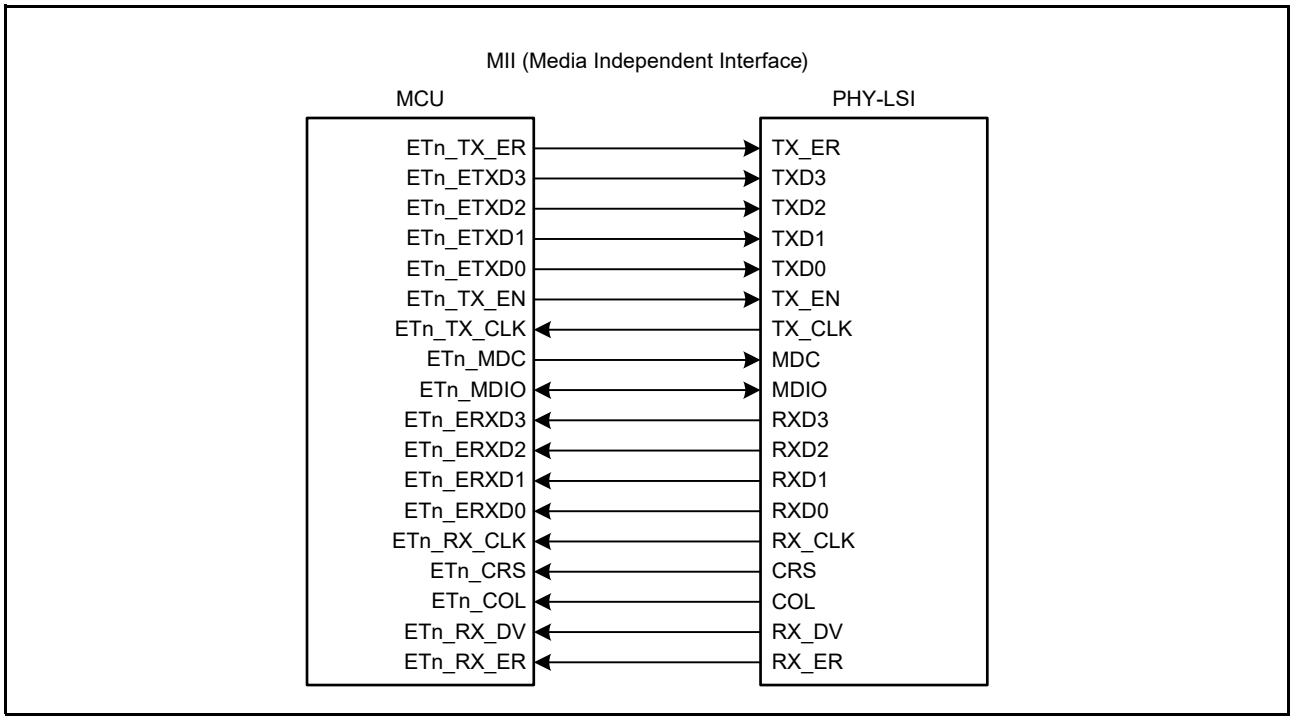


Figure 27.2 Example of Connection with the PHY-LSI for the MII

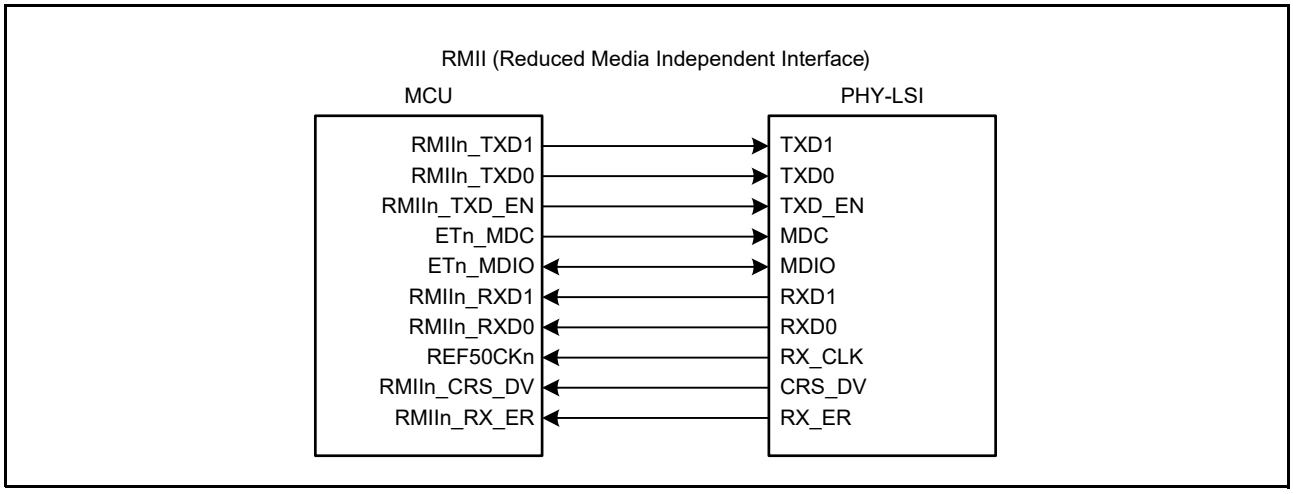


Figure 27.3 Example of Connection with the PHY-LSI for the RMII

27.2 Register Descriptions

Table 27.3 shows the register configuration.

Table 27.3 Register Configuration

Module	Register Name	Abbreviation	Address	Access size
ETHERC0	ETHERC mode register	ECMR	E820 4100h	32
	Receive frame maximum length register	RFLR	E820 4108h	32
	ETHERC status register	ECSR	E820 4110h	32
	ETHERC interrupt enable register	ECSIPR	E820 4118h	32
	PHY interface register	PIR	E820 4120h	32
	PHY status register	PSR	E820 4128h	32
	Random number generation counter limit setting register	RDMLR	E820 4140h	32
	Interpacket gap register	IPGR	E820 4150h	32
	Automatic PAUSE frame register	APR	E820 4154h	32
	Manual PAUSE frame register	MPR	E820 4158h	32
	Received PAUSE frame counter	RFCF	E820 4160h	32
	PAUSE frame retransmit count setting register	TPAUSER	E820 4164h	32
	PAUSE frame retransmit counter	TPAUSECR	E820 4168h	32
	Broadcast frame receive count setting register	BCFRR	E820 416Ch	32
	MAC address upper bit register	MAHR	E820 41C0h	32
	MAC address lower bit register	MALR	E820 41C8h	32
	Transmit retry over counter register	TROCR	E820 41D0h	32
	Late collision detect counter register	CDCR	E820 41D4h	32
	Lost carrier counter register	LCCR	E820 41D8h	32
	Carrier not detect counter register	CNDCR	E820 41DCh	32
	CRC error frame receive counter register	CEFCR	E820 41E4h	32
	Frame receive error counter register	FRECR	E820 41E8h	32
	Too-short frame receive counter register	TSFRCR	E820 41ECh	32
	Too-long frame receive counter register	TLFRCR	E820 41F0h	32
	Received alignment error frame counter register	RFCR	E820 41F4h	32
	Multicast address frame receive counter register	MAFCR	E820 41F8h	32
ETHERC1	ETHERC mode register	ECMR	E820 4300h	32
	Receive frame maximum length register	RFLR	E820 4308h	32
	ETHERC status register	ECSR	E820 4310h	32
	ETHERC interrupt enable register	ECSIPR	E820 4318h	32
	PHY interface register	PIR	E820 4320h	32
	PHY status register	PSR	E820 4328h	32
	Random number generation counter limit setting register	RDMLR	E820 4340h	32
	Interpacket gap register	IPGR	E820 4350h	32
	Automatic PAUSE frame register	APR	E820 4354h	32
	Manual PAUSE frame register	MPR	E820 4358h	32
	Received PAUSE frame counter	RFCF	E820 4360h	32
	PAUSE frame retransmit count setting register	TPAUSER	E820 4364h	32
	PAUSE frame retransmit counter	TPAUSECR	E820 4368h	32
	Broadcast frame receive count setting register	BCFRR	E820 436Ch	32
	MAC address upper bit register	MAHR	E820 43C0h	32
	MAC address lower bit register	MALR	E820 43C8h	32

Table 27.3 Register Configuration

Module	Register Name	Abbreviation	Address	Access size
ETHERC1	Transmit retry over counter register	TROCR	E820 43D0h	32
	Late collision detect counter register	CDCR	E820 43D4h	32
	Lost carrier counter register	LCCR	E820 43D8h	32
	Carrier not detect counter register	CNDCR	E820 43DCh	32
	CRC error frame receive counter register	CEFCR	E820 43E4h	32
	Frame receive error counter register	FRECR	E820 43E8h	32
	Too-short frame receive counter register	TSFRCR	E820 43ECh	32
	Too-long frame receive counter register	TLFRCR	E820 43F0h	32
	Received alignment error frame counter register	RFCR	E820 43F4h	32
	Multicast address frame receive counter register	MAFCR	E820 43F8h	32

27.2.1 ETHERC Mode Register (ECMR)

Address: ETHERC0.ECMR E820 4100h, ETHERC1.ECMR E820 4300h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	TPC	ZPF	PFR	RXF	TXF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PRCEF	—	—	MPDE	—	—	RE	TE	—	ILB	RTM	DM	PRM
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PRM	Promiscuous Mode	0: Promiscuous mode is disabled. 1: Promiscuous mode is enabled.	R/W
b1	DM	Duplex Mode	0: Half-duplex mode 1: Full-duplex mode	R/W
b2	RTM	Bit Rate	0: 10 Mbps 1: 100 Mbps	R/W
b3	ILB	Internal Loopback Mode	0: Normal data transmission or reception is performed. 1: Data is looped back in the ETHERC when full-duplex mode is selected.	R/W
b4	—	Reserved	The read value is 0. The write value should be 0.	R/W
b5	TE	Transmission Enable	0: Transmit function is disabled. 1: Transmit function is enabled	R/W
b6	RE	Reception Enable	0: Receive function is disabled. 1: Receive function is enabled.	R/W
b8, b7	—	Reserved	The read value is 0. The write value should be 0.	R/W
b9	MPDE	Magic Packet Detection Enable	0: Magic Packet detection is disabled. 1: Magic Packet detection is enabled.	R/W
b11, b10	—	Reserved	The read value is 0. The write value should be 0.	R/W
b12	PRCEF	CRC Error Frame Receive Mode	0: EDMAC is notified of a CRC error. 1: EDMAC is not notified of a CRC error.	R/W
b15 to b13	—	Reserved	The read value is 0. The write value should be 0.	R/W
b16	TXF	Transmit Flow Control Operating Mode	0: Automatic PAUSE frame transmission is disabled. (PAUSE frame is not automatically transmitted.) 1: Automatic PAUSE frame transmission is enabled. (PAUSE frame is automatically transmitted as required.)	R/W
b17	RXF	Receive Flow Control Operating Mode	0: PAUSE frame detection is disabled. 1: PAUSE frame detection is enabled.	R/W
b18	PFR	PAUSE Frame Receive Mode	0: PAUSE frame is not transferred to the EDMAC. 1: PAUSE frame is transferred to the EDMAC.	R/W
b19	ZPF	0 Time PAUSE Frame Enable	0: PAUSE frame that contains the pause_time parameter of 0 is not used. 1: PAUSE frame that contains the pause_time parameter of 0 is used.	R/W
b20	TPC	PAUSE Frame Transmit	0: PAUSE frame is transmitted even during a PAUSE period. 1: PAUSE frame is not transmitted during a PAUSE period.	R/W
b31 to b21	—	Reserved	The read value is 0. The write value should be 0.	R/W

The ECMR register controls the ETHERC operation.

Set bits in the ECMR register, excluding bits TE and RE, during initialization after a reset. When rewriting this register outside the initialization process, set the EDMACn.EDMR.SWR bit to 1 to reset the EDMAC and ETHERC, and then set this register again.

PRM Bit (Promiscuous Mode)

When the PRM bit is set to 1, the ETHERC operates in promiscuous mode where all Ethernet frames are received. In promiscuous mode, the ETHERC receives all valid frames regardless of whether or not the address matches the destination address or broadcast address and regardless of the multicast bit setting.

RTM Bit (Bit Rate)

The RTM bit sets the bit rate when the RMII is selected.

ILB Bit (Internal Loopback Mode)

When the ILB bit is set to 1, transmit frames can be looped back in the MCU. Set the DM bit to 1 (full-duplex mode) to perform a loopback test.

TE Bit (Transmission Enable)

When the TE bit is set to 1, the ETHERC transmit function is enabled.
When the TE bit is set to 0, the transmit function is disabled after the frame being processed is completely transmitted.

RE Bit (Reception Enable)

When the RE bit is set to 1, the ETHERC receive function is enabled.
When the RE bit is set to 0, the receive function is disabled after the frame being processed is completely received.

PRCEF Bit (CRC Error Frame Receive Mode)

When the PRCEF bit is set to 1, the EDMAC is not notified that a CRC error has occurred even when the error is detected in a receive frame. Accordingly, the EDMACn.EESR.CERF flag and RFS0 bit in receive descriptor 0 (RD0) do not become 1.

ZPF Bit (0 Time PAUSE Frame Enable)

When the ZPF bit is 1, a PAUSE frame that contains the pause_time parameter of 0 is transmitted when the PAUSE frame transmit request is canceled before the PAUSE time of the previously transmitted PAUSE frame has not elapsed. After the PAUSE frame that contains the pause_time parameter of 0 is received, the ETHERC is ready for transmission. When the ZPF bit is 0, even if the PAUSE frame transmit request from the receive FIFO is canceled, the next PAUSE frame is not transmitted until the PAUSE time of the previously transmitted PAUSE frame has elapsed. When a PAUSE frame that contains the pause_time parameter of 0 is received, the PAUSE frame is discarded.

27.2.2 Receive Frame Maximum Length Register (RFLR)

Address: ETHERC0.RFLR E820 4108h, ETHERC1.RFLR E820 4308h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	RFL[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b11 to b0	RFL[11:0]	Receive Frame Maximum Length	The set value becomes the maximum frame length. Values that are less than 1,518 bytes are regarded as 1,518 bytes, and values larger than 2,048 bytes are regarded as 2,048 bytes.	R/W
b31 to b12	—	Reserved	The read value is 0. The write value should be 0.	R/W

The RFLR register sets the maximum frame length that can be received by the MCU. Set the length in bytes.
Do not rewrite this register while the ECMR.RE bit is 1 (receive function is enabled).

RFL[11:0] Bits (Receive Frame Maximum Length)

The RFL[11:0] bits set a frame length to be checked. When the number of bytes for fields from the destination address to the frame check sequence (FCS) of the received frame exceeds the RFL[11:0] bit value, the EDMAC is notified of a frame-too-long error.
When the received frame length exceeds the RFL[11:0] bit value, the excess data is discarded.

27.2.3 ETHERC Status Register (ECSR)

Address: ETHERC0.ECSR E820 4110h, ETHERC1.ECSR E820 4310h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	BFR	PSRTO	—	LCHNG	MPD	ICD
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ICD	False Carrier Detect Flag	0: PHY-LSI has not detected a false carrier on the line. 1: PHY-LSI has detected a false carrier on the line.	R/W *1
b1	MPD	Magic Packet Detect Flag	0: Magic Packet has not been detected. 1: Magic Packet has been detected.	R/W *1
b2	LCHNG	Link Signal Change Flag	0: Change in the ETn_LINKSTA signal has not been detected. 1: Change in the ETn_LINKSTA signal has been detected (high to low, or low to high).	R/W *1
b3	—	Reserved	The read value is 0. The write value should be 0.	R/W
b4	PSRTO	PAUSE Frame Retransmit Over Flag	0: PAUSE frame retransmit count has not reached the upper limit. 1: PAUSE frame retransmit count has reached the upper limit.	R/W *1
b5	BFR	Continuous Broadcast Frame Reception Flag	0: Continuous reception of broadcast frames has not been detected. 1: Continuous reception of broadcast frames has been detected.	R/W *1
b31 to b6	—	Reserved	The read value is 0. The write value should be 0.	R/W

Note 1. Write 1 to clear the flag.

The ECSR register indicates the status of the ETHERC.
When any flag in the ECSR register becomes 1 while the corresponding bit in the ECSIPR register is 1 (interrupt is notified), the EDMACn.EESR.ECI flag becomes 1.

ICD Flag (False Carrier Detect Flag)

The ICD flag indicates that the PHY-LSI has detected a false carrier on the line.
The ICD flag becomes 1 when a receive error signal shown in Figure 1.11 is received from the PHY-LSI. Note that the information may not be correct when signals input from the PHY-LSI change faster than software recognizes the change. Check the timing of the PHY-LSI.

LCHNG Flag (Link Signal Change Flag)

The LCHNG flag indicates that ETn_LINKSTA signal input from the PHY-LSI has changed from high to low, or from low to high.
Refer to the PSR.LMON flag for the current link status.

PSRTO Flag (PAUSE Frame Retransmit Over Flag)

The PSRTO flag indicates that the number of retransmissions reaches the value set in the TPAUSER register when retransmitting a PAUSE frame while automatic PAUSE frame transmission is enabled.

27.2.4 ETHERC Interrupt Enable Register (ECSIPR)

Address: ETHERC0.ECSIPR E820 4118h, ETHERC1.ECSIPR E820 4318h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	BFSIPR	PSRTOIP	—	LCHNGIP	MPDIP	ICDIP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ICDIP	False Carrier Detect Interrupt Enable	0: Notification of the false carrier detect interrupt is disabled. 1: Notification of the false carrier detect interrupt is enabled.	R/W
b1	MPDIP	Magic Packet Detect Interrupt Enable	0: Notification of the Magic Packet detect interrupt is disabled. 1: Notification of the Magic Packet detect interrupt is enabled.	R/W
b2	LCHNGIP	LINK Signal Change Interrupt Enable	0: Notification of ETn_LINKSTA signal change interrupt is disabled. 1: Notification of ETn_LINKSTA signal change interrupt is enabled.	R/W
b3	—	Reserved	The read value is 0. The write value should be 0.	R/W
b4	PSRTOIP	PAUSE Frame Retransmit Over Interrupt Enable	0: Notification of PAUSE frame retransmit over interrupt is disabled. 1: Notification of PAUSE frame retransmit over interrupt is enabled.	
b5	BFSIPR	Continuous Broadcast Frame Reception Interrupt Enable	0: Notification of continuous broadcast frame reception interrupt is disabled. 1: Notification of continuous broadcast frame reception interrupt is enabled.	R/W
b31 to b6	—	Reserved	The read value is 0. The write value should be 0.	R/W

The ECSIPR register selects whether or not to notify the EDMAC of the status indicated by the ECSR register. Each bit corresponds to the flag in the ECSR register that has the same bit number.

27.2.5 PHY Interface Register (PIR)

Address: ETHERC0.PIR E820 4120h, ETHERC1.PIR E820 4320h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	MDI	MDO	MMD	MDC
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MDC	MII/RMII Management Data Clock	The MDC bit value is output from the ETn_MDC pin to supply the management data clock to the MII or RMII.	R/W
b1	MMD	MII/RMII Management Mode	0: Read 1: Write	R/W
b2	MDO	MII/RMII Management Data-Out	The MDO bit value is output from the ETn_MDIO pin when the MMD bit is 1 (write). The value is not output when the MMD bit is 0 (read).	R/W
b3	MDI	MII/RMII Management Data-In	This bit indicates the level of the ETn_MDIO pin. The write value should be 0.	R/W
b31 to b4	—	Reserved	The read value is 0. The write value should be 0.	R/W

The PIR register is used to access registers in the PHY-LSI via the MII or RMII. The management clock and management data are controlled by software.
Refer to section 27.3.4, Accessing MII/RMII Registers for details on accessing MII and RMII registers.

27.2.6 PHY Status Register (PSR)

Address: ETHERC0.PSR E820 4128h, ETHERC1.PSR E820 4328h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LMON
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	LMON	ETn_LINKSTA Pin Status Flag	The link status can be read by connecting the link signal output from the PHY-LSI to the ETn_LINKSTA pin. For details on the polarity, refer to the specifications of the connected PHY-LSI.	R
b31 to b1	—	Reserved	The read value is 0. The write value should be 0.	R

The PSR register is used to monitor interface signals from the PHY-LSI.

27.2.7 Random Number Generation Counter Limit Setting Register (RDMLR)

Address: ETHERC0.RDMLR E820 4140h, ETHERC1.RDMLR E820 4340h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	RMD[19:16]			
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RMD[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b19 to b0	RMD[19:0]	Random Number Generation Counter	00000h: Normal operation 00001h to FFFFFh: Setting prohibited	R/W
b31 to b20	—	Reserved	The read value is 0. The write value should be 0.	R/W

The RDMLR register sets the maximum value for the counter used in the random number generator.
Do not rewrite this register while the ECMR.TE bit is 1 (transmit function is enabled) or while the ECMR.RE bit is 1 (receive function is enabled).

27.2.8 Interpacket Gap Register (IPGR)

Address: ETHERC0.IPGR E820 4150h, ETHERC1.IPGR E820 4350h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	IPG[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	IPG[4:0]	Interpacket Gap	00h: 16 bit time 01h: 20 bit time : : 14h: 96 bit time (initial value) : : 1Fh: 140 bit time	R/W
b31 to b5	—	Reserved	The read value is 0. The write value should be 0.	R/W

The IPGR register sets the interpacket gap (IPG) value.
Do not rewrite this register while the ECMR.TE bit is 1 (transmit function is enabled) or while the ECMR.RE bit is 1 (receive function is enabled).
Refer to section 27.3.6, Adjusting Transmission Efficiency by Changing the IPG.

27.2.9 Automatic PAUSE Frame Register (APR)

Address: ETHERC0.APR E820 4154h, ETHERC1.APR E820 4354h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AP[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	AP[15:0]	Automatic PAUSE Time Setting	These bits set the value of the pause_time parameter for a PAUSE frame that is automatically transmitted. Transmission is not performed until the set value multiplied by 512 bit time has elapsed.	R/W
b31 to b16	—	Reserved	The read value is 0. The write value should be 0.	R/W

The APR register sets the PAUSE time of the PAUSE frame that is automatically transmitted. The value set in the APR register is used for the pause_time parameter of the PAUSE frame.
Do not rewrite this register while the ECMR.TE bit is 1 (transmit function is enabled) or while the ECMR.RE bit is 1 (receive function is enabled).

27.2.10 Manual PAUSE Frame Register (MPR)

Address: ETHERC0.MPR E820 4158h, ETHERC1.MPR E820 4358h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	MP[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	MP[15:0]	Manual PAUSE Time Setting	These bits set the value of the pause_time parameter for a PAUSE frame that is manually transmitted. Transmission is not performed until the set value multiplied by 512 bit time has elapsed. The read value is undefined.	R/W
b31 to b16	—	Reserved	The read value is 0. The write value should be 0.	R/W

The MPR register sets the PAUSE time of the PAUSE frame that is manually transmitted. The value set in the MPR register is used for the pause_time parameter of the PAUSE frame.
When a value is set to this register, a PAUSE frame is transmitted. Rewrite this register while the ECMR.TE bit is 1 (transmit function is enabled).

27.2.11 Received PAUSE Frame Counter (RFCF)

Address: ETHERC0.RFCF E820 4160h, ETHERC1.RFCF E820 4360h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	RPAUSE[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	RPAUSE [7:0]	Manual PAUSE Time Setting	Received PAUSE Frame CountNumber of received PAUSE frames	R
b31 to b8	—	Reserved	The read value is 0.	R

The RFCF register is a counter indicating the number of received PAUSE frames. The counter is reset after this register is read.

27.2.12 PAUSE Frame Retransmit Count Setting Register (TPAUSER)

Address: ETHERC0.TPAUSER E820 4164h, ETHERC1.TPAUSER E820 4364h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TPAUSE[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	TPAUSE [15:0]	Automatic PAUSE Frame Retransmit Setting	0000h: Number of retransmissions is unlimited 0001h: Maximum number of retransmissions is 1 : : FFFFh: Maximum number of retransmissions is 65,535	R/W
b31 to b16	—	Reserved	The read value is 0. The write value should be 0.	R/W

The TPAUSER register selects the maximum number of times a PAUSE frame is automatically transmitted.
Do not rewrite this register while the ECMR.TE bit is 1 (transmit function is enabled).

27.2.13 PAUSE Frame Retransmit Counter (TPAUSECR)

Address: ETHERC0.TPAUSECR E820 4168h, ETHERC1.TPAUSECR E820 4368h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	TXP[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	TXP[7:0]	PAUSE Frame Retransmit Count	Number of times a PAUSE frame was retransmitted	R
b31 to b8	—	Reserved	The read value is 0.	R

The TPAUSECR register is a counter indicating the number of times a PAUSE frame was automatically retransmitted. The counter is reset after this register is read.

27.2.14 Broadcast Frame Receive Count Setting Register (BCFRR)

Address: ETHERC0.BCFRR E820 416Ch, ETHERC1.BCFRR E820 436Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BCF[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	BCF[15:0]	Broadcast Frame Continuous Receive Count Setting	0000h: Number of receptions is unlimited. 0001h: Receive 1 frame. : FFFFh: Receive 65,535 frames.	R/W
b31 to b16	—	Reserved	The read value is 0. The write value should be 0.	R/W

The BCFRR register sets the number of times broadcast frames can be received continuously. When the number of received frames exceeds the BCF[15:0] bit value, the ECSR.BFR flag is set to 1 and the subsequent broadcast frames are discarded. Note that the internal counter that counts the continuous reception of broadcast frames is reset if it receives the frames other than broadcast frames.
Do not rewrite this register while the ECMR.RE bit is 1 (receive function is enabled).

27.2.15 MAC Address Upper Bit Register (MAHR)

Address: ETHERC0.MAHR E820 41C0h, ETHERC1.MAHR E820 43C0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The MAHR register sets the upper 32 bits (b47 to b16) of the 48-bit MAC address. For example, if the MAC address is 01-23-45-67-89-AB, set the register to 0123 4567h.
Set the MAHR register during initialization after a reset. Do not rewrite this register while the ECMR.TE bit is 1 (transmit function is enabled) or while the ECMR.RE bit is 1 (receive function is enabled). When rewriting this register, set the EDMACn.EDMR.SWR bit to 1 to reset the EDMAC and ETHERC and then set this register again.

27.2.16 MAC Address Lower Bit Register (MALR)

Address: ETHERC0.MALR E820 41C8h, ETHERC1.MALR E820 43C8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	—	These bits set the lower 16 bits of the MAC address.	R/W
b31 to b16	—	Reserved	The read value is 0. The write value should be 0.	R/W

The MALR register sets the lower 16 bits of the 48-bit MAC address. For example, if the MAC address is 01-23-45-67-89-AB, set the register to 0000 89ABh.
Set the MALR register during initialization after a reset. Do not rewrite this register while the ECMR.TE bit is 1 (transmit function is enabled) or while the ECMR.RE bit is 1 (receive function is enabled). When rewriting this register, set the EDMACn.EDMR.SWR bit to 1 to reset the EDMAC and ETHERC and then set this register again.

27.2.17 Transmit Retry Over Counter Register (TROCR)

Address: ETHERC0.TROCR E820 41D0h, ETHERC1.TROCR E820 43D0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The TROCR register is a counter indicating the number of frames that fail to be retransmitted.
The TROCR register is incremented by 1 when a frame fails to be retransmitted 15 times. The counter stops when the TROCR register value becomes FFFF FFFFh. The counter value becomes 0 by writing any value to the TROCR register.

27.2.18 Late Collision Detect Counter Register (CDCR)

Address: ETHERC0.CDCR E820 41D4h, ETHERC1.CDCR E820 43D4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The CDCR register is a counter indicating the number of late collisions that have been detected after transmission starts. When the CDCR register value becomes FFFF FFFFh, the counter stops. The counter value becomes 0 by writing any value to the CDCR register.

27.2.19 Lost Carrier Counter Register (LCCR)

Address: ETHERC0.LCCR E820 41D8h, ETHERC1.LCCR E820 43D8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The LCCR register is a counter indicating the number of times a loss of carrier is detected during frame transmission. When the LCCR register value becomes FFFF FFFFh, the counter stops. The counter value becomes 0 by writing any value to the LCCR register.

27.2.20 Carrier Not Detect Counter Register (CNDCR)

Address: ETHERC0.CNDCR E820 41DCh, ETHERC1.CNDCR E820 43DCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The CNDCR register is a counter indicating the number of times a carrier is not detected during preamble transmission. When the CNDCR register value becomes FFFF FFFFh, the counter stops. The counter value becomes 0 by writing any value to the CNDCR register.

27.2.21 CRC Error Frame Receive Counter Register (CEFCR)

Address: ETHERC0.CEFCR E820 41E4h, ETHERC1.CEFCR E820 43E4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The CEFCR register is a counter indicating the number of received frames where a CRC error has been detected. When the CEFCR register value becomes FFFF FFFFh, the counter stops. The counter value becomes 0 by writing any value to the CEFCR register.

27.2.22 Frame Receive Error Counter Register (FRECR)

Address: ETHERC0.FRECR E820 41E8h, ETHERC1.FRECR E820 43E8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The FRECR register is a counter indicating the number of times a frame receive error has occurred. The PHY-LSI notifies the ETHERC of the frame receive error using the ETn_RX_ER pin.
The FRECR register is incremented each time the ETn_RX_ER pin becomes high. When the FRECR register value becomes FFFF FFFFh, the counter stops. The counter value becomes 0 by writing any value to the FRECR register.
The SYSR.INFABT flag of the EPTPC may be set to 1 when a frame receive error occurs, regardless of whether or not the EPTPC is in use. In such cases, reset the EPTPC, PTPEDMAC, and respective channel of the ETHERC and EDMAC. For the procedure of resetting, see section 27, Handling when Control Information Included a Mismatch.

27.2.23 Too-Short Frame Receive Counter Register (TSFRCR)

Address: ETHERC0.TSFRCR E820 41ECh, ETHERC1.TSFRCR E820 43ECh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The TSFRCR register is a counter indicating the number of times a short frame that is shorter than 64 bytes has been received. When the TSFRCR register value becomes FFFF FFFFh, the counter stops. The counter value becomes 0 by writing any value to the TSFRCR register.

27.2.24 Too-Long Frame Receive Counter Register (TLFRCR)

Address: ETHERC0.TLFRCR E820 41F0h, ETHERC1.TLFRCR E820 43F0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The TLFRCR register is a counter indicating the number of times a long frame that is longer than the RFLR register value has been received.
When the TLFRCR register value becomes FFFF FFFFh, the counter stops. The counter value becomes 0 by writing any value to the TLFRCR register.
Note that the TLFRCR register is not incremented when a frame is received with the alignment error. In this case, the RFCR register is incremented.

27.2.25 Received Alignment Error Frame Counter Register (RFCR)

Address: ETHERC0.RFCR E820 41F4h, ETHERC1.RFCR E820 43F4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The RFCR register is a counter indicating the number of times a frame has been received with the alignment error (frame is not an integral number of octets).
When the RFCR register value becomes FFFF FFFFh, the counter stops. The counter value becomes 0 by writing any value to the RFCR register.
The SYSR.INFABT flag of the EPTPC may be set to 1 when an alignment error is detected, regardless of whether or not the EPTPC is in use. In such cases, reset the EPTPC, PTPEDMAC, and respective channel of the ETHERC and EDMAC. For the procedure of resetting, see section 27, Handling when Control Information Included a Mismatch.

27.2.26 Multicast Address Frame Receive Counter Register (MAFCR)

Address: ETHERC0.MAFCR E820 41F8h, ETHERC1.MAFCR E820 43F8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

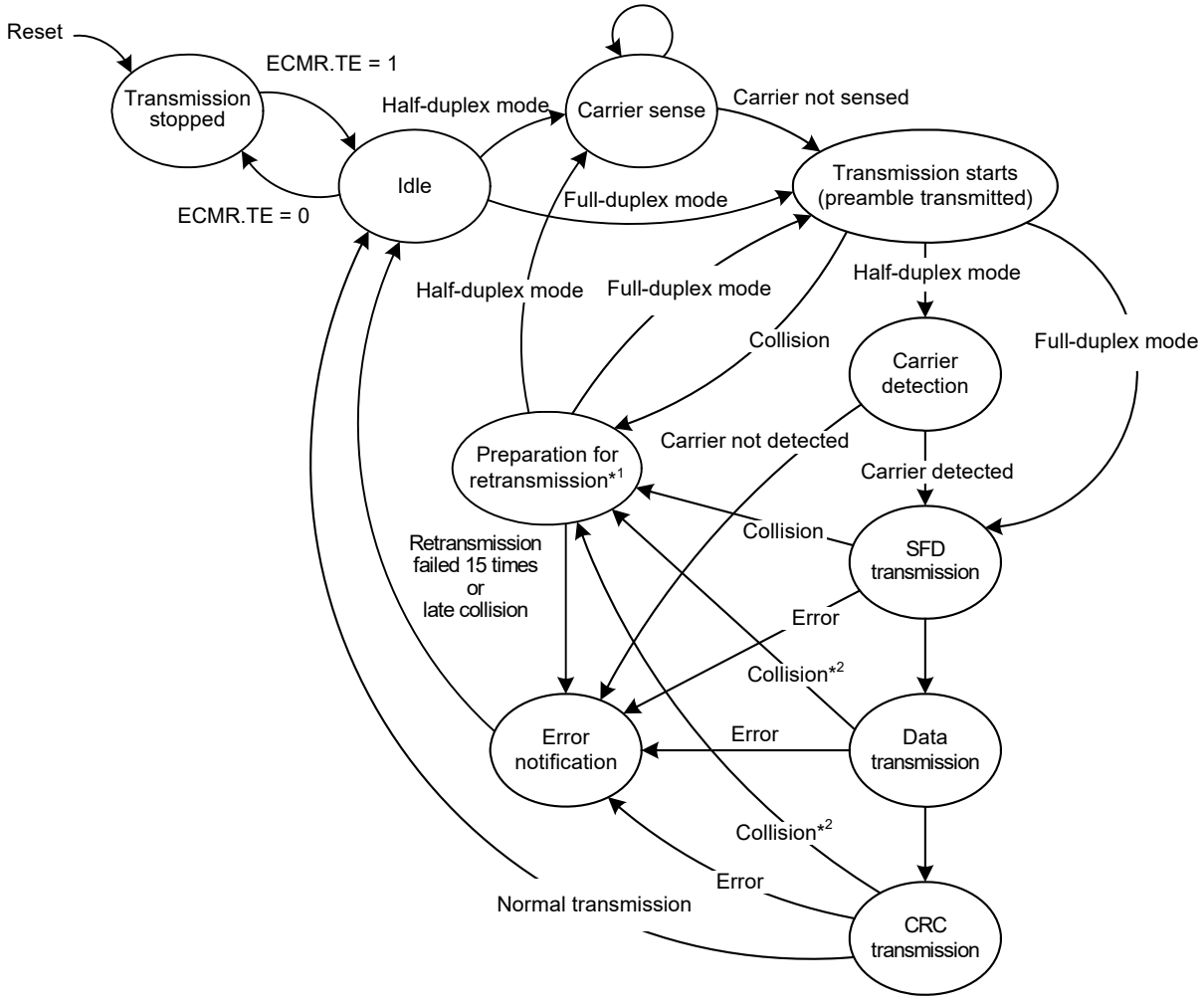
The MAFCR register is a counter indicating the number of times a frame where the multicast address is set has been received.
When the RFCR register value becomes FFFF FFFFh, the counter stops. The counter value becomes 0 by writing any value to the RFCR register.

27.3 Operation

This section is an overview of the ETHERC operations. The ETHERC supports flow control compliant with IEEE802.3x, and can transmit and receive PAUSE frames.

27.3.1 Transmission

The ETHERC transmitter assembles transmit data into a frame and outputs it to the MII/RMII when a transmit request is received from the EDMAC. The frame transmitted via the MII/RMII is transmitted on the line by the PHY-LSI. Figure 27.4 shows the state transitions of the ETHERC transmit-ter.



Note 1. Preparation for retransmission includes transmitting a jam signal and adjusting transmission time intervals using the backoff algorithm after a collision is detected.
Note 2. Preparation for retransmission is performed only when a collision is detected within 512 bit time after preamble transmission starts. When a collision is detected after 512 bit time, only a jam signal is transmitted and processing using the backoff algorithm is not performed.

Figure 27.4 ETHERC Transmitter State Transitions

1. When setting the ECMR.TE bit to 1, the ETHERC enters the transmit idle state.
2. When a transmit request is received from the EDMAC, the ETHERC enters the carrier sense state. The ETHERC waits for the interpacket gap and then transmits a preamble to the MII/RMII. When full-duplex mode is selected, it is not required to sense a carrier, so the ETHERC transmits a preamble immediately after receiving a transmit request from the EDMAC.
3. The ETHERC transmits the Start Frame Delimiter (SFD), transmit data, and CRC sequentially. When the transmission is completed successfully, the ETHERC notifies the EDMAC of successful completion, and the EDMAC sets the EDMACn.EESR.TC flag to 1. When a late collision or loss of carrier is detected during data transmission, the ETHERC stops the transmission and notifies the EDMAC of the error.
4. After the time for the interpacket gap has elapsed, the ETHERC enters the idle state and continues transmission when transmit data remains.

27.3.2 Reception

The ETHERC receiver separates the frame input from the MII/RMII into the preamble, SFD, receive data, and CRC, and transmits only receive data (destination address, source address, type/length, data/LLC). Figure 27.5 shows the state transitions of the ETHERC receiver.

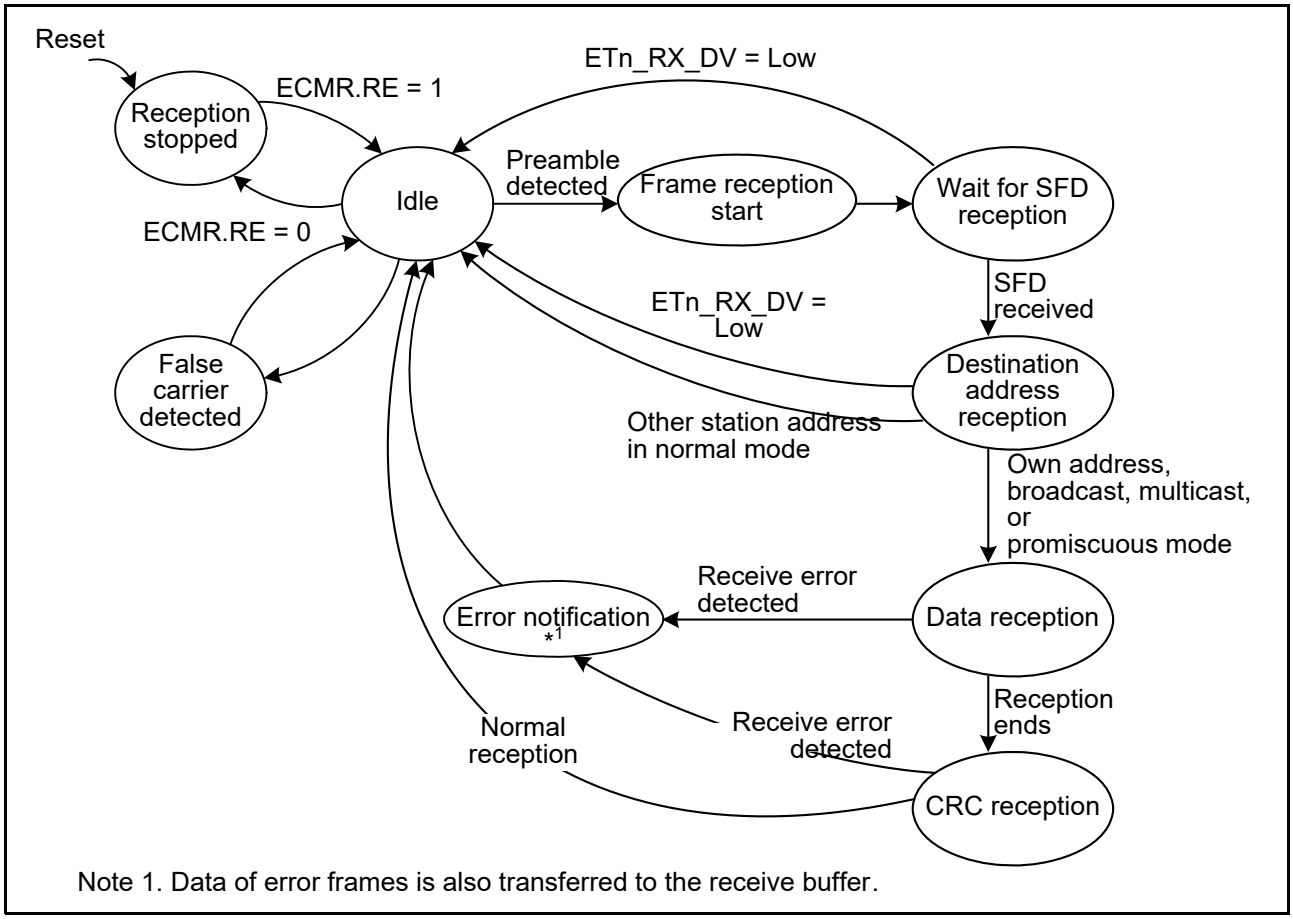


Figure 27.5 ETHERC Receiver State Transitions

1. When setting the ECMR.RE bit to 1, the ETHERC enters the receive idle state.
2. When the SFD following the preamble of the receive packet is detected, the ETHERC starts reception. If the received SFD is invalid, the ETHERC discards the frame.
3. In normal mode, the ETHERC starts data reception when the destination address of the receive frame is the address of the MCU or the receive frame is broadcast frame or multicast frame. In promiscuous mode, the ETHERC starts data reception regardless of the receive frame type.
4. After receiving data from the MII/RMII, the ETHERC performs the CRC check. The ETHERC notifies the EDMAC of the CRC check result. After the received data is transferred to the receive buffer, the CRC check result is written back to the receive descriptor as a status. The result is also reflected in the EDMACn.EESR.CERF flag.
5. When the ECMR.RE bit is 1 after one frame has been received, the ETHERC prepares to receive the next frame.

27.3.3 Frame Timing

27.3.3.1 MII Frame Timing

Figure 27.6 to Figure 27.11 show the MII frame timing.

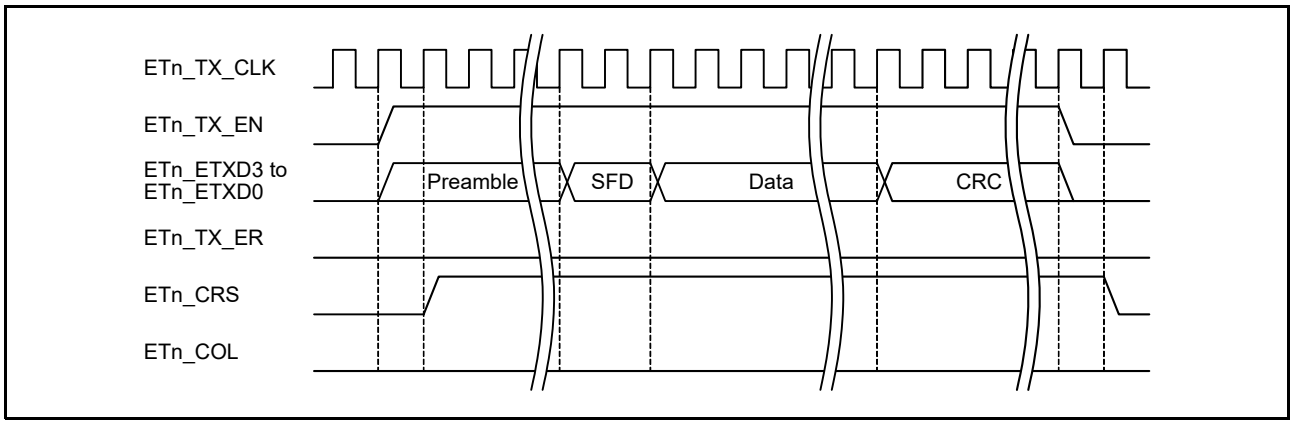


Figure 27.6 MII Frame Transmit Timing During Normal Transmission

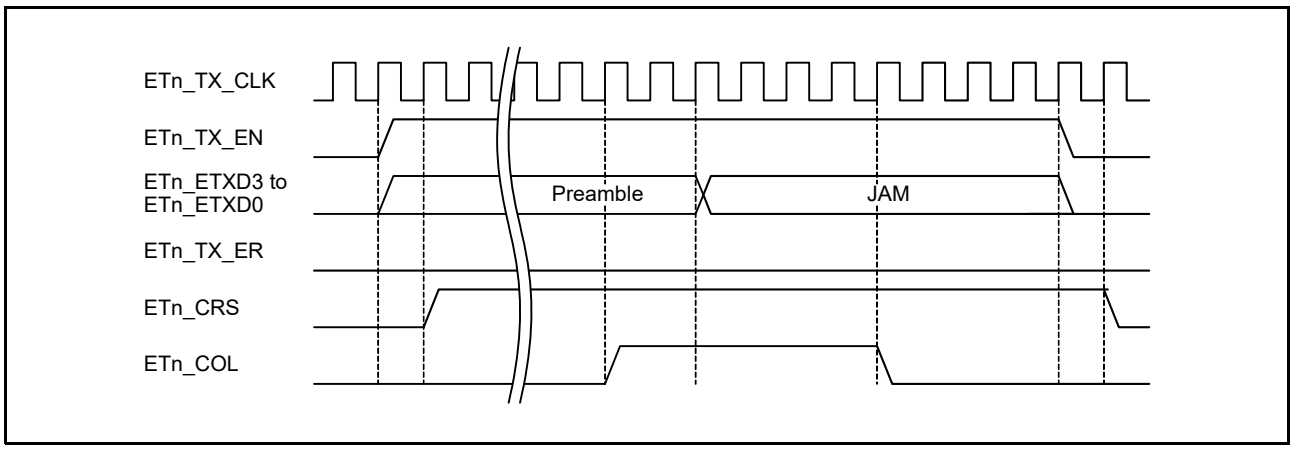


Figure 27.7 MII Frame Transmit Timing When Collision Occurs

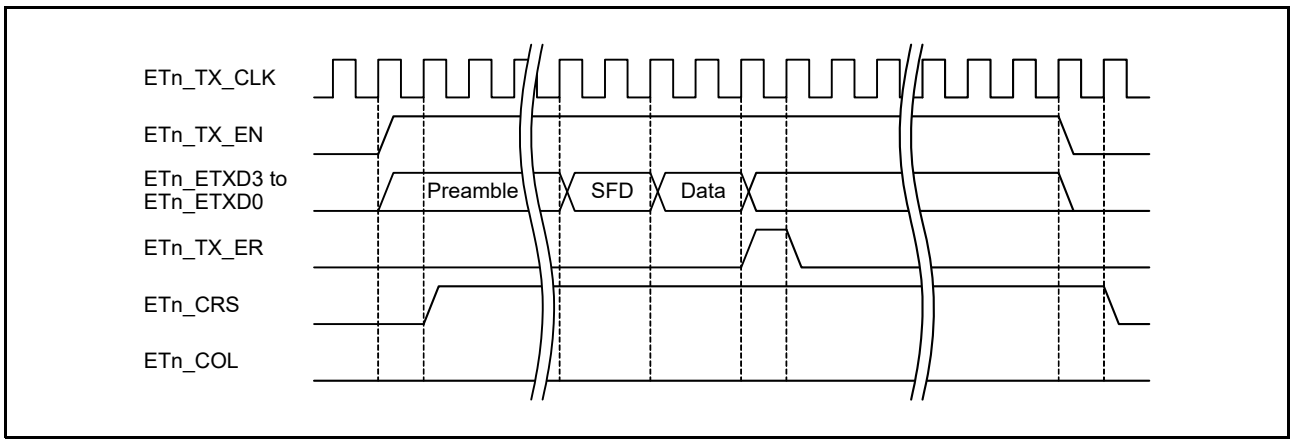


Figure 27.8 MII Frame Transmit Timing When Transmit Error Occurs

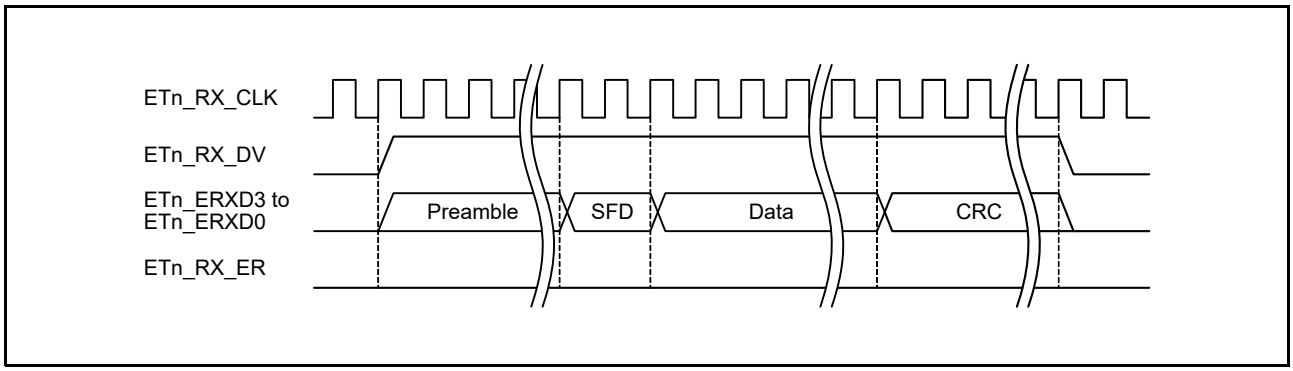


Figure 27.9 MII Frame Receive Timing During Normal Reception

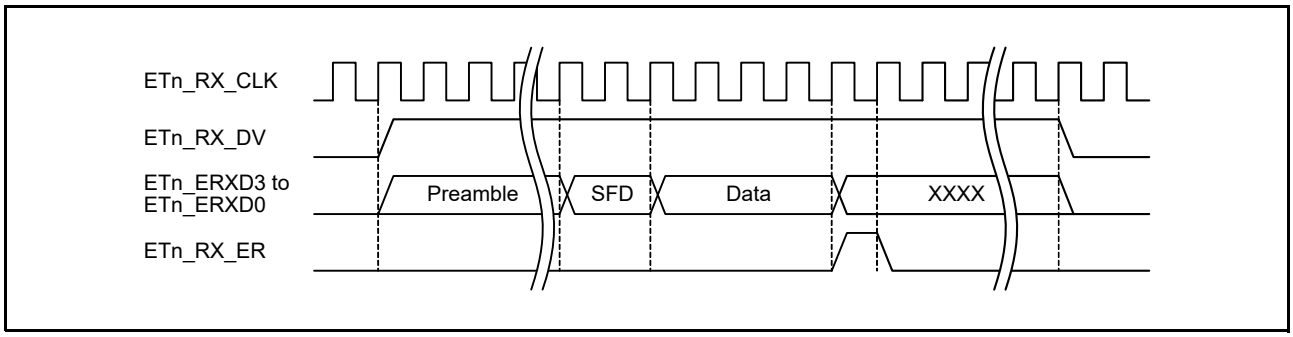


Figure 27.10 MII Frame Receive Timing for Receive Error Notification

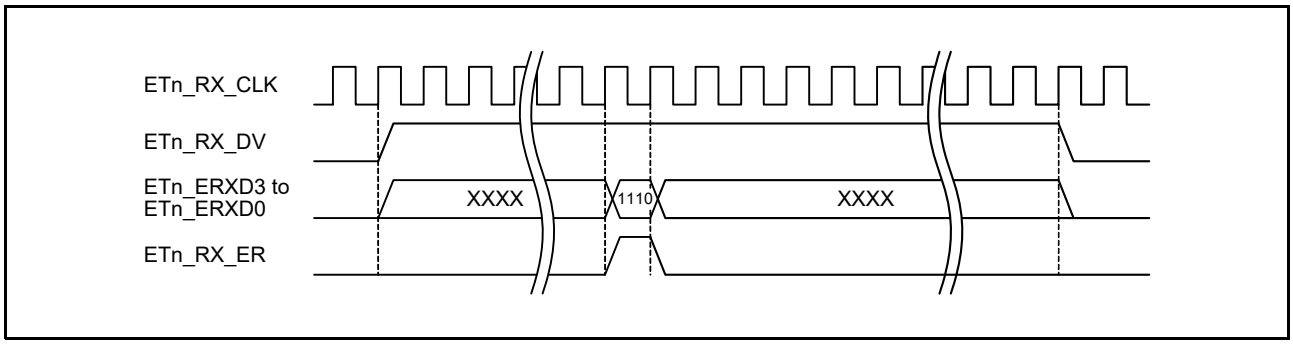


Figure 27.11 MII Fame Receive Timing for False Carrier Notification

27.3.3.2 RMII Frame Timing

The RMII frame timing is shown in Figure 27.12 to Figure 27.14.

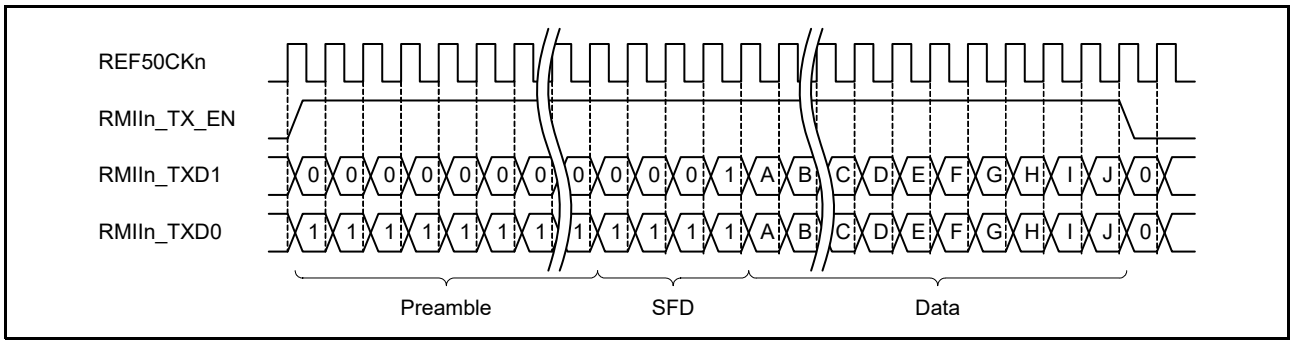


Figure 27.12 RMII Frame Transmit Timing During Normal Transmission

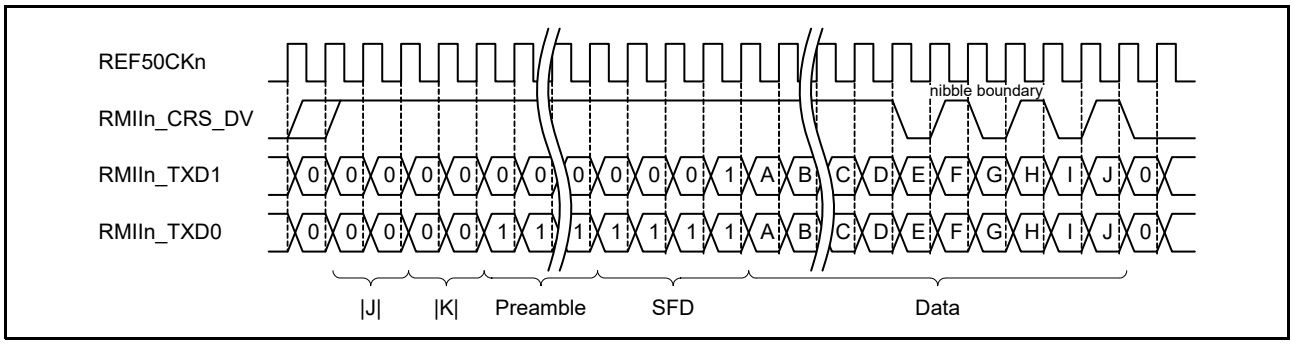


Figure 27.13 RMII Frame Receive Timing During Normal Reception

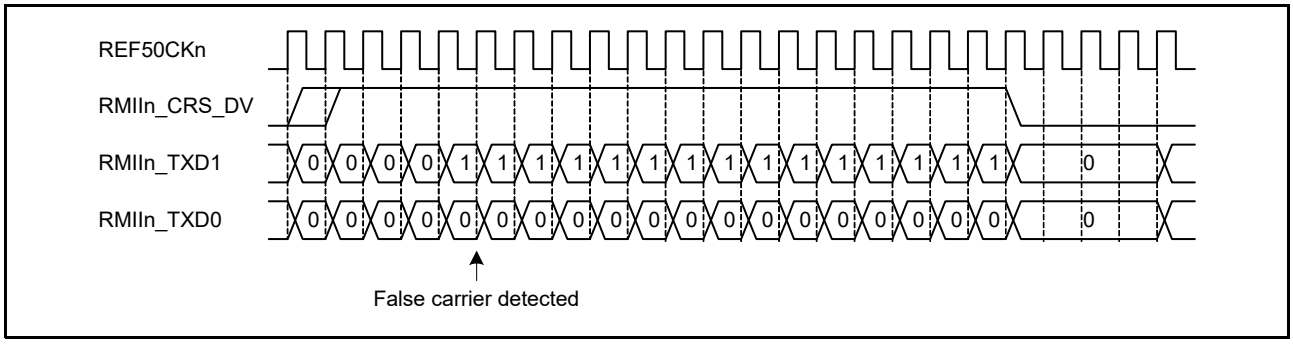


Figure 27.14 RMII Frame Receive Timing When False Carrier Is Detected

27.3.4 Accessing MII/RMII Registers

Use the PIR register to access the MII/RMII registers in the PHY-LSI. Serial data in the MII/RMII management frame format is transmitted and received via the ETn_MDC and ETn_MDIO pins controlled by software.

27.3.4.1 MII/RMII Management Frame Format

Table 27.4 lists the MII/RMII management frame format.

Table 27.4 MII/RMII Management Frame Format

Access Type	MII/RMII Management Frame							
Item	PRE	ST	OP	PHYAD	REGAD	TA	DATA	IDLE
Number of bits	32	2	2	5	5	2	16	1
Read	1...1	01	10	00001	RRRRR	Z0	DDDDDDDDDDDDDDDD	Z
Write	1...1	01	01	00001	RRRRR	10	DDDDDDDDDDDDDDDD	Z

PRE (preamble): Send 32 consecutive 1's.

ST (start of frame): Send 01b.

OP (operation code): Send 10b for read or 01b for write.

PHYAD (PHY address): Up to 32 PHY-LSIs can be connected to one MAC. PHY-LSIs are selected with these 5 bits.
When the PHY-LSI address is 1, send 00001b.

REGAD (register address): One register is selected from up to 32 registers in the PHY-LSI.
When the register address is 1, send 00001b.

TA (turnaround): 2-bit turnaround time to avoid contention between the register address and data during a read operation
(a) Send 10b during a write operation.
(b) Release the bus for 1 bit during a read operation (Z is output)
(indicated as Z0 because 0 is output from the PHY-LSI on the next clock cycle).

DATA (data): 16-bit data. Sequentially send or receive starting from the MSB.

IDLE (IDLE condition): Wait time to input the next MII/RMII management format
(a) Release the bus during a write operation (Z is output).
(b) No control is required since a bus has already been released during a read operation.

27.3.4.2 MII/RMII Register Access Procedure

Access to the MII/RMII registers includes writing data in 1-bit units, reading data in 1-bit units, and releasing the bus. Figure 27.15 to Figure 27.18 show examples of the MII/RMII register access timing. The access timing differ with the PHY-LSI type.

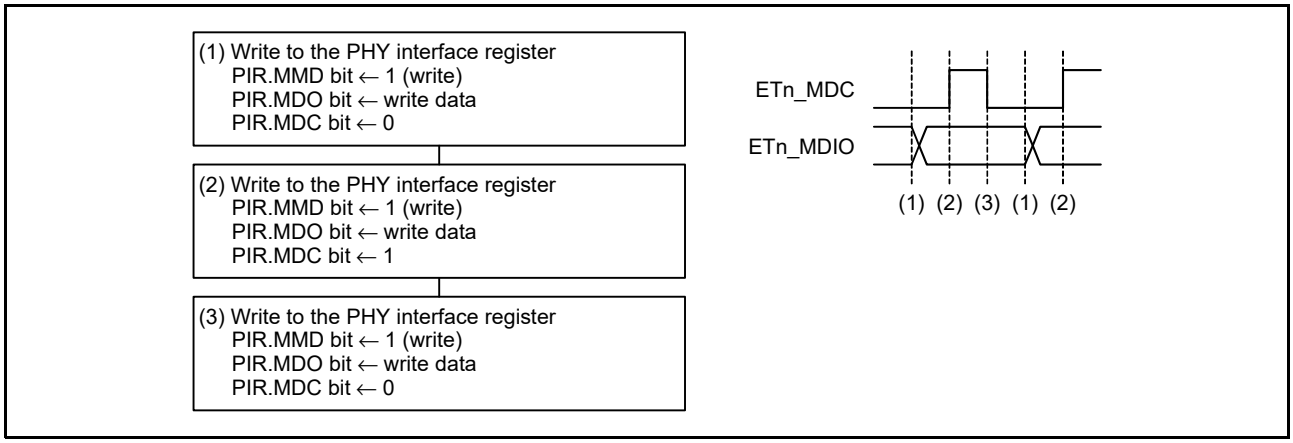


Figure 27.15 1-Bit Data Write Flow

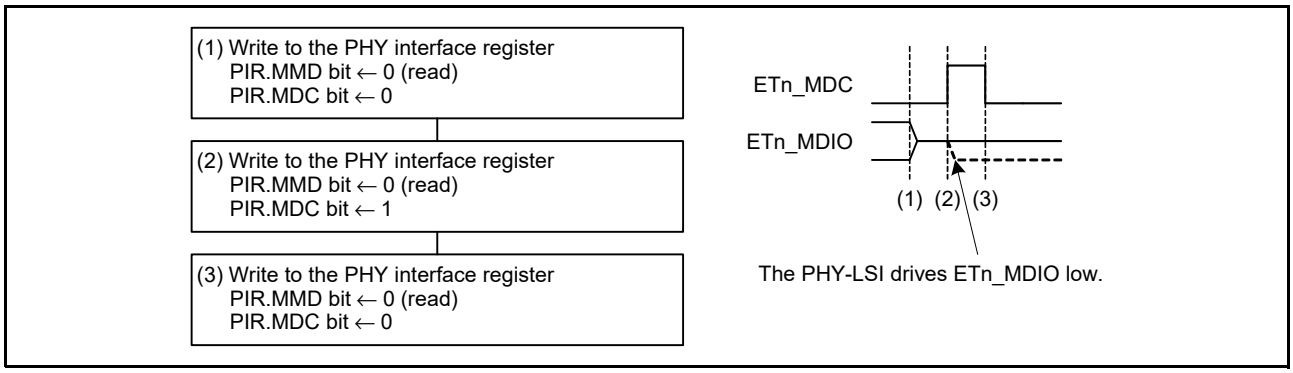


Figure 27.16 Bus Release Flow (TA in Read Operation in Table 1.3)

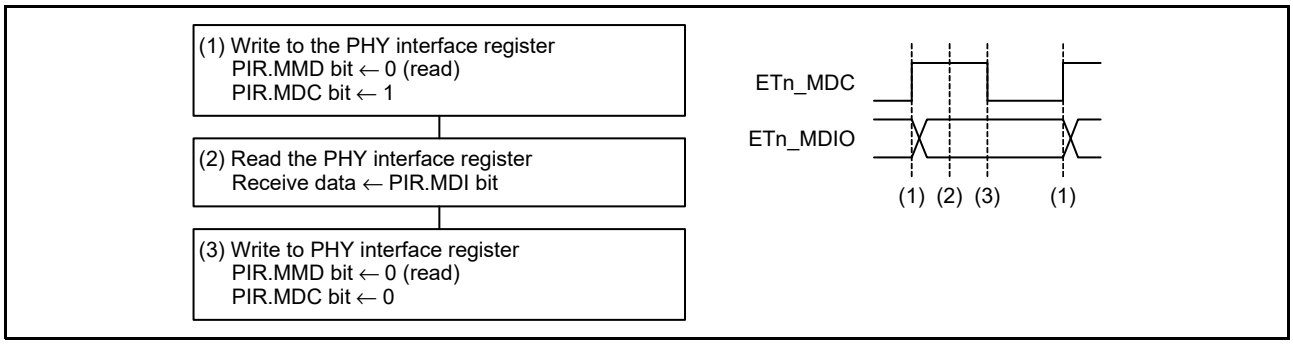


Figure 27.17 1-Bit Data Read Flow

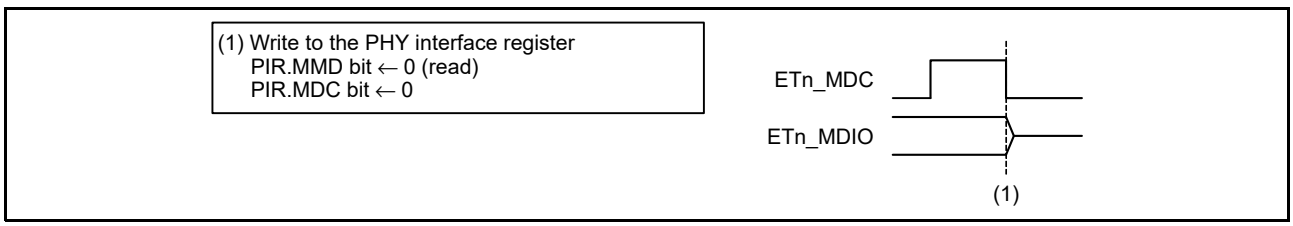


Figure 27.18 Bus Release Flow (IDLE in Write Operation in Table 1.3)

27.3.5 Magic Packet Detection

The ETHERC supports Wake-On-LAN (WOL). WOL is a function to detect a Magic Packet transmitted from a host device or other device and exit a low power consumption state such as sleep mode. When the ETHERC detects a Magic Packet, high is output from the ETn_WOL pin. Write 1 to the EDMACn.EDMR.SWR bit to set the ETn_WOL pin to low.

Since a Magic Packet is transmitted in broadcast mode, the Magic Packet is received regardless of the destination MAC address selected in the format. The ETHERC outputs high from the ETn_WOL pin only when the destination MAC address matches its own MAC address. Refer to the technical documentation provided by Advanced Micro Devices, Inc. for details on the Magic Packet. The following describes an example of the procedure to use WOL in the MCU.

1. Set the interrupt controller to disable the EINTn interrupt request.
2. Set the ECMR.MPDE bit to 1 to enable Magic Packet detection. Set the ECMR.RE bit to 1 to enable reception.
3. Set the ECSIPR.MPDIP bit to 1 to enable notification of the Magic Packet detect interrupt.
4. Set the EDMACn.EESIPR.ECIIP bit to 1 to enable the ETHERC status register source interrupt.
5. Set the interrupt controller to enable the EINTn interrupt request.
6. Change the CPU operating mode to sleep mode or place unused peripherals in the module stop state as needed.
7. When a Magic Packet is detected, an interrupt request is sent to the CPU. High is output from the ETn_WOL pin to notify peripheral devices that the Magic Packet has been detected.

27.3.5.1 Notes on Magic Packet Detection

The ETHERC receives packets, including broadcast packets, even when waiting to receive a Magic Packet. Therefore, receive data may have been stored in the receive FIFO of the EDMAC when a Magic Packet is detected. Also, flags in registers ECSR and EDMACn.EESR may have been changed. When returning to normal operation by detecting a Magic Packet, set the EDMACn.EDMR.SWR bit to 1 to reset the ETHERC and EDMAC.

27.3.6 Adjusting Transmission Efficiency by Changing the IPG

The IPG is a non-transmit period between transmit frames. The ETHERC can change the value of the IPG. Transmission efficiency can be increased or decreased by setting the IPGR register. The typical value of the IPG is specified by the IEEE802.3 standard. When changing the setting, confirm that all devices operate normally in the same network.

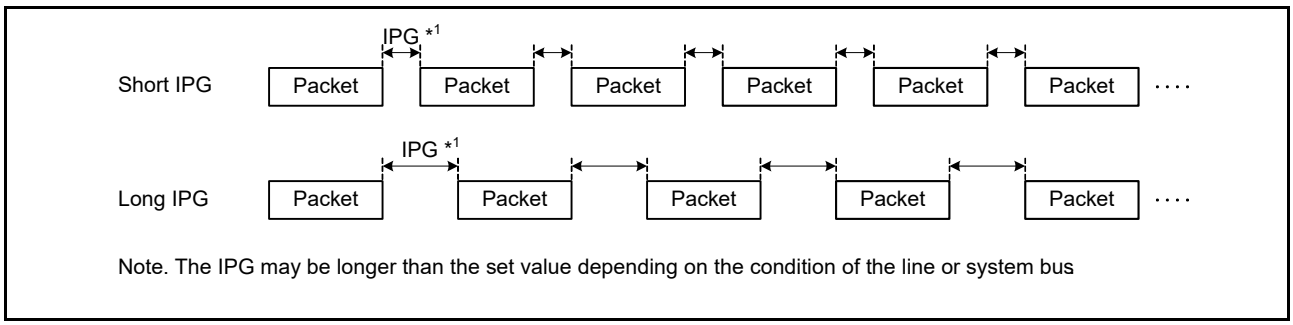


Figure 27.19 Differences in Transmission Efficiency Based on Changes in the IPG

27.3.7 Flow Control

The ETHERC can perform flow control compliant with IEEE802.3x in full-duplex mode, and the receiver and transmitter can be set individually. PAUSE frames can be transmitted automatically or manually.

27.3.7.1 Automatic PAUSE Frame Transmission

When the ECMR.TXF bit is set to 1, automatic PAUSE frame transmission is enabled. A PAUSE frame is automatically transmitted by a PAUSE frame transmit request from the EDMAC. The APR.AP[15:0] bit value is used for the pause_time parameter of the PAUSE frame.
After a PAUSE frame is transmitted, if the EDMAC is still requesting PAUSE frame transmission when the PAUSE time elapses, a PAUSE frame is transmitted again. The maximum number of PAUSE frame retransmissions can be set in the TPAUSER.TPAUSE[15:0] bits. If the maximum number of retransmissions is reached, subsequent PAUSE frames are not transmitted.
Figure 27.20 shows the procedure to set automatic PAUSE frame transmission.

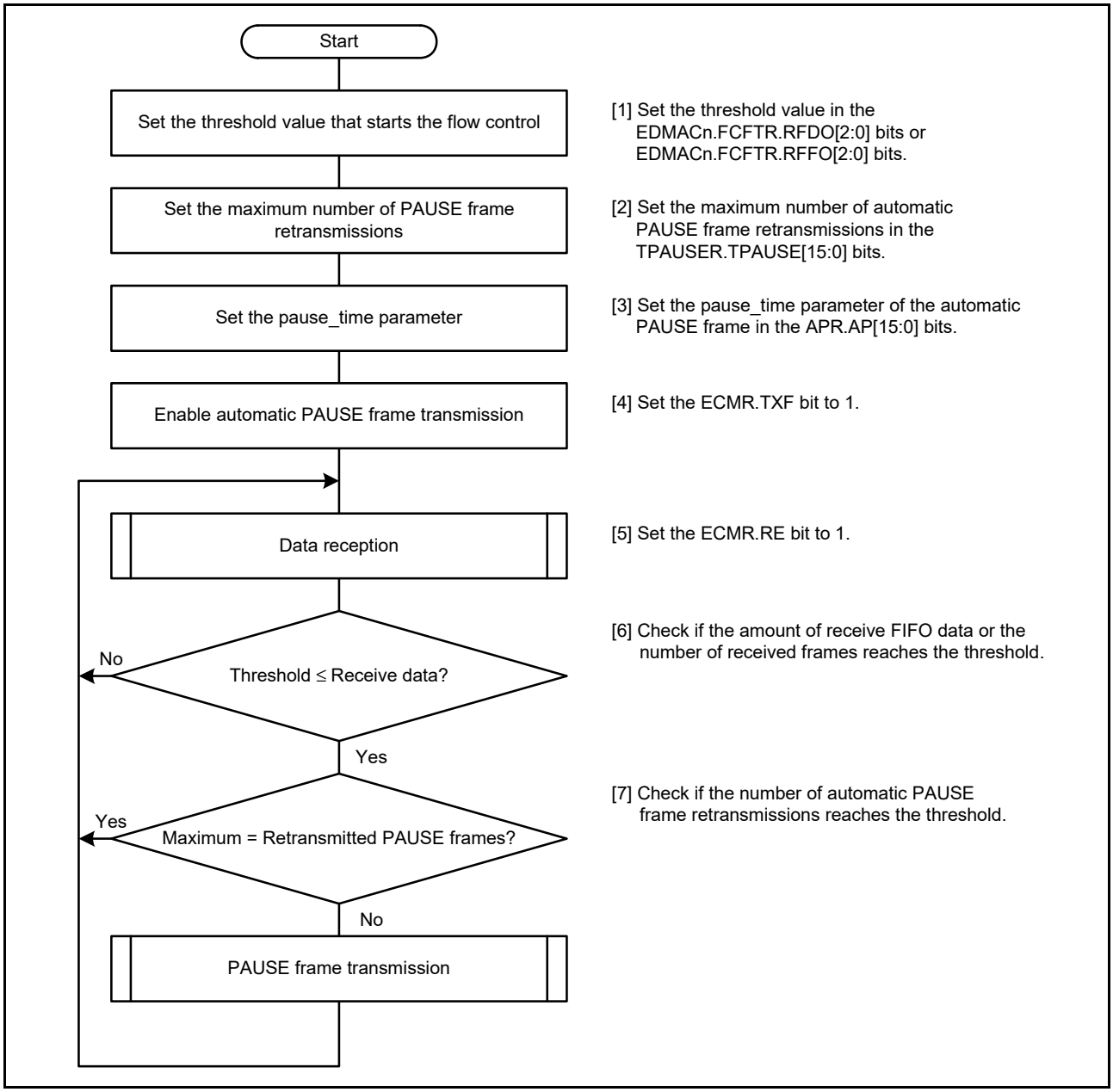


Figure 27.20 Example of Procedure to Set Automatic PAUSE Frame Transmission

27.3.7.2 Manual PAUSE Frame Transmission

A PAUSE frame can be manually transmitted at any time. When writing the pause_time parameter of the PAUSE frame to the MPR.MP[15:0] bits by software, the ETHERC transmits a PAUSE frame once. When transmitting a PAUSE frame more than once, write to the MPR.MP[15:0] bits for each transmission.

27.3.7.3 PAUSE Frame Reception

When setting the ECMR.RXF bit to 1, PAUSE frame detection is enabled. After a PAUSE frame is received, the ETHERC completes transmitting the current frame and waits for the PAUSE time of the received PAUSE frame to elapse before the next frame can be transmitted. Also, the ETHERC increments the RFCF.RPAUSE[7:0] bit value.
However, while waiting for the PAUSE time to elapse, if a PAUSE frame that contains a pause_time parameter of 0 is received and the ECMR.ZPF bit is 1, the ETHERC becomes ready to transmit.

27.4 Interrupts

When a flag in the ECSR register becomes 1 and the corresponding bit in the ECSIPR register is 1, the ETHERC notifies the EDMAC of the status as an interrupt source. After receiving the notification, the EDMAC sets the EDMACn.EESR.ECI flag to 1. When the EDMACn.EESIPR.ECIIP bit is 1, the EDMAC sends an EINTn interrupt request to the CPU. Refer to section 29, DMA Controller for the Ethernet Controller (EDMACa) for details.

27.5 Usage Notes

27.5.1 Conditions for the LCHNG Flag to Become 1

The ECSR.LCHNG flag may become 1 even when the input level of the ETn_LINKSTA pin remains the same. In this case, high is input to the ETn_LINKSTA pin when setting the GPIO.PmnPFS register to assign the ETn_LINKSTA signal to a port or when releasing the ETHERC and EDMAC software reset using the EDMACn.EDMR.SWR bit. The ECSR.LCHNG flag becomes 1 because the ETn_LINKSTA signal in the ETHERC is fixed low regardless of the input level to the external pin while the GPIO does not assign the ETn_LINKSTA signal or during the ETHERC and EDMAC software reset.
To avoid wrongly generating a link signal change interrupt, clear the ECSR.LCHNG flag and then set the ECSIPR.LCHNGIP bit to 1.

27.5.2 Input to the RMIIn_RX_ER Pin While the RMII is Selected

When the width of a reception error signal received from the PHY-LSI is only one cycle of the REF50CKn clock (50 MHz) while the RMII is selected, the signal is not recognized as an error signal.

27.5.3 Handling when Control Information Included a Mismatch

When the setting of the BYPASS.BYPASSn bit of the EPTPC is 0, the SYSR.INFABT flag of the EPTPCn might be set to 1 on reception of an alignment error or a frame error. In such cases, reset the EPTPC, PTPEDMAC, and respective channel of the ETHERC and EDMAC. For the procedure of resetting, see section 27, Procedure of Resetting the Ethernet Controller.

27.5.4 Points to Note when the EPTPC Is not in Use

When the EPTPC is not to be used, set the BYPASS0 and BYPASS1 bits in the BYPASS register of the EPTPC to 1.

27.5.5 Procedure of Resetting the Ethernet Controller

Follow the steps below to reset the ETHERC, EPTPC, and EDMAC.

(a) When the EPTPC is not in use:

(1) Write 0000 0001h to the EDMR register in the respective channel of the EDMAC (resetting the ETHERCn and EDMACn).
(2) Wait for 64 cycles of Bφ until initialization is completed.

(b) When the EPTPC is in use:

(1) Write 0000 0001h to the PTRSTR register of the EPTPC (resetting the EPTPC).
(2) Write 0000 0001h to the EDMR register of the PTPEDMAC (resetting the PTPEDMAC).
(3) Write 0000 0001h to the EDMR register of the respective channel of EDMAC (resetting the ETHERCn and EDMACn).
(4) Wait for 64 cycles of Bφ until initialization is completed.
(5) Write 0000 0000h to the PTRSTR register of the EPTPC (releasing the EPTPC from the reset state).
(6) Wait for 256 cycles of Bφ until the EPTPC is released from the reset state.

28. PTP Module for the Ethernet Controller (EPTPCa)

28.1 Overview

This MCU has an on-chip Precision Time Protocol (PTP) module for the Ethernet controller (EPTPC). The module applies the PTP defined in version 2 of the IEEE 1588-2008 standard to handle timing and synchronization between devices. The EPTPC is composed of synchronization frame processing units (SYNFP0 and SYNFP1), a packet relation controller unit (PRC-TC), and a statistical time correction algorithm unit (STCA).
Use the EPTPC in combination with the on-chip Ethernet controller (ETHERC) and the DMA controller for the PTP Ethernet controller (PTPEDMAC).
Table 28.1 lists the EPTPC specifications, and Figure 28.1 shows the EPTPC configuration.

Table 28.1 EPTPC Specifications

Item	Description
Protocol	Compliant with the Precision Time Protocol (PTP) defined in IEEE 1588
Synchronization frame processing units (SYNFP0 and SYNFP1)	• Transmits and receives PTP messages as a master or slave. • The following four clock devices are supported: Ordinary clock (OC) Boundary clock (BC) End-to-end transparent clock (E2E TC) Peer-to-peer transparent clock (P2P TC) • Calculates the meanPathDelay and offsetFromMaster values defined in IEEE 1588. • Capable of generating a master clock. • Hardware filtering of received multicast packets with a MAC address • Capable of hardware filtering in accord with the type of PTP message • Supports PTP message frames in layer 4 (IPv4 and UDP) and layer 2 (Ethernet frames). • Can be used as a normal Ethernet port when time synchronization is not in use.
Packet relation controller unit (PRC-TC)	• Relaying of received data between Ethernet ports 0 and 1 • Setting the same MAC address for Ethernet ports 0 and 1 allows transmission of data from the two ports or from only one of them. • Store-and-forward method or cut-through method selectable for the relaying of packets
Statistical time correction algorithm unit (STCA)	• Frequency of the clock signal supplied to the statistical time correction algorithm unit is selectable as 20, 25, or 50 MHz. The clock source can be selected from the external clock ETn_SCLKIN (20, 25, or 50 MHz), REF50CKn (50 MHz), or ETn_RX_CLK (25 MHz). Note: When ETn_RX_CLK is selected, ETn_RX_CLK must be fixed to 25 MHz (100 Mbps). • In slave operation, the synchronized state can be indicated from the offsetFromMaster value staying below a previously specified threshold. Additionally, the threshold can be calculated statistically from collected positive and negative gradient values (worst-10 acquisition). • The local clock counter holds corrected time information obtained from a master clock. • The STCA clock can be used as the clock source for generating pulse signals from pulse output timer m (m = 0 to 5).
Interrupt sources	MINT interrupt • Requested when the state of the individual modules is changed. • Requested on rising edges of the pulse signal generated by the pulse output timer. IPLS interrupt • Requested on rising or falling edges of the pulse signal generated by the previously selected pulse output timer group. • Can be requested on every edge or only once.

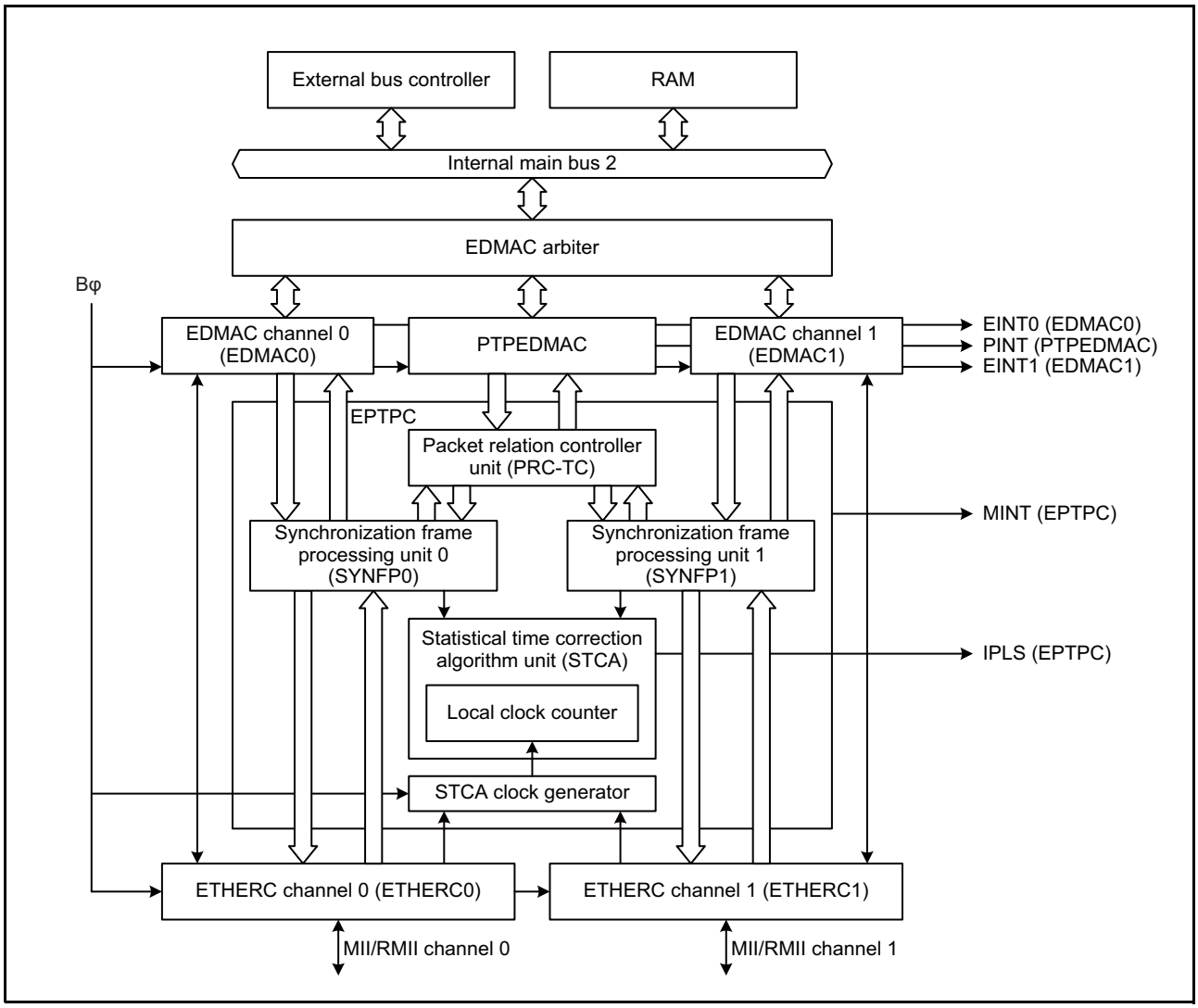


Figure 28.1 EPTPC Configuration

In this section, channels may not be mentioned in overall descriptions of modules that have multiple channels. Examples of the notation are listed in Table 28.2.

Table 28.2 Examples of the Notation

Module Name	Channel	Meaning
SYNFPn module	n = 0, 1	Synchronization processing units 0 and 1 (SYNFP0 and SYNFP1)
Pulse output timer m	m = 0 to 5	Pulse output timer channels 0 to 5

28.1.1 Combination of Clock Device and Ethernet Port

The EPTPC supports operation as three types of clock device: an ordinary clock (OC), boundary clock (BC), or transparent clock (TC). Furthermore, it supports both end-to-end (E2E) and peer-to-peer (P2P) operation for all three clock devices.
Available combinations for usage of Ethernet ports 0 and 1 are listed in Table 28.3.

Table 28.3 Combination of Clock Device and Ethernet Port

Clock Device	Ethernet Port 0		Ethernet Port 1	
EPTPC control disabled	PTP packets are not handled.			
Only Ethernet port 0 is used for handling PTP packets Ordinary Clock (OC)	Master	End-to-end (E2E)	PTP packets are not handled.	
		Peer-to-peer (P2P)		
	Slave	E2E		
		P2P		
Only Ethernet port 1 is used for handling PTP packets (OC)	PTP packets are not handled.		Master	E2E
				P2P
			Slave	E2E
				P2P
Boundary clock (BC)	Master	E2E	Master	E2E
		E2E		P2P
		P2P		E2E
		P2P		P2P
	Master	E2E	Slave	E2E
		E2E		P2P
		P2P		E2E
		P2P		P2P
	Slave	E2E	Master	E2E
		E2E		P2P
		P2P		E2E
		P2P		P2P
Transparent clock (TC)	E2E TC			
	P2P TC			

28.1.2 Frame Format of PTP Messages

The frame format of PTP messages can be selected from the four/two types by setting the FORM0 and FORM1 bits in the SYNFP frame format setting register (SYFORMR).
Figure 28.2 shows the PTP message formats for transmission and reception by the EPTPC.

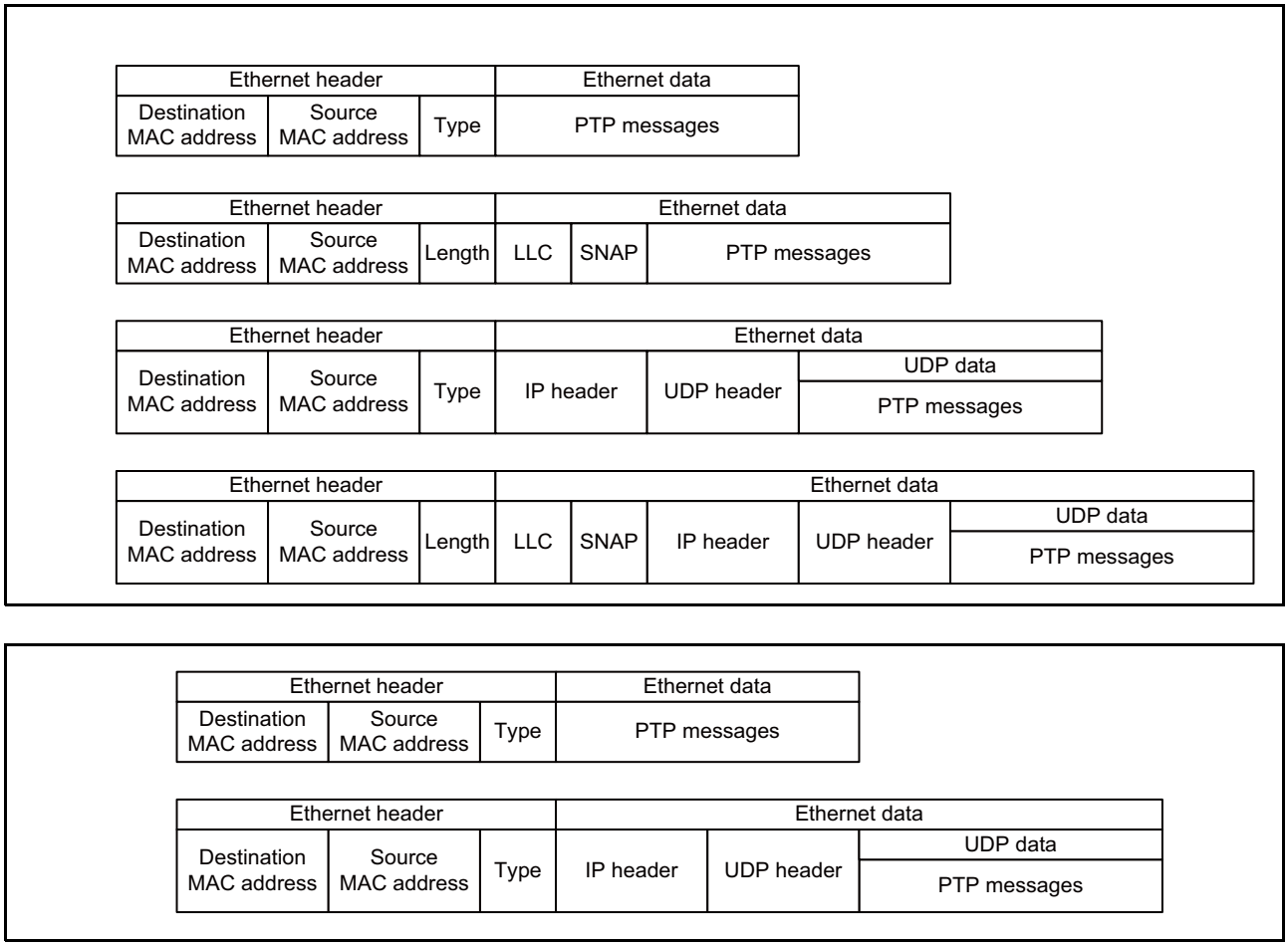


Figure 28.2 Frame Format of PTP Messages

The EPTPC sends PTP messages. The MAC and IP addresses that indicate the destination of each PTP message should usually be multicast addresses defined by IEEE 1588 in accord with the type of the PTP message.
The port number, which must be specified when a PTP message is encapsulated for use with the UDP, is also defined by IEEE 1588 and should be selected in accord with the type of the PTP message.
Table 28.4 lists the types of PTP messages and values that need to be specified for respective Ethernet frame formats.

Table 28.4 Type of PTP Message for Multicast and Information to Specify the Ethernet Frame Format

Type of PTP Message			IEEE802.3 Frame Format (SYFORMR.FORM0 Bit = 1)		Ethernet II Frame Format (SYFORMR.FORM0 Bit = 0)		UDP Port No.*1
			MAC Address	IP Address	MAC Address	Ethertype	
PTP-primary	Event messages	Sync	01-00-5E-00-01-81	224.0.1.129	01-1B-19-00-00-00	88F7h	319
		Delay_Req					
PTP-pdelay		Pdelay_Req	01-00-5E-00-00-6B	224.0.0.107	01-80-C2-00-00-0E		
		Pdelay_Resp					
	General messages	Pdelay_Resp_Follow_Up					320
PTP-primary		Announce	01-00-5E-00-01-81	224.0.1.129	01-1B-19-00-00-00		
		Follow_Up					
		Delay_Resp					
		Signaling					
		Management					

Note 1. The port number must be specified only when a PTP message is encapsulated for use with the UDP (i.e. the SYFORMR.FORM1 bit = 1).

28.1.3 Type of PTP Message and Details of Processing

Table 28.5 and Table 28.6 respectively give details of processing by the EPTPC for receiving and transmitting PTP messages.

Table 28.5 List of PTP Messages for Reception by the EPTPC

Message Type	Message	Details of Processing
Event	Sync	Calculates the value of offsetFromMaster if twoStepFlag in flagField is FALSE.
	Delay_Req	Responds to Delay_Resp.
	Pdelay_Req	Responds to Pdelay_Resp.
	Pdelay_Resp	Calculates the value of meanPathDelay if twoStepFlag in flagField is FALSE.
General	Announce	—
	Follow_Up	Calculates the value of offsetFromMaster if twoStepFlag in flagField of the most recently received Sync message was TRUE and the value of meanPathDelay is fixed.
	Delay_Resp	Calculates the value of meanPathDelay.
	Pdelay_Resp_Follow_Up	Calculates the value of meanPathDelay if twoStepFlag in flagField of the most recently received Pdelay_Resp message was TRUE.
	Management	—
	Signaling	—

Table 28.6 List of PTP Messages for Transmission by the EPTPC

Message Type	Message	Details of Processing
Event	Sync	Sync messages are transmitted at the fixed interval specified by the SYTLIR.SYNC[7:0] bits.
	Delay_Req	Transmission proceeds with an interval from 0 to twice the interval set by the value of the SYTLIR.DREQ[7:0] bits and determined by a random number.
	Pdelay_Req	Pdelay_Req messages are transmitted at the fixed interval specified by the SYTLIR.DREQ[7:0] bits.
	Pdelay_Resp	Transmission of responses to Pdelay_Req.
General	Announce	Announce messages are transmitted at the fixed interval specified by the SYTLIR.ANCE[7:0] bits.
	Follow_Up	—
	Delay_Resp	Transmission of responses to Delay_Req
	Pdelay_Resp_Follow_Up	—
	Management	—
	Signaling	—

28.2 Register Descriptions

Table 28.7 shows the register configuration.

Table 28.7 Register Configuration

Module	Register Name	Abbreviation	Address	Access size
EPTPC	PTP reset register	PTRSTR	E820 4500h	32
	STCA clock select register	STCSELR	E820 4504h	32
	1588 module bypass register	BYPASS	E820 4508h	32
	MINT interrupt source status register	MIESR	E820 5000h	32
	MINT interrupt request enable register	MIEIPR	E820 5004h	32
	IPLS interrupt request enable register	ELIPPR	E820 5010h	32
	IPLS interrupt enable automatic clearing register	ELIPACR	E820 5014h	32
	STCA status register	STSR	E820 5040h	32
	STCA status notification enable register	STIPR	E820 5044h	32
	STCA clock frequency setting register	STCFR	E820 5050h	32
	STCA operating mode register	STMR	E820 5054h	32
	Sync message reception timeout register	SYNTOR	E820 5058h	32
	IPLS interrupt request timer select register	IPTSELR	E820 5060h	32
	MINT interrupt request timer select register	MITSELR	E820 5064h	32
	Time synchronization channel select register	STCHSELR	E820 506Ch	32
	Slave time synchronization start register	SYNSTARTR	E820 5080h	32
	Local clock counter initial value load directive register	LCIVLDR	E820 5084h	32
	Synchronization loss detection threshold register	SYNTDARU	E820 5090h	32
	Synchronization loss detection threshold register	SYNTDARL	E820 5094h	32
	Synchronization detection threshold register	SYNTDBRU	E820 5098h	32
	Synchronization detection threshold register	SYNTDBRL	E820 509Ch	32
	Local clock counter initial value register	LCIVRU	E820 50B0h	32
	Local clock counter initial value register	LCIVRM	E820 50B4h	32
	Local clock counter initial value register	LCIVRL	E820 50B8h	32
	Worst 10 acquisition directive register	GETW10R	E820 5124h	32
	Positive gradient limit register	PLIMITRU	E820 5128h	32
	Positive gradient limit register	PLIMITRM	E820 512Ch	32
	Positive gradient limit register	PLIMITRL	E820 5130h	32
	Negative gradient limit register	MLIMITRU	E820 5134h	32
	Negative gradient limit register	MLIMITRM	E820 5138h	32
	Negative gradient limit register	MLIMITRL	E820 513Ch	32
	Statistical information retention control register	GETINFOR	E820 5140h	32
	Local clock counter	LCCVRU	E820 5170h	32
	Local clock counter	LCCVRM	E820 5174h	32
	Local clock counter	LCCVRL	E820 5178h	32
	Positive gradient worst 10 value register	PW10VRU	E820 5210h	32
	Positive gradient worst 10 value register	PW10VRM	E820 5214h	32
	Positive gradient worst 10 value register	PW10VRL	E820 5218h	32
	Negative gradient worst 10 value register	MW10RU	E820 52D0h	32
	Negative gradient worst 10 value register	MW10RM	E820 52D4h	32
	Negative gradient worst 10 value register	MW10RL	E820 52D8h	32
	Timer start time setting register	TMSTTRU0	E820 5300h	32

Table 28.7 Register Configuration

Module	Register Name	Abbreviation	Address	Access size
EPTPC	Timer start time setting register	TMSTTRL0	E820 5304h	32
	Timer cycle setting register 0	TMCYCR0	E820 5308h	32
	Timer pulse width setting register 0	TMPLSR0	E820 530Ch	32
	Timer start time setting register	TMSTTRU1	E820 5310h	32
	Timer start time setting register	TMSTTRL1	E820 5314h	32
	Timer cycle setting register 1	TMCYCR1	E820 5318h	32
	Timer pulse width setting register 1	TMPLSR1	E820 531Ch	32
	Timer start time setting register	TMSTTRU2	E820 5320h	32
	Timer start time setting register	TMSTTRL2	E820 5324h	32
	Timer cycle setting register 2	TMCYCR2	E820 5328h	32
	Timer pulse width setting register 2	TMPLSR2	E820 532Ch	32
	Timer start time setting register	TMSTTRU3	E820 5330h	32
	Timer start time setting register	TMSTTRL3	E820 5334h	32
	Timer cycle setting register 3	TMCYCR3	E820 5338h	32
	Timer pulse width setting register 3	TMPLSR3	E820 533Ch	32
	Timer start time setting register	TMSTTRU4	E820 5340h	32
	Timer start time setting register	TMSTTRL4	E820 5344h	32
	Timer cycle setting register 4	TMCYCR4	E820 5348h	32
	Timer pulse width setting register 4	TMPLSR4	E820 534Ch	32
	Timer start time setting register	TMSTTRU5	E820 5350h	32
	Timer start time setting register	TMSTTRL5	E820 5354h	32
	Timer cycle setting register 5	TMCYCR5	E820 5358h	32
	Timer pulse width setting register 5	TMPLSR5	E820 535Ch	32
	Timer start register	TMSTARTR	E820 537Ch	32
	PRC-TC status register	PRSR	E820 5400h	32
	PRC-TC status notification enable register	PRIPR	E820 5404h	32
	Channel 0 local MAC address register	PRMACRU0	E820 5410h	32
	Channel 0 local MAC address register	PRMACRL0	E820 5414h	32
	Channel 1 local MAC address register	PRMACRU1	E820 5418h	32
	Channel 1 local MAC address register	PRMACRL1	E820 541Ch	32
	Packet transmission control register	TRNDISR	E820 5420h	32
	Relay mode register	TRNMR	E820 5430h	32
	Cut-through transfer start threshold register	TRNCTTDR	E820 5434h	32
EPTPC0	SYNFP status register	SYSR	E820 5800h	32
	SYNFP status notification enable register	SYIPR	E820 5804h	32
	SYNFP MAC address register	SYMACRU	E820 5810h	32
	SYNFP MAC address register	SYMACRL	E820 5814h	32
	SYNFP LLC-CTL value register	SYLLCCTLR	E820 5818h	32
	SYNFP local IP address register	SYIPADDRR	E820 581Ch	32
	SYNFP specification version setting register	SYSPVRR	E820 5840h	32
	SYNFP domain number setting register	SYDOMR	E820 5844h	32
	Announce message flag field setting register	ANFR	E820 5850h	32
	Sync message flag field setting register	SYNFR	E820 5854h	32
	Delay_Req message flag field setting register	DYRQFR	E820 5858h	32
	Delay_Resp message flag field setting register	DYRPFR	E820 585Ch	32

Table 28.7 Register Configuration

Module	Register Name	Abbreviation	Address	Access size
EPTPC0	SYNFP local clock ID register	SYCIDRL	E820 5860h	32
	SYNFP local clock ID register	SYCIDRU	E820 5864h	32
	SYNFP local port number register	SYPNUMR	E820 5868h	32
	SYNFP register value load directive register	SYRVLDR	E820 5880h	32
	SYNFP reception filter register 1	SYRFL1R	E820 5890h	32
	SYNFP reception filter register 2	SYRFL2R	E820 5894h	32
	SYNFP transmission enable register	SYTRENR	E820 5898h	32
	Master clock ID register	MTCIDL	E820 58A0h	32
	Master clock ID register	MTCIDU	E820 58A4h	32
	Master clock port number register	MTPID	E820 58A8h	32
	SYNFP transmission interval setting register	SYTLIR	E820 58C0h	32
	SYNFP received logMessageInterval value indication register	SYRLIR	E820 58C4h	32
	offsetFromMaster value register	OFMRL	E820 58C8h	32
	offsetFromMaster value register	OFMRU	E820 58CCh	32
	meanPathDelay value register	MPDRU	E820 58D0h	32
	meanPathDelay value register	MPDRL	E820 58D4h	32
	grandmasterPriority field setting register	GMPR	E820 58E0h	32
	grandmasterClockQuality field setting register	GMCQR	E820 58E4h	32
	grandmasterIdentity field setting register	GMIDRU	E820 58E8h	32
	grandmasterIdentity field setting register	GMIDRL	E820 58ECh	32
	currentUtcOffset/timeSource field setting register	CUOTSR	E820 58F0h	32
	stepsRemoved field setting register	SRR	E820 58F4h	32
	PTP-primary message destination MAC address setting register	PPMACRU	E820 5900h	32
	PTP-primary message destination MAC address setting register	PPMACRL	E820 5904h	32
	PTP-pdelay message MAC address setting register	PDMACRU	E820 5908h	32
	PTP-pdelay message MAC address setting register	PDMACRL	E820 590Ch	32
	PTP message Ethertype setting register	PETYPER	E820 5910h	32
	PTP-primary message destination IP address setting register	PPIPR	E820 5920h	32
	PTP-pdelay message destination IP address setting register	PDIPR	E820 5924h	32
	PTP event message TOS setting register	PETOSR	E820 5928h	32
	PTP general message TOS setting register	PGTOSR	E820 592Ch	32
	PTP-primary message TTL setting register	PPTTLR	E820 5930h	32
	PTP-pdelay message TTL setting register	PDTTLR	E820 5934h	32
	PTP event message UDP destination port number setting register	PEUDPR	E820 5938h	32
	PTP general message UDP destination port number setting register	PGUDPR	E820 593Ch	32
	Frame reception filter setting register	FFLTR	E820 5940h	32
	Frame reception filter MAC address 0 setting register	FMAC0RU	E820 5960h	32
	Frame reception filter MAC address 0 setting register	FMAC0RL	E820 5964h	32
	Frame reception filter MAC address 1 setting register	FMAC1RU	E820 5968h	32
	Frame reception filter MAC address 1 setting register	FMAC1RL	E820 596Ch	32
	Asymmetric delay setting register	DASYMRU	E820 59C0h	32
	Asymmetric delay setting register	DASYMRL	E820 59C4h	32
	Timestamp latency setting register	TSLATR	E820 59C8h	32
	SYNFP operation setting register	SYCONFR	E820 59CCh	32
	SYNFP frame format setting register	SYFORMR	E820 59D0h	32

Table 28.7 Register Configuration

Module	Register Name	Abbreviation	Address	Access size
EPTPC0	Response message reception timeout register	RSTOUTR	E820 59D4h	32
EPTPC1	SYNFP status register	SYSR	E820 5C00h	32
	SYNFP status notification enable register	SYIPR	E820 5C04h	32
	SYNFP MAC address register	SYMACRU	E820 5C10h	32
	SYNFP MAC address register	SYMACRL	E820 5C14h	32
	SYNFP LLC-CTL value register	SYLLCCTLR	E820 5C18h	32
	SYNFP local IP address register	SYIPADDRR	E820 5C1Ch	32
	SYNFP specification version setting register	SYSPVRR	E820 5C40h	32
	SYNFP domain number setting register	SYDOMR	E820 5C44h	32
	Announce message flag field setting register	ANFR	E820 5C50h	32
	Sync message flag field setting register	SYNFR	E820 5C54h	32
	Delay_Req message flag field setting register	DYRQFR	E820 5C58h	32
	Delay_Resp message flag field setting register	DYRPFR	E820 5C5Ch	32
	SYNFP local clock ID register	SYCIDRL	E820 5C60h	32
	SYNFP local clock ID register	SYCIDRU	E820 5C64h	32
	SYNFP local port number register	SYPNUMR	E820 5C68h	32
	SYNFP register value load directive register	SYRVLDR	E820 5C80h	32
	SYNFP reception filter register 1	SYRFL1R	E820 5C90h	32
	SYNFP reception filter register 2	SYRFL2R	E820 5C94h	32
	SYNFP transmission enable register	SYTRENR	E820 5C98h	32
	Master clock ID register	MTCIDL	E820 5CA0h	32
	Master clock ID register	MTCIDU	E820 5CA4h	32
	Master clock port number register	MTPID	E820 5CA8h	32
	SYNFP transmission interval setting register	SYTLIR	E820 5CC0h	32
	SYNFP received logMessageInterval value indication register	SYRLIR	E820 5CC4h	32
	offsetFromMaster value register	OFMRL	E820 5CC8h	32
	offsetFromMaster value register	OFMRU	E820 5CCCh	32
	meanPathDelay value register	MPDRU	E820 5CD0h	32
	meanPathDelay value register	MPDRL	E820 5CD4h	32
	grandmasterPriority field setting register	GMPR	E820 5CE0h	32
	grandmasterClockQuality field setting register	GMCQR	E820 5CE4h	32
	grandmasterIdentity field setting register	GMIDRU	E820 5CE8h	32
	grandmasterIdentity field setting register	GMIDRL	E820 5CECh	32
	currentUtcOffset/timeSource field setting register	CUOTSR	E820 5CF0h	32
	stepsRemoved field setting register	SRR	E820 5CF4h	32
	PTP-primary message destination MAC address setting register	PPMACRU	E820 5D00h	32
	PTP-primary message destination MAC address setting register	PPMACRL	E820 5D04h	32
	PTP-pdelay message MAC address setting register	PDMACRU	E820 5D08h	32
	PTP-pdelay message MAC address setting register	PDMACRL	E820 5D0Ch	32
	PTP message Ethertype setting register	PETYPER	E820 5D10h	32
	PTP-primary message destination IP address setting register	PPIPR	E820 5D20h	32
	PTP-pdelay message destination IP address setting register	PDIPR	E820 5D24h	32
	PTP event message TOS setting register	PETOSR	E820 5D28h	32
	PTP general message TOS setting register	PGTOSR	E820 5D2Ch	32
	PTP-primary message TTL setting register	PPTTLR	E820 5D30h	32

Table 28.7 Register Configuration

Module	Register Name	Abbreviation	Address	Access size
EPTPC1	PTP-pdelay message TTL setting register	PDTTLR	E820 5D34h	32
	PTP event message UDP destination port number setting register	PEUDPR	E820 5D38h	32
	PTP general message UDP destination port number setting register	PGUDPR	E820 5D3Ch	32
	Frame reception filter setting register	FFLTR	E820 5D40h	32
	Frame reception filter MAC address 0 setting register	FMAC0RU	E820 5D60h	32
	Frame reception filter MAC address 0 setting register	FMAC0RL	E820 5D64h	32
	Frame reception filter MAC address 1 setting register	FMAC1RU	E820 5D68h	32
	Frame reception filter MAC address 1 setting register	FMAC1RL	E820 5D6Ch	32
	Asymmetric delay setting register	DASYMRU	E820 5DC0h	32
	Asymmetric delay setting register	DASYMRL	E820 5DC4h	32
	Timestamp latency setting register	TSLATR	E820 5DC8h	32
	SYNFP operation setting register	SYCONFR	E820 5DCCh	32
	SYNFP frame format setting register	SYFORMR	E820 5DD0h	32
	Response message reception timeout register	RSTOUTR	E820 5DD4h	32

28.2.1 MINT Interrupt Source Status Register (MIESR)

Address: EPTPC.MIESR E820 5000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	CYC5	CYC4	CYC3	CYC2	CYC1	CYC0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	PRC	SY1	SY0	ST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ST	STCA Status Flag	0: No change in the state of the STCA module 1: A change in the state of the STCA module	R
b1	SY0	SYNFP0 Status Flag	0: No change in the state of the SYNFP0 module 1: A change in the state of the SYNFP0 module	R
b2	SY1	SYNFP1 Status Flag	0: No change in the state of the SYNFP1 module 1: A change in the state of the SYNFP1 module	R
b3	PRC	PRC-TC Status Flag	0: No change in the state of the PRC-TC module 1: A change in the state of the PRC-TC module	R
b15 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b16	CYC0	Pulse Output Timer 0 Rising Edge Detection Flag	0: A rising edge in the periodic pulse signal from pulse output timer 0 is not detected. 1: A rising edge in the periodic pulse signal from pulse output timer 0 is detected.	R/W*1
b17	CYC1	Pulse Output Timer 1 Rising Edge Detection Flag	0: A rising edge in the periodic pulse signal from pulse output timer 1 is not detected. 1: A rising edge in the periodic pulse signal from pulse output timer 1 is detected.	R/W*1
b18	CYC2	Pulse Output Timer 2 Rising Edge Detection Flag	0: A rising edge in the periodic pulse signal from pulse output timer 2 is not detected. 1: A rising edge in the periodic pulse signal from pulse output timer 2 is detected.	R/W*1
b19	CYC3	Pulse Output Timer 3 Rising Edge Detection Flag	0: A rising edge in the periodic pulse signal from pulse output timer 3 is not detected. 1: A rising edge in the periodic pulse signal from pulse output timer 3 is detected.	R/W*1
b20	CYC4	Pulse Output Timer 4 Rising Edge Detection Flag	0: A rising edge in the periodic pulse signal from pulse output timer 4 is not detected. 1: A rising edge in the periodic pulse signal from pulse output timer 4 is detected.	R/W*1
b21	CYC5	Pulse Output Timer 5 Rising Edge Detection Flag	0: A rising edge in the periodic pulse signal from pulse output timer 5 is not detected. 1: A rising edge in the periodic pulse signal from pulse output timer 5 is detected.	R/W*1
b31 to b22	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note 1. Writing 1 clears the flag. Writing 0 does not affect the flag value.

The MIESR register indicates changes in the states of the STCA, SYNFPn, and PRC-TC modules which act as sources of MINT interrupts and the detection of rising edges of pulse output timers m.
For the MINT interrupt, see section 28.4, Interrupts.

28.2.2 MINT Interrupt Request Enable Register (MIEIPR)

Address: EPTPC.MIEIPR E820 5004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	CYCP5	CYCP4	CYCP3	CYCP2	CYCP1	CYCP0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	PR	SY1	SY0	ST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ST	STCA Status Interrupt Request Enable	0: Disables the generation of MINT interrupt requests by the STCA status flag. 1: Enables the generation of MINT interrupt requests by the STCA status flag.	R/W
b1	SY0	SYNFP0 Status Interrupt Request Enable	0: Disables the generation of MINT interrupt requests by the SYNFP0 status flag. 1: Enables the generation of MINT interrupt requests by the SYNFP0 status flag.	R/W
b2	SY1	SYNFP1 Status Interrupt Request Enable	0: Disables the generation of MINT interrupt requests by the SYNFP1 status flag. 1: Enables the generation of MINT interrupt requests by the SYNFP1 status flag.	R/W
b3	PR	PRC-TC Status Interrupt Request Enable	0: Disables the generation of MINT interrupt requests by the PRC-TC status flag. 1: Enables the generation of MINT interrupt requests by the PRC-TC status flag.	R/W
b15 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b16	CYC0	Pulse Output Timer 0 Rising Edge Detection Interrupt Request Enable	0: Disables the generation of MINT interrupt requests in response to detection of a rising edge of pulse output timer 0. 1: Enables the generation of MINT interrupt requests in response to detection of a rising edge of pulse output timer 0.	R/W
b17	CYC1	Pulse Output Timer 1 Rising Edge Detection Interrupt Request Enable	0: Disables the generation of MINT interrupt requests in response to detection of a rising edge of pulse output timer 1. 1: Enables the generation of MINT interrupt requests in response to detection of a rising edge of pulse output timer 1.	R/W
b18	CYC2	Pulse Output Timer 2 Rising Edge Detection Interrupt Request Enable	0: Disables the generation of MINT interrupt requests in response to detection of a rising edge of pulse output timer 2. 1: Enables the generation of MINT interrupt requests in response to detection of a rising edge of pulse output timer 2.	R/W
b19	CYC3	Pulse Output Timer 3 Rising Edge Detection Interrupt Request Enable	0: Disables the generation of MINT interrupt requests in response to detection of a rising edge of pulse output timer 3. 1: Enables the generation of MINT interrupt requests in response to detection of a rising edge of pulse output timer 3.	R/W
b20	CYC4	Pulse Output Timer 4 Rising Edge Detection Interrupt Request Enable	0: Disables the generation of MINT interrupt requests in response to detection of a rising edge of pulse output timer 4. 1: Enables the generation of MINT interrupt requests in response to detection of a rising edge of pulse output timer 4.	R/W
b21	CYC5	Pulse Output Timer 5 Rising Edge Detection Interrupt Request Enable	0: Disables the generation of MINT interrupt requests in response to detection of a rising edge of pulse output timer 5. 1: Enables the generation of MINT interrupt requests in response to detection of a rising edge of pulse output timer 5.	R/W
b31 to b22	—	Reserved	These bits are read as 0. The write value should be 0.	R

The MIEIPR register is used to enable or disable the generation of MINT interrupt requests when MINT interrupt source conditions are satisfied.

28.2.3 IPLS Interrupt Request Enable Register (ELIPPR)

Address: EPTPC.ELIPPR E820 5010h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	PLSN	—	—	—	—	—	—	—	PLSP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	These bits are always read as 1 The write value should always be 1.	R
b7, b6	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b13 to b8	—	Reserved	These bits are always read as 1 The write value should always be 1.	R
b15, b14	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b16	PLSP	Pulse Output Timer Rising Edge Detection IPLS Interrupt Request Enable	0: Disables IPLS interrupt requests due to rising edges of signals from the selected pulse output timer. 1: Enables IPLS interrupt requests due to rising edges of signals from the selected pulse output timer.	R/W
b23 to b17	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b24	PLSN	Pulse Output Timer Falling Edge Detection IPLS Interrupt Request Enable	0: Disables IPLS interrupt requests due to falling edges of signals from the selected pulse output timer. 1: Enables IPLS interrupt requests due to falling edges of signals from the selected pulse output timer.	R/W
b31 to b25	—	Reserved	These bits are always read as 0. The write value should always be 0.	R

The register also enables and disables IPLS interrupts due to rising or falling edges of signals from the pulse output timer selected in the IPTSELR register.

The ELIPACR register can be used to set up the one-time-only output of IPLS interrupt requests.

For the IPLS interrupt, see section 28.4, Interrupts.

28.2.4 IPLS Interrupt Enable Automatic Clearing Register (ELIPACR)

Address: EPTPC.ELIPACR E820 5014h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	PLSN	—	—	—	—	—	—	—	PLSP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b16	PLSP	ELIPPR.PLSP Bit Automatic Clearing	0: Disables automatic clearing of the enable bit for IPLS interrupt requests in response to detection of rising edges of the pulse output timer. 1: Enables automatic clearing of the enable bit for IPLS interrupt requests in response to detection of rising edges of the pulse output timer.	R/W
b23 to b17	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b24	PLSN	ELIPPR.PLSN Bit Automatic Clearing	0: Disables automatic clearing of the enable bit for IPLS interrupt requests in response to detection of falling edges of the pulse output timer. 1: Enables automatic clearing of the enable bit for IPLS interrupt requests in response to detection of falling edges of the pulse output timer.	R/W
b31 to b25	—	Reserved	These bits are always read as 0. The write value should always be 0.	R

The ELIPACR register is used to enable one-time output of an IPLS interrupt request triggered by detecting edges of the periodic pulses of pulse output timer m.
Normally, an IPLS interrupt request is generated at each edge of the periodic pulses of pulse output timer m while the corresponding bit in the ELIPPR register is 1 (enabled).
When a bit in the ELIPPR register is 1 while the corresponding bit in the ELIPACR register is also 1, the bit in the ELIPPR register automatically becomes 0 when an IPLS interrupt request is generated.
For the IPLS interrupt, see section 28.4, Interrupts.

28.2.5 STCA Status Register (STSR)

Address: EPTPC.STSR E820 5040h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	W10D	SYNTOUT	—	SYNCOUT	SYNC
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SYNC	Synchronized State Detection Flag	0: Synchronization is not detected. 1: Synchronization is detected.	R/W*1
b1	SYNCOUT	Synchronization Loss Detection Flag	0: A loss of synchronization is not detected. 1: A loss of synchronization is detected.	R/W*1
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R
b3	SYNTOUT	Sync Message Reception Timeout Detection Flag	0: A Sync message reception timeout is not detected. 1: A Sync message reception timeout is detected.	R/W*1
b4	W10D	Worst 10 Acquisition Completion Flag	0: The worst 10 values have not been acquired yet. 1: The worst 10 values have been acquired.	R/W*1
b31 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: When the SYNSTARTR.STR bit is 0, the value of the corresponding flag remains.
Note 1. Writing 1 clears the flag. Writing 0 does not affect the flag value.

The STSR register indicates the state of the STCA module.

SYNC Flag (Synchronized State Detection Flag)

This flag indicates that synchronization has occurred at least the same as the number of times specified by the STMR.SYTH[3:0] bits in succession when the STMR.ALEN0 bit is 1. When the STMR.ALEN0 bit is 0, the SYNC flag is not set to 1 even if synchronization has occurred more than the specified number of times in succession.

SYNCOUT Flag (Synchronization Loss Detection Flag)

This flag indicates that loss of synchronization has occurred at least the same as the number of times specified by the STMR.DVTH[3:0] bits in succession when the STMR.ALEN0 bit is 1.
Since the time is not synchronized immediately after time synchronization is started (SYNSTARTR.STR bit is set to 1), the SYNCOUT becomes 1 regardless of the STMR.ALEN0 bit setting. When using the SYNTOUT flag, set the SYNTOUT flag to 0 immediately after starting time synchronization.
When the STMR.ALEN0 bit is 0, the SYNCOUT flag does not become 1 even if loss of synchronization has occurred at least the same as the specified number of times in succession after time synchronization is started and the SYNTOUT flag is immediately set to 0.

SYNTOUT Flag (Sync Message Reception Timeout Detection Flag)

This flag indicates that a Sync message has not been received during the period specified by the SYNTOR register when the STMR.ALEN1 bit is 1. The SYNTOUT flag becomes 1 immediately if time synchronization is started (SYNSTARTR.STR bit is set to 1) when no Sync message has been received after the EPTPC starts. When using the SYNTOUT flag, set the SYNTOUT flag to 0 immediately after starting time synchronization.

W10D Flag (Worst 10 Acquisition Completion Flag)

This flag indicates that acquisition of the worst 10 has been completed.

28.2.6 STCA Status Notification Enable Register (STIPR)

Address: EPTPC.STIPR E820 5044h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	W10D	SYNTOUT	—	SYNCOUT	SYNC
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SYNC	SYNC Status Notification Enable	0: Disables notification of the state of STSR.SYNC. 1: Enables notification of the state of STSR.SYNC.	R/W
b1	SYNCOUT	SYNCOUT Status Notification Enable	0: Disables notification of the state of STSR.SYNCOUT. 1: Enables notification of the state of STSR.SYNCOUT.	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R
b3	SYNTOUT	SYNTOUT Status Notification Enable	0: Disables notification of the state of STSR.SYNTOUT. 1: Enables notification of the state of STSR.SYNTOUT.	R/W
b4	W10D	W10D Status Notification Enable	0: Disables notification of the state of STSR.W10D. 1: Enables notification of the state of STSR.W10D.	R/W
b31 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R

The STIPR register specifies whether the MIESR.ST flag does or does not reflect changes in the state of the STCA module.

28.2.7 STCA Clock Frequency Setting Register (STCFR)

Address: EPTPC.STCFR E820 5050h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCF[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	STCF[1:0]	STCA Clock Frequency	b1 b0 0 0: 20 MHz 0 1: 25 MHz 1 0: 50 MHz 1 1: Setting prohibited	R/W
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: Set a value in this register before starting the EDMAC, ETHERC, and PTPEDMAC. Do not change the settings during operations.

The STCFR register is used to select the frequency of the clock source for the STCA module (STCA clock).
The setting in the STCFR register must be for the same frequency as that selected by the setting of the STCSELR register.

STCF[1:0] Bits (STCA Clock Frequency)

These bits select the frequency of the STCA clock.
To enable synchronous control in compliance with IEEE 1588, the STCA clock frequency must be selected from 20 MHz, 25 MHz, or 50 MHz. If the frequency selected by these bits differs from the clock frequency actually input to the STCA module, operation is not guaranteed.

28.2.8 STCA Operating Mode Register (STMR)

Address(es): EPTPC.STMR E820 5054h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	ALEN1	ALEN1	—	—	—	—	DVTH[3:0]				SYTH[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	W10S	—	CMOD	—	—	—	—	—	WINT[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	WINT[7:0]	Worst 10 Acquisition Time	00h: The worst 10 values are not acquired. 01h: Sync message reception: 1 time : FFh: Sync message reception: 255 times	R/W
b12 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b13	CMOD	Time Synchronization Correction Mode	0: Mode 1 1: Mode 2	R/W
b14	—	Reserved	This bit is read as 0. The write value should be 0.	R
b15	W10S	Worst 10 Acquisition Control Select	0: Measurement is started by hardware and the value acquired in the PW10VR or MW10R register is used as the limit for filtering. 1: Measurement is started by the GETW10R.GW10 bit. Also, the value set in the PLIMITR or MLIMITR register is used as the limit for filtering.	R/W
b19 to b16	SYTH[3:0]	Synchronized State Detection Threshold Setting	0h: None*1 1h: 1 time : Fh: 15 times	R/W
b23 to b20	DVTH[3:0]	Synchronization Loss Detection Threshold Setting	0h: None*2	
b27 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R
b28	ALEN0	Alarm Detection Enable 0	0: The STSR.SYNC or SYNCOUT flag is not set to 1 on detection of synchronization or loss of synchronization. 1: The STSR.SYNC or SYNCOUT flag is set to 1 on detection of synchronization or loss of synchronization.	R/W
b29	ALEN1	Alarm Detection Enable 1	0: The STSR.SYNTOUT flag is not set to 1 on detection of the Sync message reception timeout interrupt. 1: The STSR.SYNTOUT flag is not set to 1 on detection of the Sync message reception timeout interrupt.	R/W
b31, b30	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note 1. The STSR.SYNC flag does not become 1 regardless of the ALEN0 bit setting.
Note 2. The STSR.SYNCOUT flag does not become 1 regardless of the ALEN0 bit setting.

The STMR register is used to set the operating mode of the STCA module.

WINT[7:0] Bits (Worst 10 Acquisition Time)

These bits set the time for acquiring the worst 10 gradients (the number of times Sync messages are received). Usually, it is recommended to set the number of times of Sync message receptions to 32 or more.

CMOD Bit (Time Synchronization Correction Mode)

Mode 1 or mode 2 can be selected to correct the local time information when the EPTPC operates as a slave clock. Select the appropriate mode in consideration of system configuration, etc.
Table 28.8 is a summary of the two individual correction modes.

Table 28.8 Types and Features of Correction Mode

Correction Mode	Function	Features	Notes
Mode 1	Mode for correcting the counter every Sync message reception by using the current offsetFromMaster (operation is in mode 1 after the start of correction, and then shifts to the specified mode).	The time information of the master clock is set as the local time information at a time.	(e.g. packets are temporarily being discarded due to a failure of communications).
Mode 2	Mode where the gradient value calculated from offsetFromMaster (worst-10 control) is retained and used in correcting the local time information so that it approximates the time information of the master clock.	Even if calculating offsetFromMaster is not possible, a certain level of synchronization can be guaranteed in this mode (the counter is still corrected from the gradient information).	Establishing synchronization takes longer.

W10S Bit (Worst 10 Acquisition Control Select)

This bit selects the value used for measurement and filtering of worst 10 gradients. When this bit is set to 0, the values acquired in registers PW10VRU, PW10VRM, and PW10VRL and registers MW10RU, MW10RM, and MW10RL are used as the limit for the filter. When the bit is set to 1, the values set in registers PLIMITRU, PLIMITRM, and PLIMITRL and registers MLIMITRU, MLIMITRM, and MLIMITRL are used as the limit for the filter.

SYTH[3:0] Bits (Synchronized State Detection Threshold Setting)

These bits specify a value for the number of consecutive times being below the thresholds will be judged as synchronization. Specifically, the value falling within the thresholds set in registers SYNTDBRU and SYNTDBRL as many times as the value of these bits indicates the synchronized state. When the ALEN0 bit is 1, the STSR.SYNCOUT flag becomes 1.

DVTH[3:0] Bits (Synchronization Loss Detection Threshold Setting)

These bits specify a value for the number of consecutive times exceeding a threshold will be judged as a loss of synchronization. Specifically, synchronization is judged to have been lost when the value in registers SYNTDARU and SYNTDARL is consecutively exceeded as many times as the value of these bits. When the ALEN0 bit is 1, the STSR.SYNCOUT flag becomes 1.

ALEN0 Bit (Alarm Detection Enable 0)

When this bit is 1, the STSR.SYNC or SYNCOUT flag is set to 1 on detection of synchronization or loss of synchronization. When this bit is 0, the SYNC or SYNCOUT flag is not set to 1 even if synchronization or loss of synchronization is detected.

ALEN1 Bit (Alarm Detection Enable 1)

When this bit is 1, the STSR.SYNTOUT flag is set to 1 if a Sync message is not received within the time specified by the SYNTOR register. When this bit is 0, the SYNTOUT flag is not set to 1 even if the reception timeout has occurred.

28.2.9 Sync Message Reception Timeout Register (SYNTOR)

Address(es): EPTPC.SYNTOR E820 5058h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	A Sync message not being received within 1024 × n (ns), where n is the setting, leads to a timeout for reception of Sync messages, leading to the STSR.SYNTOUT flag being set to 1.	R/W

The SYNTOR register is used to specify the timeout period for reception of Sync messages. The timeout period is 1024 times the setting of these bits, in units of nanoseconds.

A Sync message not being received within the period specified by these bits is judged to represent a timeout.

When the SYNTOR register is 0, the STSR.SYNTOUT flag does not become 1.

28.2.10 IPLS Interrupt Request Timer Select Register (IPTSELR)

Address(es): EPTPC.IPTSELR E820 5060h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	IPTSEL5	IPTSEL4	IPTSEL3	IPTSEL2	IPTSEL1	IPTSEL0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	IPTSEL0	Pulse Output Timer 0 Select	0: Pulse output timer 0 is not selected as a source of IPLS interrupt requests. 1: Pulse output timer 0 is selected as a source of IPLS interrupt requests.	R/W
b1	IPTSEL1	Pulse Output Timer 1 Select	0: Pulse output timer 1 is not selected as a source of IPLS interrupt requests. 1: Pulse output timer 1 is selected as a source of IPLS interrupt requests.	R/W
b2	IPTSEL2	Pulse Output Timer 2 Select	0: Pulse output timer 2 is not selected as a source of IPLS interrupt requests. 1: Pulse output timer 2 is selected as a source of IPLS interrupt requests.	R/W
b3	IPTSEL3	Pulse Output Timer 3 Select	0: Pulse output timer 3 is not selected as a source of IPLS interrupt requests. 1: Pulse output timer 3 is selected as a source of IPLS interrupt requests.	R/W
b4	IPTSEL4	Pulse Output Timer 4 Select	0: Pulse output timer 4 is not selected as a source of IPLS interrupt requests. 1: Pulse output timer 4 is selected as a source of IPLS interrupt requests.	R/W
b5	IPTSEL5	Pulse Output Timer 5 Select	0: Pulse output timer 5 is not selected as a source of IPLS interrupt requests. 1: Pulse output timer 5 is selected as a source of IPLS interrupt requests.	R/W
b31 to b6	—	Reserved	These bits are read as 0. The write value should be 0.	R

The IPTSELR register is used to select the pulse output timer that generates IPLS interrupt requests.
Each pulse output timer m takes the clock signal from the STCA as its clock source and produces pulses with a specified period and duty cycle. An IPLS interrupt is requested on rising edges if the ELIPPR.PLSP bit is set to 1 and on falling edges if the PLSN bit in the same register is set to 1. The interrupt request signal becomes the logical OR of the interrupt requests from the given channels when bits for multiple channels in register IPTSELR are set to 1. For the IPLS interrupt, see section 28.4, Interrupts.

28.2.11 MINT Interrupt Request Timer Select Register (MITSELR)

Address(es): EPTPC.MITSELR E820 5064h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	MINTEN5	MINTEN4	MINTEN3	MINTEN2	MINTEN1	MINTEN0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MINTEN0	Pulse Output Timer 0 MINT Interrupt Output Enable	0: Output of rising edges by pulse output timer 0 is not reflected by the MIESR.CYC0 flag as a MINT interrupt source. 1: Output of rising edges by pulse output timer 0 is reflected by the MIESR.CYC0 flag as a MINT interrupt source.	R/W
b1	MINTEN1	Pulse Output Timer 1 MINT Interrupt Output Enable	0: Output of rising edges by pulse output timer 1 is not reflected by the MIESR.CYC1 flag as a MINT interrupt source. 1: Output of rising edges by pulse output timer 1 is reflected by the MIESR.CYC1 flag as a MINT interrupt source.	R/W
b2	MINTEN2	Pulse Output Timer 2 MINT Interrupt Output Enable	0: Output of rising edges by pulse output timer 2 is not reflected by the MIESR.CYC2 flag as a MINT interrupt source. 1: Output of rising edges by pulse output timer 2 is reflected by the MIESR.CYC2 flag as a MINT interrupt source.	R/W
b3	MINTEN3	Pulse Output Timer 3 MINT Interrupt Output Enable	0: Output of rising edges by pulse output timer 3 is not reflected by the MIESR.CYC3 flag as a MINT interrupt source. 1: Output of rising edges by pulse output timer 3 is reflected by the MIESR.CYC3 flag as a MINT interrupt source.	R/W
b4	MINTEN4	Pulse Output Timer 4 MINT Interrupt Output Enable	0: Output of rising edges by pulse output timer 4 is not reflected by the MIESR.CYC4 flag as a MINT interrupt source. 1: Output of rising edges by pulse output timer 4 is reflected by the MIESR.CYC4 flag as a MINT interrupt source.	R/W
b5	MINTEN5	Pulse Output Timer 5 MINT Interrupt Output Enable	0: Output of rising edges by pulse output timer 5 is not reflected by the MIESR.CYC5 flag as a MINT interrupt source. 1: Output of rising edges by pulse output timer 5 is reflected by the MIESR.CYC5 flag as a MINT interrupt source.	R/W
b31 to b6	—	Reserved	These bits are read as 0. The write value should be 0.	R

The MITSELR register is used to select pulse output timer m that generates MINT interrupt requests.
Each pulse output timer m takes the clock signal from the STCA as its clock source and produces pulses with a specified period and duty cycle. A MINT interrupt is requested on rising edges of the pulse signal from the corresponding pulse output timer m if the setting of the MIEIPR.CYCm bit is 1.
For the MINT interrupt, see section 28.4, Interrupts.

28.2.12 Time Synchronization Channel Select Register (STCHSELR)

Address(es): EPTPC.STCHSELR E820 506Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SYSEL
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SYSEL	Timer Information Input Select	0: Time information from the SYNFP0 module is used. 1: Time information from the SYNFP1 module is used.	R/W*1
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note 1. Do not change the value of this bit while the SYNSTARTR.STR bit is 1.

The STCHSELR register selects the time information input to the STCA module.

28.2.13 Slave Time Synchronization Start Register (SYNSTARTR)

Address(es): EPTPC.SYNSTARTR E820 5080h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	STR	Slave Time Synchronization Control	0: Slave time synchronization is stopped. 1: Slave time synchronization is started.	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R

The SYNSTARTR register is used to start or stop time synchronization. This register is used when the EPTPC is operating as a slave node.

28.2.14 Local Clock Counter Initial Value Load Directive Register (LCIVLDR)

Address(es): EPTPC.LCIVLDR E820 5084h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LOAD
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	LOAD	Local Clock Counter Initial Value Load Directive	0: The initial value is not loaded into the local clock counter. 1: The initial value is loaded into the local clock counter.	W*1
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note 1. Do not change the value of this bit while the SYNSTARTR.STR bit is 1.

The LCIVLDR register is used to set the value of registers LCIVRU, LCIVRM, and LCIVRL as the initial value of the local clock counter.

28.2.15 Synchronization Loss Detection Threshold Registers (SYNTDARU, SYNTDARL)

Address(es): EPTPC.SYNTDARU E820 5090h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the higher-order 32 bits of the threshold for detection of loss of synchronization.	R/W

Address(es): EPTPC.SYNTDARL E820 5094h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the lower-order 32 bits of the threshold for detection of loss of synchronization.	R/W

The settings of the SYNTDARU and SYNTDARL registers specify the threshold value for offsetFromMaster to be used in judging loss of synchronization. When setting a threshold value, write the higher-order 32 bits to SYNTDARU and the lower-order 32 bits to SYNTDARL, in that order and in consecutive operations.
If the offsetFromMaster value becomes at least the same as the value specified in SYNTDARU and SYNTDARL, a loss of synchronization is detected. Set a value in SYNTDARU and SYNTDARL in units of nanoseconds.
SYNTDARU and SYNTDARL are not used when the device is operating as a master clock.

28.2.16 Synchronization Detection Threshold Registers (SYNTDBRU, SYNTDBRL)

Address(es): EPTPC.SYNTDBRU E820 5098h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the higher-order 32 bits of the threshold for detection of synchronization.	R/W

Address(es): EPTPC.SYNTDBRL E820 509Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the lower-order 32 bits of the threshold for detection of synchronization.	R/W

The settings of the SYNTDBRU and SYNTDBRL registers specify the threshold value for offsetFromMaster to be used in judging synchronization. When setting a threshold value, write the higher-order 32 bits to SYNTDBRU and the lower-order 32 bits to SYNTDBRL, in that order and in consecutive operations.
If the offsetFromMaster value is less than the value specified in SYNTDBRU and SYNTDBRL, synchronization is detected. Set a value in SYNTDBRU and SYNTDBRL in units of nanoseconds.
SYNTDBRU and SYNTDBRL are not used when the device is operating as a master clock.
If the offsetFromMaster value is less than the value specified in SYNTDBRU and SYNTDBRL, synchronization is detected. Set a value in SYNTDBRU and SYNTDBRL in units of nanoseconds.
SYNTDBRU and SYNTDBRL are not used when the device is operating as a master clock.

28.2.17 Local Clock Counter Initial Value Registers (LCIVRU, LCIVRM, LCIVRL)

Address(es): EPTPC.LCIVRU E820 50B0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the higher-order 16 bits of the integer portion of the initial value for the local clock counter.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Address(es): EPTPC.LCIVRM E820 50B4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the lower-order 32 bits of the integer portion of the initial value for the clock counter.	R/W

Address(es): EPTPC.LCIVRL E820 50B8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the fractional portion of the initial value of the clock counter in nanoseconds.	R/W

Registers LCIVRU, LCIVRM, and LCIVRL specify the initial value in seconds of the clock counter. When setting an initial value, write the higher-order 16 bits of the integer portion to LCIVRU, the lower-order 32 bits of the integer portion to LCIVRM, and the fractional portion in nanoseconds to LCIVRL, in that order and in consecutive operations. The value of registers LCIVRU, LCIVRM, and LCIVRL can be used as the initial value of the local clock counter. When setting the value of registers LCIVRU, LCIVRM, and LCIVRL in the local clock counter, set the LCIVLDR.LOAD bit to 1.

Example) When 2.000000025s is to be set as the initial value, write the following values to each of the registers.
LCIVRU: 0000 0000h
LCIVRM: 0000 0002h
LCIVRL: 0000 0019h

28.2.18 Worst 10 Acquisition Directive Register (GETW10R)

Address(es): EPTPC.GETW10R E820 5124h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GW10
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	GW10	Worst 10 Acquisition Directive	0: The worst-10 values are not acquired. 1: Starts acquisition of the worst-10 values.	R/W*1
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Do not set this bit to 1 while the STMR.W10S bit is 0.

The GETW10R register is used in software control to start calculation of gradient values for use in selecting the worst-10 values.

A gradient value is the amount by which the timer counter of a slave is incremented when a given interval elapses. Setting the GW10 bit to 1 while the value of the STMR.W10S bit is 1 selects calculation of a gradient value by the EPTPC each time it receives a Sync message. Gradient values are calculated the number of times specified by the STMR.WINT[7:0] bits. The GW10 bit is cleared to 0 on completion of this number of calculations. The GETW10R register is not used in operation as a master clock.

28.2.19 Positive Gradient Limit Registers (PLIMITRU, PLIMITRM, PLIMITRL)

Address(es): EPTPC.PLIMITRU E820 5128h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b30 to b0	—	—	These bits hold the setting for the higher-order 31 bits of the limit for the positive gradient.	R/W
b31	—	Reserved	This bit is read as 0. The write value should be 0.	R

Address(es): EPTPC.PLIMITRM E820 512Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the middle-order 32 bits of the limit for the positive gradient.	R/W

Address(es): EPTPC.PLIMITRL E820 5130h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

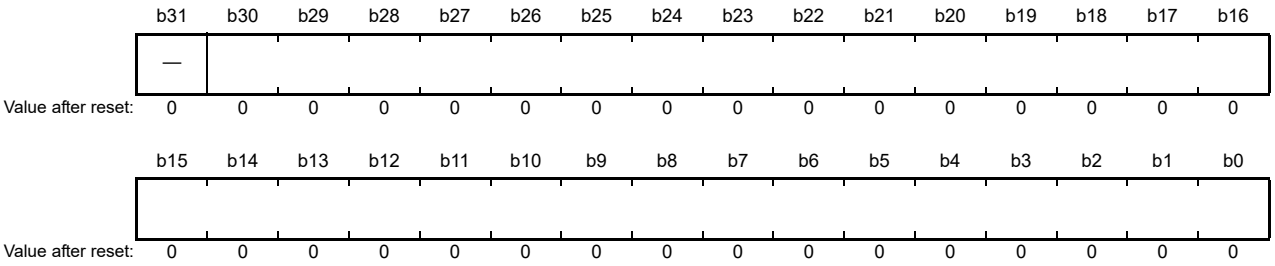
Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the lower-order 32 bits of the limit for the positive gradient.	R/W

Registers PLIMITRU, PLIMITRM, and PLIMITRL are used to specify an upper limit on gradient (= positive gradient) for use in time synchronization. When setting an upper limit, write values consecutively to PLIMITRU, PLIMITRM, and PLIMITRL in that order.
The gradients that is at least the same as the value specified in registers PLIMITRU, PLIMITRM, and PLIMITRL are not used in time synchronization. Registers PLIMITRU, PLIMITRM, PLIMITRL are not used when the device is operating as a master clock. Registers PLIMITRU, PLIMITRM, and PLIMITRL are effective while the STMR.CMOD and W10S bits are 1.
The gradient value to be set in the register is calculated by using the following expression.
PLIMITRU, PLIMITRM, and PLIMITRL register values = A (s)/T (s) × 232
A: The time (s) by which the slave local clock counter advances during the interval between received Sync messages
T: The actual time (s) between received Sync messages

For example, if the interval between Sync messages is 0.5 seconds and the local clock counter advances by 0.7 seconds during that time, and this is to be set as the limit, then the setting for PLIMITR = 0.7/0.5 × 232 = 6 012 954 214 = 1 6666 6666h, and the settings for the individual registers are as follows: PLIMITRU = 0000 0000h, PLIMITRM = 0000 0001h, and PLIMITRL = 6666 6666h.
The minimum setting depends on the STCA clock frequency as the clock source for counting by the local clock counter. For example, if the STCA clock frequency is 50 MHz, then the minimum allowable setting for registers PLIMITRU, PLIMITRM, and PLIMITRL = (1/50 (MHz)) (s)/0.5 (s) × 232 = 172 = ACh, and the settings for the individual registers are as follows: PLIMITRU = 0000 0000h, PLIMITRM = 0000 0000h, and PLIMITRL = 0000 00ACh.
The gradient limit values to be set are valid when time synchronization correction mode is mode 2 (the STMR.CMOD bit is 1) and the gradient is controlled by software (the STMR.W10S bit is 1).

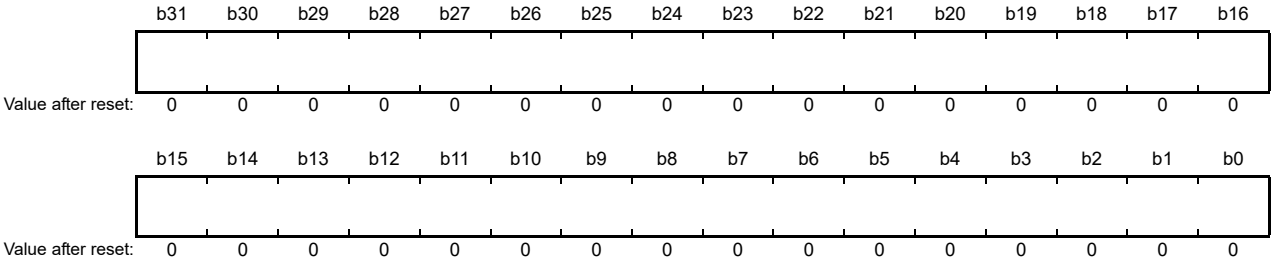
28.2.20 Negative Gradient Limit Registers (MLIMITRU, MLIMITRM, MLIMITRL)

Address(es): EPTPC.MLIMITRU E820 5134h



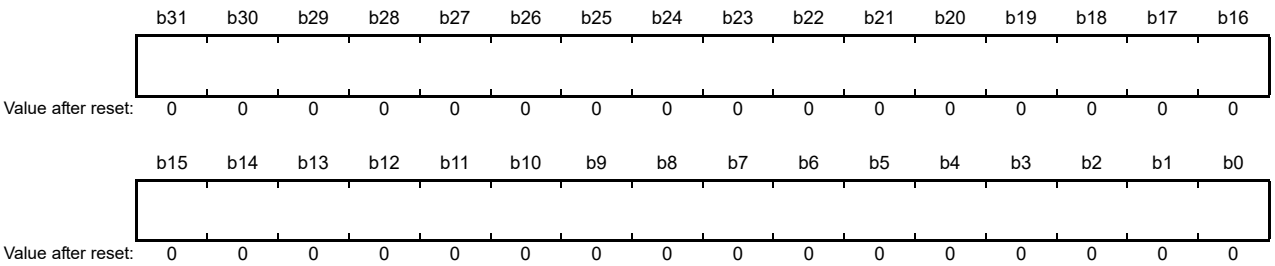
Bit	Symbol	Bit Name	Description	R/W
b30 to b0	—	—	These bits hold the setting for the higher-order 31 bits of the limit for the negative gradient.	R/W
b31	—	Reserved	This bit is read as 0. The write value should be 0.	R

Address(es): EPTPC.MLIMITRM E820 5138h



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the middle-order 32 bits of the limit for the negative gradient.	R/W

Address(es): EPTPC.MLIMITRL E820 513Ch



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the lower-order 32 bits of the limit for the negative gradient.	R/W

Registers MLIMITRU, MLIMITRM, and MLIMITRL are used to specify a lower limit for the gradient (= negative gradient) for use in time synchronization. Use two's complement to set a lower limit. When setting a lower limit, write values consecutively to MLIMITRU, MLIMITRM, and MLIMITRL in that order.
The gradients that are less than the value specified in registers MLIMITRU, MLIMITRM, and MLIMITRL are not used in time synchronization. Registers MLIMITRU, MLIMITRM, and MLIMITRL are not used when the device is operating as a master clock. Registers MLIMITRU, MLIMITRM, and MLIMITRL are effective while the STMR.CMOD and W10S bits are 1.
The procedure for setting a value and the minimum value that can be set are the same as for registers PLIMITRU, PLIMITRM, and PLIMITRL.

28.2.21 Statistical Information Retention Control Register (GETINFOR)

Address(es): EPTPC.GETINFOR E820 5140h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INFO
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	INFO	Information Retention Control	[When written] 0: Has no effects. 1: Information is retained. [When read] 0: Information retention is completed. 1: Processing for information retention is in progress. Once information fetching is directed, values of various statistical information read before completion of information fetching are not guaranteed.	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R

The GETINFOR register controls retention of the following statistical information.

- Registers LCCVRU, LCCVRM, and LCCVRL
- Registers PW10VRU, PW10VRM, and PW10VRL
- Registers MW10RU, MW10RM, and MW10RL

The only value that is writable to the INFO bit is 1. When setting a value in registers PW10VRU, PW10VRM, and PW10VRL, or registers MW10RU, MW10RM, and MW10RL, set the INFO bit to 1 only while the STMR.W10S bit is 1. If the INFO bit is set to 1 before acquisition of the worst-10 values is completed, whether the information retained in registers PW10VRU, PW10VRM, and PW10VRL and registers MW10RU, MW10RM, MW10RL is correct or not is not guaranteed. Use the GETW10R.GW10 bit to confirm that acquisition has been completed before setting the INFO bit to 1. The INFO bit is automatically returned to 0 on completion of information fetching.

28.2.22 Local Clock Counters (LCCVRU, LCCVRM, LCCVRL)

Address(es): EPTPC.LCCVRU E820 5170h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	—	These bits are for reading the higher-order 16 bits of the integer portion of the local clock counter's value.	R
b31 to b16	—	Reserved	These bits are read as 0.	R

Address(es): EPTPC.LCCVRM E820 5174h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits are for reading the lower-order 32 bits of the integer portion of the local clock counter's value.	R

Address(es): EPTPC.LCCVRL E820 5178h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits are for reading the fractional portion of the local clock counter's value (in nanoseconds).	R

Registers LCCVRU, LCCVRM, and LCCVRL indicate the local clock counter's value.
When the GETINFOR.INFO bit is set to 1, the value of the local clock counter at that time is stored in registers LCCVRU, LCCVRM, and LCCVRL. The higher-order 16 bits of the integer portion in seconds are saved in LCCVRU, the lower-order 32 bits of the integer portion in seconds are saved in LCCVRM, and the fractional portion in nanoseconds is saved in LCCVRL.
For example, if the local time information is 14:25, 44 seconds, 10 milliseconds, 23 microseconds, and 39 nanoseconds, registers LCCVRU, LCCVRM, and LCCVRL have 14 × 3600 + 25 × 60 + 44 = 51944 (s) = 0000 0000 CAE8h as the setting of the higher-order 48 bits and 10 × 106 + 23 × 103 + 39 = 10023039 (ns) = 0098 F07Fh as the setting of the lower-order 32 bits.

28.2.23 Positive Gradient Worst 10 Value Registers (PW10VRU, PW10VRM, PW10VRL)

Address(es): EPTPC.PW10VRU E820 5210h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits are for reading the higher-order 32 bits of the positive gradient value.	R

Address(es): EPTPC.PW10VRM E820 5214h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits are for reading the middle-order 32 bits of the positive gradient value.	R

Address(es): EPTPC.PW10VRL E820 5218h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits are for reading the lower-order 32 bits of the positive gradient value.	R

Registers PW10VRU, PW10VRM, and PW10VRL indicate the worst 10 of the positive gradient values.
When the GETINFOR.INFO bit is set to 1, the worst 10 value at that time is stored in registers PW10VRU, PW10VRM, and PW10VRL.
The format of the worst 10 gradients stored in registers PW10VRU, PW10VRM, and PW10VRL are the same as for registers PLIMITRU, PLIMITRM, and PLIMITRL. For details on the format of the values, see the description of the PLIMITR registers.
Registers PW10VRU, PW10VRM, and PW10VRL are not used when the device is used as a master clock.

28.2.24 Negative Gradient Worst 10 Value Registers (MW10RU, MW10RM, MW10RL)

Address(es): EPTPC.MW10RU E820 52D0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits are for reading the higher-order 32 bits of the negative gradient value.	R

Address(es): EPTPC.MW10RM E820 52D4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits are for reading the middle-order 32 bits of the negative gradient value.	R

Address(es): EPTPC.MW10RL E820 52D8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits are for reading the lower-order 32 bits of the negative gradient value.	R

Registers MW10RU, MW10RM, and MW10RL indicate the worst 10 of the negative gradient values.
When the GETINFOR.INFO bit is set to 1, the worst 10 value at that time is stored in registers MW10RU, MW10RM, and MW10RL.
The format of the worst 10 gradients stored in registers MW10RU, MW10RM, and MW10RL are the same as for registers MLIMITRU, MLIMITRM, and MLIMITRL. For details on the format of the values, see the description of the MLIMITR registers.
Registers MW10RU, MW10RM, and MW10RL are not used when the device is used as a master clock.

28.2.25 Timer Start Time Setting Registers (TMSTTRUm, TMSTTRLm) (m = 0 to 5)

Address(es): EPTPC.TMSTTRU0 E820 5300h, EPTPC.TMSTTRU1 E820 5310h, EPTPC.TMSTTRU2 E820 5320h, EPTPC.TMSTTRU3 E820 5330h, EPTPC.TMSTTRU4 E820 5340h, EPTPC.TMSTTRU5 E820 5350h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the higher-order 32 bits of the start time of the pulse output timer in nanoseconds	R/W

Address(es): EPTPC.TMSTTRL0 E820 5304h, EPTPC.TMSTTRL1 E820 5314h, EPTPC.TMSTTRL2 E820 5324h, EPTPC.TMSTTRL3 E820 5334h, EPTPC.TMSTTRL4 E820 5344h, EPTPC.TMSTTRL5 E820 5354h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the lower-order 32 bits of the start time of the pulse output timer in nanoseconds.	R/W

Registers TMSTTRUm and TMSTTRLm are used to set the start time of pulse output timer m.
Set the start time of pulse output timer m (64 bits) in units of nanoseconds. Though the setting is in units of nanoseconds, the start time of pulse output timer m depends on the resolution of the STCA clock.
For example, if the STCA clock is running at 50 MHz, one cycle takes 20 ns, so the time at which the timer starts may differ from the time set in registers TMSTTRUm and TMSTTRLm by up to 20 ns.
When writing to registers TMSTTRUm and TMSTTRLm, write values consecutively in order of TMSTTRUm and then TMSTTRLm while the TMSTARTR.ENm bit is 0.
Note, however, that the format for setting times in registers TMSTTRUm and TMSTTRLm differs from that described in section 28.2.22, Local Clock Counters (LCCVRU, LCCVRM, LCCVRL).

28.2.26 Timer Cycle Setting Registers m (TMCYCRm) (m = 0 to 5)

Address(es): EPTPC.TMCYCR0 E820 5308h, EPTPC.TMCYCR1 E820 5318h, EPTPC.TMCYCR2 E820 5328h, EPTPC.TMCYCR3 E820 5338h, EPTPC.TMCYCR4 E820 5348h, EPTPC.TMCYCR5 E820 5358h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—														
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b29 to b0	—	—	These bits set the cycle of the pulse output timer in nanoseconds. Set a value that is equivalent to at least four cycles of the STCA clock.	R/W
b31 to b30	—	Reserved	These bits are read as 0. The write value should be 0.	R

A TMCYCRm register specifies the period of the output signal generated by the corresponding pulse output timer m. Set a value in nanoseconds that is equivalent to at least four cycles of the STCA clock while the value of the TMSTARTR.ENm bit is 0.

Though the settings of TMCYCRm registers are in units of nanoseconds, the period of the output signal generated by pulse output timer m and the time at which the timer starts depend on the period of the STCA clock. For example, one cycle of the STCA clock running at 50 MHz takes 20 ns, so the clock source for counting by pulse output timer m may differ from the period set in the TMCYCRm registers by up to 19 ns. The SYNFP module handles calculations to correct this difference.

For example, if the setting for the timer period is 81 ns and the STCA clock is running at 50 MHz, the only available settings close to the actual timer period are for 80 ns or 100 ns. By setting the timer period in the SYNFP module to 80 ns for 19 and to 100 ns for 1 of every 20 cycles, we are able to adjust the average period to 81 ns.

$(80\ (\text{ns}) \times 19 + 100\ (\text{ns}) \times 1)/20 = 81\ (\text{ns})$

The minimum value that can be set in a TMCYCRm register is four cycles of the STCA clock. For example, if the STCA clock is running at 50 MHz, the minimum setting corresponds to 80 ns. Timer operation is not guaranteed if a value set in one of these registers is less than this value.

28.2.27 Timer Pulse Width Setting Register m (TMPLSRm) (m = 0 to 5)

Address(es): EPTPC.TMPLSR0 E820 530Ch, EPTPC.TMPLSR1 E820 531Ch, EPTPC.TMPLSR2 E820 532Ch, EPTPC.TMPLSR3 E820 533Ch, EPTPC.TMPLSR4 E820 534Ch, EPTPC.TMPLSR5 E820 535Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—													
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b28 to b0	—	—	These bits set the width at high level of the pulse signal from the timer in nanoseconds. Set a value that is equivalent to at least two cycles of the STCA clock.	R/W
b31 to b29	—	Reserved	These bits are read as 0. The write value should be 0.	R

A TMPLSRm register specifies the width at high level of the output signal generated by the corresponding pulse output timer m.

When the TMSTARTR.ENm bit is 0, set a value corresponding to a time no shorter than two cycles of the STCA clock in units of nanoseconds.

The higher-order 3 bits of the TMPLSRm register are reserved. These bits are read as 000b. When writing, write 000b to these bits.

Though the settings of the TMPLSRm register are in units of nanoseconds, the width at high level of the signal from the timer depends on the period of the STCA clock. The method for correcting the width at high level of the signal from the timer is the same as that for correcting the timer periods set by the TMCYCRm register.

28.2.28 Timer Start Register (TMSTARTR)

Address(es): EPTPC.TMSTARTR E820 537Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	EN5	EN4	EN3	EN2	EN1	EN0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	EN0	Pulse Output Timer 0 Start	1: Starts pulse output timer 0. 1: Starts pulse output timer 0.	R/W
b1	EN1	Pulse Output Timer 1 Start	0: Stops pulse output timer 1. 1: Starts pulse output timer 1.	R/W
b2	EN2	Pulse Output Timer 2 Start	0: Stops pulse output timer 2. 1: Starts pulse output timer 2.	R/W
b3	EN3	Pulse Output Timer 3 Start	0: Stops pulse output timer 3. 1: Starts pulse output timer 3.	R/W
b4	EN4	Pulse Output Timer 4 Start	0: Stops pulse output timer 4. 1: Starts pulse output timer 4.	R/W
b5	EN5	Pulse Output Timer 5 Start	0: Stops pulse output timer 5. 1: Starts pulse output timer 5.	R/W
b31 to b6	—	Reserved	These bits are read as 0. The write value should be 0.	R

The TMSTARTR register is used to start and stop pulse output timer m.

28.2.29 PRC-TC Status Register (PRSR)

Address(es): EPTPC.PRSR E820 5400h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	URE1	URE0	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	MACE	—	—	—	—	OVRE3	OVRE2	OVRE1	OVRE0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OVRE0	Relay Packet Overflow Detection Flag 0	0: No overflow in transfer of data from SYNFP1 to PTPEDMAC 1: An overflow has been detected in transfer of data from SYNFP1 to PTPEDMAC.	R/W*1
b1	OVRE1	Relay Packet Overflow Detection Flag 1	0: No overflow in transfer of data from SYNFP0 to PTPEDMAC 1: An overflow has been detected in transfer of data from SYNFP0 to PTPEDMAC.	R/W*1
b2	OVRE2	Relay Packet Overflow Detection Flag 2	0: No overflow in transfer of data from SYNFP1 to SYNFP0 1: An overflow has been detected in transfer of data from SYNFP1 to SYNFP0.	R/W*1
b3	OVRE3	Relay Packet Overflow Detection Flag 3	0: No overflow in transfer of data from SYNFP0 to SYNFP1 1: An overflow has been detected in transfer of data from SYNFP0 to SYNFP1.	R/W*1
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b8	MACE	Source MAC Address Mismatch Detection Flag	0: A MAC address mismatch has not been detected. 1: A MAC address mismatch has been detected.	R/W*1
b15 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R
b27 to b16	—	Reserved	These bits are read as undefined. The write value should be 0.	R/W
b28	URE0	Relay Packet Underflow Detection Flag 0	0: No underflow in transfer of data from SYNFP1 to SYNFP0 1: An underflow has been detected in transfer of data from SYNFP1 to SYNFP0.	R/W*1
b29	URE1	Relay Packet Underflow Detection Flag 1	0: No underflow in transfer of data from SYNFP0 to SYNFP1 1: An underflow has been detected in transfer of data from SYNFP0 to SYNFP1.	R/W*1
b31, b30	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note 1. Writing 1 clears the flag. Writing 0 does not affect the flag value.

The PRSR register indicates the state of the PRC-TC module.
The PRC-TC module transmits and receives Ethernet packets through Ethernet port 0, Ethernet port 1, and the PTPEDMAC.
The PRSR register indicates the states as overflows, underflows, and non-matching MAC addresses in the transmission and reception of Ethernet packets.

OVRE0 Flag (Relay Packet Overflow Detection Flag 0)

This flag indicates that the FIFO buffer has overflowed while packets were being transferred from the SYNFP1 module to the PTPEDMAC. The data received in the PTPEDMAC may be incorrect if the setting of the OVRE0 flag is 1.

OVRE1 Flag (Relay Packet Overflow Detection Flag 1)

This flag indicates that the FIFO buffer has overflowed while packets were being transferred from the SYNFP0 module to the PTPEDMAC. The data received in the PTPEDMAC may be incorrect if the setting of the OVRE1 flag is 1.

OVRE2 Flag (Relay Packet Overflow Detection Flag 2)

This flag indicates that the FIFO buffer has overflowed while packets were being transferred from the SYNFP1 module to the SYNFP0 module.

OVRE3 Flag (Relay Packet Overflow Detection Flag 3)

This flag indicates that the FIFO buffer has overflowed while packets were being transferred from the SYNFP0 module to the SYNFP1 module.

MACE Flag (Source MAC Address Mismatch Detection Flag)

This flag becomes 1 if the source MAC address of a packet for transmission from the PTPEDMAC matches none of registers PRMACRU0, PRMACRL0, PRMACRU1, and PRMACRL1.

URE0 Flag (Relay Packet Underflow Detection Flag 0)

This flag indicates that the FIFO buffer has underflowed while packets were being transferred from the SYNFP1 module to the SYNFP0 module.

URE1 Flag (Relay Packet Underflow Detection Flag 1)

This flag indicates that the FIFO buffer has underflowed while packets were being transferred from the SYNFP0 module to the SYNFP1 module.

28.2.30 PRC-TC Status Notification Enable Register (PRIPR)

Address(es): EPTPC.PRIPR E820 5404h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	URE1	URE0	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	MACE	—	—	—	—	OVRE3	OVRE2	OVRE1	OVRE0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OVRE0	PRSR.OVRE0 Status Notification Enable	0: Disables notification of the state of PRSR.OVRE0. 1: Enables notification of the state of PRSR.OVRE0.	R/W
b1	OVRE1	PRSR.OVRE1 Status Notification Enable	0: Disables notification of the state of PRSR.OVRE1. 1: Enables notification of the state of PRSR.OVRE1.	R/W
b2	OVRE2	PRSR.OVRE2 Status Notification Enable	0: Disables notification of the state of PRSR.OVRE2. 1: Enables notification of the state of PRSR.OVRE2.	R/W
b3	OVRE3	PRSR.OVRE3 Status Notification Enable	0: Disables notification of the state of PRSR.OVRE3. 1: Enables notification of the state of PRSR.OVRE3.	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b8	MACE	PRSR.MACE Status Notification Enable	0: Disables notification of the state of PRSR.MACE 1: Enables notification of the state of PRSR.MACE	R/W
b27 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R
b28	URE0	PRSR.URE0 Status Notification Enable	0: Disables notification of the state of PRSR.URE0. 1: Enables notification of the state of PRSR.URE0.	R/W
b29	URE1	PRSR.URE1 Status Notification Enable	0: Disables notification of the state of PRSR.URE1. 1: Enables notification of the state of PRSR.URE1.	R/W
b31, b30	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: Refer to section 28.4, Interrupts, for details on interrupt system.

The PRIPR register specifies whether the MIESR.PRC flag does or does not reflect changes in the state of the PRC-TC module.

28.2.31 Channel 0 Local MAC Address Registers (PRMACRU0, PRMACRL0)

Address(es): EPTPC.PRMACRU0 E820 5410h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the higher-order 24 bits of the local MAC address for Ethernet port 0.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

Address(es): EPTPC.PRMACRL0 E820 5414h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the lower-order 24 bits of the local MAC address for Ethernet port 0.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

Registers PRMACRU0 and PRMACRL0 are used to set the local MAC address for Ethernet port 0. Set the higher-order 24 bits and the lower-order 24 bits of the MAC address in PRMACRU0 and PRMACRL0, respectively.
These registers are used in transmission from the PTPEDMAC to send frames from transmission sources with MAC addresses matching the value of this register to Ethernet port 0. Set registers PRMACRU0 and PRMACRL0 to the same value as registers SYMACRU and SYMACRL for the EPTPC0 module.
Rewrite registers PRMACRU0 and PRMACRL0 before starting the EDMAC, ETHERC, and PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.32 Channel 1 Local MAC Address Registers (PRMACRU1, PRMACRL1)

Address(es): EPTPC.PRMACRU1 E820 5418h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the higher-order 24 bits of the local MAC address for Ethernet port 1.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

Address(es): EPTPC.PRMACRL1 E820 541Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the lower-order 24 bits of the local MAC address for Ethernet port 1.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

Registers PRMACRU1 and PRMACRL1 are used to set the local MAC address for Ethernet port 1. Set the higher-order 24 bits and the lower-order 24 bits of the MAC address in registers PRMACRU1 and PRMACRL1, respectively.
These registers are used in transmission from the PTPEDMAC to send frames from transmission sources with MAC addresses matching the value of this register to Ethernet port 1. Set registers PRMACRU1 and PRMACRL1 to the same value as registers SYMACRU and SYMACRL for the EPTPC1 module.
Rewrite registers PRMACRU1 and PRMACRL1 before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.33 Packet Transmission Control Register (TRNDISR)

Address(es): EPTPC.TRNDISR E820 5420h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TDIS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	TDIS[1:0]	Packet Transmission Control	b1 b0 0 0: PTP packets are transmitted through both Ethernet port 0 and Ethernet port 1. 0 1: PTP packets are only transmitted through Ethernet port 0. 1 0: PTP packets are only transmitted through Ethernet port 1. 1 1: Setting prohibited	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

The TRNDISR register controls transmission of PTP packets when Ethernet port 0 and Ethernet port 1 have the same local MAC address.

When registers PRMACRU0 and PRMACRL0 and registers PRMACRU1 and PRMACRL1 have the same local MAC address setting, the EPTPC can selects whether PTP packets with the matching address are transmitted through both ports or are only transmitted through one port or the other.

If Ethernet port 0 and Ethernet port 1 have different local MAC address settings, set the TDIS[1:0] bits to 00b. If the setting is other than 00b, the transmission of frames with an source MAC address matching that of the Ethernet port may be blocked.

Rewrite the TRNDISR register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.34 Relay Mode Register (TRNMR)

Address(es): EPTPC.TRNMR E820 5430h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	FWD1	FWD0	—	—	—	—	—	—	—	MOD
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MOD	Cut-Through Mode	0: Store-and-forward 1: Cut-through	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b8	FWD0	Channel 0 Relay Enable	0: Unicast, multicast (other than PTP packets), and broadcast messages from the other node are not relayed from port 0 to port 1. 1: Unicast, multicast (other than PTP packets), and broadcast messages from the other node are relayed from port 0 to port 1.	R/W
b9	FWD1	Channel 1 Relay Enable	0: Unicast, multicast (other than PTP packets), and broadcast messages from the other node are not relayed from port 1 to port 0. 1: Unicast, multicast (other than PTP packets), and broadcast messages from the other node are relayed from port 1 to port 0.	R/W
b31 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R

The TRNMR register is used to control relaying of packets between Ethernet port 0 and Ethernet port 1 by the PRC-TC module. For example, as Figure 28.3 shows, if the network is in a daisy-chain configuration, packets can be transferred through all three devices by enabling relaying of packets between Ethernet port 0 and Ethernet port 1 in the center of the figure.

For recalculating the CRC value in packet transfer, in the case of cut-through mode, a packet that had an abnormal CRC value may be transferred with a normal CRC value. Use store-and-forward mode when abnormal packets need to be discarded. There is no difference in the latency of transfer in cut-through mode and store-and-forward mode when the length of the packets is shorter than 96 bytes.

Rewrite the TRNMR register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

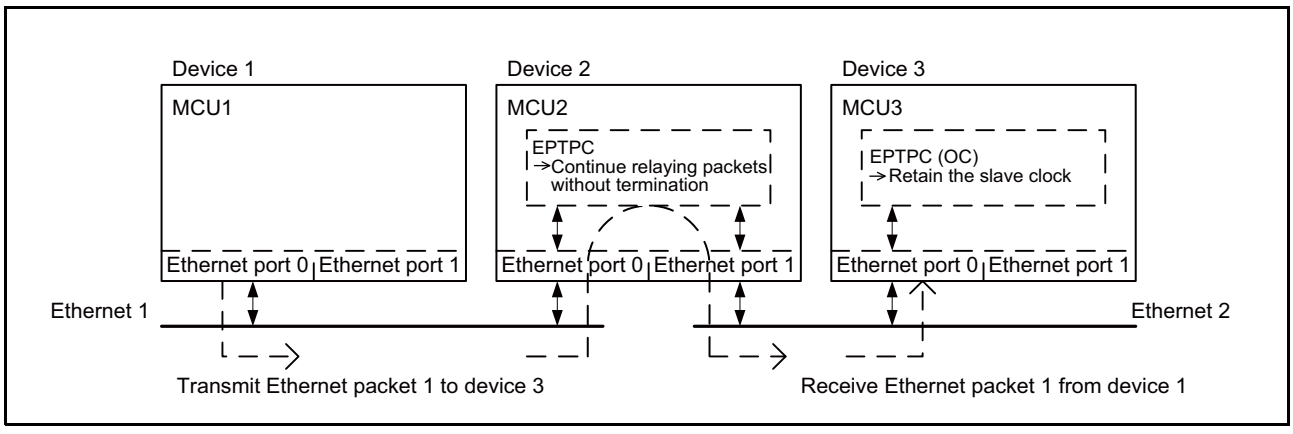


Figure 28.3 Network Configuration Example

28.2.35 Cut-Through Transfer Start Threshold Register (TRNCTTDR)

Address(es): EPTPC.TRNCTTDR E820 5434h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	THVAL[10:0]										
Value after reset:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	THVAL [10:0]	FIFO Read Start Threshold	Threshold for starting to read data from the relay FIFO in cut-through mode (specified as the number of bytes)*1	R
b10 to b2				R/W
b31 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	

Note 1. A value cannot be set in the lower-order 2 bits. These bits are fixed to 0.
A value of less than 96 bytes cannot be set.

The TRNCTTDR register is used to set the threshold for the transmitting port starting to read data from the relay FIFO when the cut-through method is selected for relaying between the Ethernet ports.
When the TRNMR.MOD bit is 1, data transfer can start without waiting for all frame data to be stored in the relay FIFO. The setting of the THVAL[10:0] bits is the amount of data, in bytes, that must be stored in the relay FIFO before transmission starts. Set a multiple of four as the threshold value.
Reading from the relay FIFO starts when either of the following conditions is met.

- The relay FIFO holds at least as much data as the number of bytes specified by the THVAL[10:0] bits.
- The relay FIFO holds at least one frame.

Rewrite the TRNCTTDR register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.36 SYNFP Status Register (SYSR)

Address(es): EPTPC0.SYSR E820 5800h, EPTPC1.SYSR E820 5C00h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GENDN	RESDN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	INFABT	—	RECLP	—	—	—	—	—	DRQOVR	INTDEV	DRPTO	—	MPDUD	INTCHG	OFMUD
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OFMUD	offsetFromMaster Value Update Flag	0: The offsetFromMaster value has not been updated. 1: The offsetFromMaster value has been updated.	R/W*1
b1	INTCHG	Receive logMessageInterval Value Change Detection Flag	0: No change in the received logMessageInterval value. 1: A change in the received logMessageInterval value.	R/W*1
b2	MPDUD	meanPathDelay Value Update Flag	0: The meanPathDelay value has not been updated. 1: The meanPathDelay value has been updated.	R/W*1
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R
b4	DRPTO	Delay_Resp/ Pdelay_Resp Reception Timeout Detection Flag	0: A Delay_Resp/Pdelay_Resp timeout has not occurred. 1: A Delay_Resp/Pdelay_Resp timeout has occurred.	R/W*1
b5	INTDEV	Receive logMessageInterval Value Out-of-Range Flag	0: The received logMessageInterval value is within the range. 1: The received logMessageInterval value is out of the range.	R/W*1
b6	DRQOVR	Delay_Req Reception FIFO Overflow Detection Flag	0: The received Delay_Req has not caused the reception FIFO to overflow. 1: The received Delay_Req has caused the reception FIFO to overflow.	R/W*1
b11 to b7	—	Reserved	These bits are read as 0. The write value should be 0.	R
b12	RECLP	Loop Reception Detection Flag	0: A received message has not returned through a loop. 1: A received message has returned through a loop.	R/W*1
b13	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b14	INFABT	Control Information Abnormality Detection Flag	0: No abnormality in control information 1: Abnormality in control information	R/W*1
b15	—	Reserved	This bit is read as 0. The write value should be 0.	R
b16	RESDN	Response Stop Completion Detection Flag	0: Stopping responses has not been completed. 1: Stopping responses has been completed.	R/W*1
b17	GENDN	Generation Stop Completion Detection Flag	0: Stopping generation has not been completed. 1: Stopping generation has been completed.	R/W*1
b23 to b18	—	Reserved	These bits are read as undefined. The write value should be 0.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note 1. Writing 1 clears the flag. Writing 0 does not affect the flag value.

The SYSR register indicates the state of the SYNFP module.

OFMUD Flag (offsetFromMaster Value Update Flag)

This flag indicates that the value of offsetFromMaster has been updated.

INTCHG Flag (Receive logMessageInterval Value Change Detection Flag)

This flag indicates that the logMessageInterval value of the Delay_Resp, Sync or Announce message differs from the previously received value.

MPDUD Flag (meanPathDelay Value Update Flag)

This flag indicates that the value of meanPathDelay has been updated.

INTDEV Flag (Receive logMessageInterval Value Out-of-Range Flag)

This flag indicates the reception of a Delay_Resp message with a logMessageInterval value outside the range -7 to 6.

DRQOVR Flag (Delay_Req Reception FIFO Overflow Detection Flag)

This flag indicates that the FIFO buffer for storing information from received Delay_Req messages holds 32 or more entries.

RECLP Flag (Loop Reception Detection Flag)

This flag indicates that the value of the sourcePortIdentity field in a received PTP message matches the local PortIdentity (as set by registers SYCIDRU, SYCIDRL, and SYPNUMR).

INFABT Flag (Control Information Abnormality Detection Flag)

This flag indicates that the control information included a mismatch.
When this bit is set to 1, reset the EPTPC, PTPEDMAC, and respective channel of the ETHERC and EDMAC. Follow the procedure below to reset these functional blocks.

(1) Write “0000 0001h” to the PTRSTR register of the EPTPC.
(2) Write “0000 0001h” to the EDMR register of the PTPEDMAC.
(3) Write “0000 0001h” to the EDMR register of the respective channel of the EDMAC.
(4) Wait until the initialization is completed (for 64 cycles of Bφ).
(5) Write “0000 0000h” to the PTRSTR register of the EPTPC.

For details on resetting these functional blocks, see section 28.2.83, PTP Reset Register (PTRSTR) and section 29.2.1, EDMAC Mode Register (EDMR).

RESDN Flag (Response Stop Completion Detection Flag)

This flag indicates the end of processing for transmission of a Delay_Resp or Pdelay_Resp as response messages when the handling of a received Delay_Req or Pdelay_Req by the SYNFP module has been disabled by the setting of the SYRFL1R or SYRVLDR register.

GENDN Flag (Generation Stop Completion Detection Flag)

This flag indicates the end of processing for transmission of messages of a type disabled in the SYTRENR or SYRVLDR register.

28.2.37 SYNFP Status Notification Enable Register (SYIPR)

Address(es): EPTPC0.SYIPR E820 5804h, EPTPC1.SYIPR E820 5C04h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GENDN	RESDN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	INFABT	—	RECLP	—	—	—	—	—	DRQOVR	INTDEV	DRPTO	—	MPDUD	INTCHG	OFMUD
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OFMUD	SYSR.OFMUD Status Notification Enable	0: Disables notification of the state of SYSR.OFMUD. 1: Enables notification of the state of SYSR.OFMUD.	R/W
b1	INTCHG	SYSR.INTCHG Status Notification Enable	0: Disables notification of the state of SYSR.INTCHG. 1: Enables notification of the state of SYSR.INTCHG.	R/W
b2	MPDUD	SYSR.MPDUD Status Notification Enable	0: Disables notification of the state of SYSR.MPDUD. 1: Enables notification of the state of SYSR.MPDUD.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R
b4	DRPTO	SYSR.DRPTO Status Notification Enable	0: Disables notification of the state of SYSR.DRPTO. 1: Enables notification of the state of SYSR.DRPTO.	R/W
b5	INTDEV	SYSR.INTDEV Status Notification Enable	0: Disables notification of the state of SYSR.INTDEV. 1: Enables notification of the state of SYSR.INTDEV.	R/W
b6	DRQOVR	SYSR.DRQOVR Status Notification Enable	0: Disables notification of the state of SYSR.DRQOVR. 1: Enables notification of the state of SYSR.DRQOVR	R/W
b11 to b7	—	Reserved	These bits are read as 0. The write value should be 0.	R
b12	RECLP	SYSR.RECLP Status Notification Enable	0: Disables notification of the state of SYSR.RECLP. 1: Enables notification of the state of SYSR.RECLP.	R/W
b13	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b14	INFABT	SYSR.INFABT Status Notification Enable	0: Disables notification of the state of SYSR.INFABT. 1: Enables notification of the state of SYSR.INFABT.	R/W
b15	—	Reserved	This bit is read as 0. The write value should be 0.	R
b16	RESDN	SYSR.RESDN Status Notification Enable	0: Disables notification of the state of SYSR.RESDN. 1: Enables notification of the state of SYSR.RESDN.	R/W
b17	GENDN	SYSR.GENDN Status Notification Enable	0: Disables notification of the state of SYSR.GENDN. 1: Enables notification of the state of SYSR.GENDN.	R/W
b23 to b18	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

The SYIPR register specifies whether the MIESR.SYn flag does or does not reflect changes in the state of the SYNFPn module.

28.2.38 SYNFP MAC Address Registers (SYMACRU, SYMACRL)

Address(es): EPTPC0.SYMACRU E820 5810h, EPTPC1.SYMACRU E820 5C10h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the higher-order 24 bits of the local MAC address.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Address(es): EPTPC0.SYMACRL E820 5814h, EPTPC1.SYMACRL E820 5C14h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the lower-order 24 bits of the local MAC address.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Registers SYMACRU and SYMACRL are used to specify the local MAC address for Ethernet ports 0 and 1.
Rewrite registers SYMACRU and SYMACRL before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.39 SYNFP LLC-CTL Value Register (SYLLCCTLR)

Address(es): EPTPC0.SYLLCCTLR E820 5818h, EPTPC1.SYLLCCTLR E820 5C18h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CTL[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	CTL[7:0]	LLC-CTL Field	Set a value used for the control field in the LLC sublayer when generating IEEE802.3 frames.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R

The SYLLCCTLR register is used to set the control field (LLC-CTL) value of LCC frames generated by the SYNFP module.
Rewrite the SYLLCCTLR register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.40 SYNFP Local IP Address Register (SYIPADDRR)

Address(es): EPTPC0.SYIPADDRR E820 581Ch, EPTPC1.SYIPADDRR E820 5C1Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the local IP address.	R/W

The SYIPADDRR register is used to specify the local IP address for Ethernet ports 0 and 1.
Rewrite the SYIPADDRR register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.41 SYNFP Specification Version Setting Register (SYSPVRR)

Address(es): EPTPC0.SYSPVRR E820 5840h, EPTPC1.SYSPVRR E820 5C40h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	TRSP[3:0]				VER[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	VER[3:0]	versionPTP Field Value	These bits are used to set the versionPTP field value of the PTP v2 header. When a message is received, this value is compared with the versionPTP field of the received frame. In generating messages, the value is used for the versionPTP field of the frame for transmission. Set these bits to 0010b (PTP v2).	R/W
b7 to b4	TRSP[3:0]	transportSpecific Field Value	These bits are used to set the transportSpecific field value of the PTP v2 header. When a message is received, this value is compared with the transportSpecific field of the received frame. In generating messages, the value is used for the transportSpecific field of the frame for transmission. Set these bits to 0000b (IEEE 1588).	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R

The SYSPVRR register is used to set the transportSpecific and versionPTP field values of the PTP v2 message header. The value should not be modified while reception or transmission of PTP messages is enabled.

28.2.42 SYNFP Domain Number Setting Register (SYDOMR)

Address(es): EPTPC0.SYDOMR E820 5844h, EPTPC1.SYDOMR E820 5C44h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	DNUM[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	DNUM[7:0]	domainNumber Field Value Setting	These bits are used to set the domainNumber field value of the PTP v2 header. When a message is received, this value is compared with the domainNumber field of the received frame as a condition for PTP reception processing. In generating messages, the value is used for the domainNumber field of the frame for transmission.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R

The SYDOMR register is used to set the domainNumber field value of the PTP v2 message header. The value should not be modified while reception or transmission of PTP messages is enabled.

28.2.43 Announce Message Flag Field Setting Register (ANFR)

Address(es): EPTPC0.ANFR E820 5850h, EPTPC1.ANFR E820 5C50h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	FLAG14	FLAG13	—	—	FLAG10	—	FLAG8	—	—	FLAG5	FLAG4	FLAG3	FLAG2	FLAG1	FLAG0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	FLAG0	leap61	This bit is used to set the logical value of the leap61 member of timePropertiesDS. 0: leap61 is set to FALSE. 1: leap61 is set to TRUE.	R/W
b1	FLAG1	leap59	This bit is used to set the logical value of the leap59 member of timePropertiesDS. 0: leap59 is set to FALSE. 1: leap59 is set to TRUE.	R/W
b2	FLAG2	currentUtcOffsetValid	This bit is used to set the logical value of the currentUtcOffsetValid member of timePropertiesDS. 0: currentUtcOffsetValid is set to FALSE. 1: currentUtcOffsetValid is set to TRUE.	R/W
b3	FLAG3	ptpTimescale	This bit is used to set the logical value of the ptpTimescale member of timePropertiesDS. 0: ptpTimescale is set to FALSE. 1: ptpTimescale is set to TRUE.	R/W
b4	FLAG4	timeTraceable	This bit is used to set the logical value of the timeTraceable member of timePropertiesDS. 0: timeTraceable is set to FALSE. 1: timeTraceable is set to TRUE.	R/W
b5	FLAG5	frequencyTraceable	This bit is used to set the logical value of the frequencyTraceable member of timePropertiesDS. 0: frequencyTraceable is set to FALSE. 1: frequencyTraceable is set to TRUE.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	FLAG8	alternateMasterFlag	0: alternateMasterFlag is set to FALSE. 1: alternateMasterFlag is set to TRUE.	R/W
b9	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b10	FLAG10	unicastFlag	0: unicastFlag is set to FALSE. 1: unicastFlag is set to TRUE.	R/W
b12, b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b13	FLAG13	PTP profile Specific 1	0: PTP profile Specific 1 is set to FALSE. 1: PTP profile Specific 1 is set to TRUE.	R/W
b14	FLAG14	PTP profile Specific 2	0: PTP profile Specific 2 is set to FALSE. 1: PTP profile Specific 2 is set to TRUE.	R/W
b31 to b15	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The ANFR register is used to set the flagField section of the header when the SYNFP module is to generate an Announce message.

The value specified by the ANFR register is only reflected in the SYNFP module after the SYRVLDR.ANUP bit is set to 1.

28.2.44 Sync Message Flag Field Setting Register (SYNFR)

Address(es): EPTPC0.SYNFR E820 5854h, EPTPC1.SYNFR E820 5C54h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	FLAG 14	FLAG 13	—	—	FLAG 10	FLAG9	FLAG8	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	FLAG8	alternateMasterFlag	0: alternateMasterFlag is set to FALSE. 1: alternateMasterFlag is set to TRUE.	R/W
b9	FLAG9	twoStepFlag	Set this bit to 0 (FALSE).	R/W
b10	FLAG10	unicastFlag	0: unicastFlag is set to FALSE. 1: unicastFlag is set to TRUE.	R/W
b12, b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b13	FLAG13	PTP profile Specific 1	0: PTP profile Specific 1 is set to FALSE. 1: PTP profile Specific 1 is set to TRUE.	R/W
b14	FLAG14	PTP profile Specific 2	0: PTP profile Specific 2 is set to FALSE. 1: PTP profile Specific 2 is set to TRUE.	R/W
b31 to b15	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The SYNFR register is used to set the flagField section of the header when the SYNFP module is to generate a Sync message.
The value specified by the SYNFR register is only reflected in the SYNFP module after the SYRVLDR.STUP bit is set to 1.

28.2.45 Delay_Req Message Flag Field Setting Register (DYRQFR)

Address(es): EPTPC0.DYRQFR E820 5858h, EPTPC1.DYRQFR E820 5C58h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	FLAG14	FLAG13	—	—	FLAG10	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b9 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b10	FLAG10	unicastFlag	0: unicastFlag is set to FALSE. 1: unicastFlag is set to TRULE.	R/W
b12, b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b13	FLAG13	PTP profile Specific 1	0: PTP profile Specific 1 is set to FALSE. 1: PTP profile Specific 1 is set to TRULE.	R/W
b14	FLAG14	PTP profile Specific 2	0: PTP profile Specific 2 is set to FALSE. 1: PTP profile Specific 2 is set to TRULE.	R/W
b31 to b15	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Register DYRQFR is used to set the flagField section of the header when the SYNFP module is to generate a Delay_Req or Pdelay_Req message.
The value specified by DYRQFR is only reflected in the SYNFP module after the SYRVLDR.STUP bit is set to 1.

28.2.46 Delay_Resp Message Flag Field Setting Register (DYRPFR)

Address(es): EPTPC0.DYRPFR E820 585Ch, EPTPC1.DYRPFR E820 5C5Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	FLAG14	FLAG13	—	—	FLAG10	FLAG9	FLAG8	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	FLAG8	alternateMasterFlag*1	0: alternateMasterFlag is set to FALSE. 1: alternateMasterFlag is set to TRUE.	R/W
b9	FLAG9	twoStepFlag*2	Set this bit to 0 (FALSE).	R/W
b10	FLAG10	unicastFlag	0: unicastFlag is set to FALSE. 1: unicastFlag is set to TRUE.	R/W
b12, b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b13	FLAG13	PTP profile Specific 1	0: PTP profile Specific 1 is set to FALSE. 1: PTP profile Specific 1 is set to TRUE.	R/W
b14	FLAG14	PTP profile Specific 2	0: PTP profile Specific 2 is set to FALSE. 1: PTP profile Specific 2 is set to TRUE.	R/W
b31 to b15	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. This bit is reserved for Pdelay_Resp messages. Set the bit to 0.
Note 2. This bit is reserved for Delay_Resp messages.

The DYRPFR register is used to set the flagField section of the header when the SYNFP module is to generate a Delay_Resp or PDelay_Resp message.
The value specified by the DYRPFR register is only reflected in the SYNFP module after the SYRVLDR.STUP bit is set to 1.
Do not change the value of the DYRPFR register while processing for the transmission of Delay_Resp or Pdelay_Resp messages is enabled. Furthermore, after disabling this processing for transmission, do not change the value of the DYRPFR register until the SYSR.RESDN flag becomes 1.

28.2.47 SYNFP Local Clock ID Registers (SYCIDRU, SYCIDRL)

Address(es): EPTPC0.SYCIDRU E820 5860h, EPTPC1.SYCIDRU E820 5C60h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the higher-order 32 bits of the clock-ID of the local port.	R/W

Address(es): EPTPC0.SYCIDRL E820 5864h, EPTPC0.SYCIDRL E820 5C64h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the higher-order 32 bits of the clock-ID of the local port.	R/W

The SYCIDR registers are used to set the clock-ID of the local port.
The registers are used for the clockIdentity section in the sourcePortIdentity field of the header when the SYNFP module is to generate a PTP message.
When a PTP message is received, the value of these registers is compared with the clockIdentity section in the sourcePortIdentity field of the PTP message to determine whether the message is one that was transmitted by you.
Usually, the setting should be the same as the value of portDS.portIdentity.clockIdentity.
Do not change the value of these registers while processing for the reception or transmission of PTP messages is enabled.

28.2.48 SYNFP Local Port Number Register (SYPNUMR)

Address(es): EPTPC0.SYPNUMR E820 5868h, EPTPC1.SYPNUMR E820 5C68h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PNUM[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	PNUM[15:0]	Local Port Number Setting	These bits hold the setting for the port number of the local port.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R

The SYPNUMR register is used to set the port number of the local port.
This register is used for the portNumber section in the sourcePortIdentity field of the header when the SYNFP module is to generate a PTP message.
When a PTP message is received, the value of this register is compared with the portNumber section in the sourcePortIdentity field of the PTP message to determine whether the message is one that was transmitted by the local device.
Usually, the setting should be the same as the value of portDS.portIdentity.portNumber.
Do not change the value of this register while processing for the reception or transmission of PTP messages is enabled.

28.2.49 SYNFP Register Value Load Directive Register (SYRVLDR)

Address(es): EPTPC0.SYRVLDR E820 5880h, EPTPC1.SYRVLDR E820 5C80h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	ANUP	STUP	BMUP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	BMUP	BMC Update	When this bit is set to 1, the SYNFP module simultaneously reflects values of registers storing the MasterClock identifying information.	W
b1	STUP	State Update	When this bit is set to 1, the SYNFP module simultaneously reflects register values for PTP message reception and transmission.	W
b2	ANUP	Announce Message Generation Information Update	When this bit is set to 1, the Announce message generation block simultaneously reflects register values required for generating Announce messages.	W
b31 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R

The SYRVLDR register is used for simultaneously updating multiple register values in the SYNFP module.

BMUP Bit (BMC Update)

When the BMUP bit is set to 1, the SYNFP module simultaneously reflects the values of registers storing the MasterClock identifying information listed below.

- Registers MTCIDU and MTCIDL
- MTPID register

STUP Bit (State Update)

When the STUP bit is set to 1, the SYNFP module simultaneously reflects the values of registers and bits for PTP message reception and transmission listed below.

- SYNFR register
- DYRQFR register
- SYTLIR.DREQ[7:0] bits
- RSTOUTR register
- SYRFL1R register
- SYRFL2R register
- SYTRENR register

ANUP Bit (Announce Message Generation Information Update)

When the ANUP bit is set to 1, the Announce message generation block simultaneously reflects the values of registers and bits required for generating Announce messages listed below.

- ANFR register
- SYTLIR.ANCE[7:0] bits
- GMPR register
- GMCQR register
- Registers GMIDRU and GMIDRL
- CUOTSR register
- SRR register

28.2.50 SYNFP Reception Filter Register 1 (SYRFL1R)

Address(es): EPTPC0.SYRFL1R E820 5890h, EPTPC1.SYRFL1R E820 5C90h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	PDFUP[2:0]			—	PDRP[2:0]			—	PDRQ[2:0]			—	DRP[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	DRQ[2:0]			—	FUP[2:0]			—	SYNC[2:0]			—	—	ANCE[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ANCE[1:0]	Announce Message Processing	0: Messages are not transferred to the PTPEDMAC. 1: Messages are transferred to the PTPEDMAC.	R/W
b1			0: The PRC-TC does not relay messages between ports 0 and 1. 1: The PRC-TC relays messages between ports 0 and 1.	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	SYNC[2:0]	Sync Message Processing	0: Messages are not transferred to the PTPEDMAC. 1: Messages are transferred to the PTPEDMAC.	R/W
b5			0: The PRC-TC does not relay messages between ports 0 and 1. 1: The PRC-TC relays messages between ports 0 and 1.	R/W
b6			0: The SYNFP does not process messages. 1: The SYNFP processes messages.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R
b8	FUP[2:0]	Follow_Up Message Processing	0: Messages are not transferred to the PTPEDMAC. 1: Messages are transferred to the PTPEDMAC.	R/W
b9			0: The PRC-TC does not relay messages between ports 0 and 1. 1: The PRC-TC relays messages between ports 0 and 1.	R/W
b10			0: The SYNFP does not process messages. 1: The SYNFP processes messages.	R/W
b11	—	Reserved	This bit is read as 0. The write value should be 0.	R
b12	DRQ[2:0]	Delay_Req Message Processing	0: Messages are not transferred to the PTPEDMAC. 1: Messages are transferred to the PTPEDMAC.	R/W
b13			0: The PRC-TC does not relay messages between ports 0 and 1. 1: The PRC-TC relays messages between ports 0 and 1.	R/W
b14			0: The SYNFP does not process messages. 1: The SYNFP processes messages.	R/W
b15	—	Reserved	This bit is read as 0. The write value should be 0.	R
b16	DRP[2:0]	Delay_Resp Message Processing	0: Messages are not transferred to the PTPEDMAC. 1: Messages are transferred to the PTPEDMAC.	R/W
b17			0: The PRC-TC does not relay messages between ports 0 and 1. 1: The PRC-TC relays messages between ports 0 and 1.	R/W
b18			0: The SYNFP does not process messages. 1: The SYNFP processes messages.	R/W
b19	—	Reserved	This bit is read as 0. The write value should be 0.	R
b20	PDRQ[2:0]	Pdelay_Req Message Processing	0: Messages are not transferred to the PTPEDMAC. 1: Messages are transferred to the PTPEDMAC.	R/W
b21			0: The PRC-TC does not relay messages between ports 0 and 1. 1: The PRC-TC relays messages between ports 0 and 1.	R/W
b22			0: The SYNFP does not process messages. 1: The SYNFP processes messages.	R/W
b23	—	Reserved	This bit is read as 0. The write value should be 0.	R

Bit	Symbol	Bit Name	Description	R/W
b24	PDRP[2:0]	Pdelay_Resp Message Processing	0: Messages are not transferred to the PTPEDMAC. 1: Messages are transferred to the PTPEDMAC.	R/W
b25			0: The PRC-TC does not relay messages between ports 0 and 1. 1: The PRC-TC relays messages between ports 0 and 1.	R/W
b26			0: The SYNFP does not process messages. 1: The SYNFP processes messages.	R/W
b27	—	Reserved	This bit is read as 0. The write value should be 0.	R
b28	PDFUP [2:0]	Pdelay_Resp_Follow_Up Message Processing	0: Messages are not transferred to the PTPEDMAC. 1: Messages are transferred to the PTPEDMAC.	R/W
b29			0: The PRC-TC does not relay messages between ports 0 and 1. 1: The PRC-TC relays messages between ports 0 and 1.	R/W
b30			0: The SYNFP does not process messages. 1: The SYNFP processes messages.	R/W
b31	—	Reserved	This bit is read as 0. The write value should be 0.	R

The SYRFL1R register is used to set up filtering for the reception of PTP messages.
Multiple bits corresponding to different types of messages can be set to 1. Setting all bits for a type of message to 0 leads to all messages of the given type being discarded.
The value specified by the SYRFL1R register is only reflected in the SYNFP module after the SYRVLDR.STUP bit is set to 1.

28.2.51 SYNFP Reception Filter Register 2 (SYRFL2R)

Address(es): EPTPC0.SYRFL2R E820 5894h, EPTPC0.SYRFL2R E820 5C94h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	ILL[1:0]		—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	SIG[1:0]		—	—	MAN[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MAN[1:0]	Management Message Processing Setting	0: Messages are not transferred to the PTPEDMAC. 1: Messages are transferred to the PTPEDMAC.	R/W
b1			0: The PRC-TC does not relay messages between ports 0 and 1. 1: The PRC-TC relays messages between ports 0 and 1.	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	SIG[1:0]	Signaling Message Processing Setting	0: Messages are not transferred to the PTPEDMAC. 1: Messages are transferred to the PTPEDMAC.	R/W
b5			0: The PRC-TC does not relay messages between ports 0 and 1. 1: The PRC-TC relays messages between ports 0 and 1.	R/W
b27 to b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b28	ILL[1:0]	Illegal Message Processing Setting*1	0: Messages are not transferred to the PTPEDMAC. 1: Messages are transferred to the PTPEDMAC.	R/W
b29			0: The PRC-TC does not relay messages between ports 0 and 1. 1: The PRC-TC relays messages between ports 0 and 1.	R/W
b31, b30	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note 1. PTP messages other than PTP v2 messages and messages of undefined type are handled as illegal messages.

The SYRFL2R register is used to set up filtering for the reception of PTP messages.
Multiple bits corresponding to different types of messages can be set to 1. Setting all bits for a type of message to 0 leads to all messages of the given type being discarded.
The value specified by the SYRFL2R register is only reflected in the SYNFP module after the SYRVLDR.STUP bit is set to 1.

28.2.52 SYNFP Transmission Enable Register (SYTRENR)

Address(es): EPTPC0.SYTRENR E820 5898h, EPTPC1.SYTRENR E820 5C98h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PDRQ	—	—	—	DRQ	—	—	—	SYNC	—	—	—	ANCE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ANCE	Announce Message Transmission Enable	0: Announce messages are not transmitted. 1: Announce messages are transmitted.	R/W
b3 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	SYNC	Sync Message Transmission Enable	0: Sync messages are not transmitted. 1: Sync messages are transmitted.	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b8	DRQ	Delay_Req Message Transmission Enable	0: Delay_Req messages are not transmitted. 1: Delay_Req messages are transmitted.	R/W
b11 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R
b12	PDRQ	Pdelay_Req Message Transmission Enable	0: Pdelay_Req messages are not transmitted. 1: Pdelay_Req messages are transmitted.	R/W
b31 to b13	—	Reserved	These bits are read as 0. The write value should be 0.	R

The SYTRENR register is used to enable or disable transmission of PTP messages.
The PDRQ and DRQ bits should not be set to 1 at the same time. Operation is not guaranteed when both bits are set to 1.
The value specified by the SYTRENR register is only reflected in the SYNFP module after the SYRVLDR.STUP bit is set to 1.

28.2.53 Master Clock ID Registers (MTCIDU, MTCIDL)

Address(es): EPTPC0.MTCIDU E820 58A0h, EPTPC1.MTCIDU E820 5CA0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the higher-order 32 bits of the clock-ID of the master clock.	R/W

Address(es): EPTPC0.MTCIDL E820 58A4h, EPTPC1.MTCIDL E820 5CA4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the higher-order 32 bits of the clock-ID of the master clock.	R/W

Registers MTCIDU and MTCIDL are used to set the clock-ID of the master for synchronization.
The value specified by registers MTCIDU and MTCIDL is only reflected in the SYNFP module after the SYRVLDR.BMUP bit is set to 1.

28.2.54 Master Clock Port Number Register (MTPID)

Address(es): EPTPC0.MTPID E820 58A8h, EPTPC1.MTPID E820 5CA8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PNUM[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	PNUM [15:0]	Master Clock Port Number Setting	These bits hold the setting for the port number of the master clock.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R

The MTPID register is used to set the port number of the master for synchronization.
The value specified by the MTPID register is only reflected in the SYNFP module after the SYRVLDR.BMUP bit is set to 1.
In normal usage, set the value of parentDS.parentPortIdentity.portNumber in this register.

28.2.55 SYNFP Transmission Interval Setting Register (SYTLIR)

Address(es): EPTPC0.SYTLIR E820 58C0h, EPTPC1.SYTLIR E820 5CC0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	DREQ[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC[7:0]								ANCE[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	ANCE[7:0]	Announce Message Transmission Interval Setting	These bits set the interval for the transmission of Announce messages.	R/W
b15 to b8	SYNC[7:0]	Sync Message Transmission Interval Setting	These bits set the interval for the transmission of Sync messages. The setting is also placed in the logMessageInterval field of transmitted Sync messages.	R/W
b23 to b16	DREQ[7:0]	Delay_Req Transmission Interval Average Value/ Pdelay_Req Transmission Interval Setting	The bits set the average interval for the transmission of Delay_Req messages and the interval for the transmission of Pdelay_Req messages. The setting is also placed in the logMessageInterval field of Delay_Resp messages.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

The SYTLIR register is used to set the interval for the transmission of messages generated by the SYNFP module. The setting is an integer logarithm in base 2 (log2(x)) and determines a value x in seconds.
In other words, the interval for transmission is 2n (s), where n is the setting. The available settings are from -7 (F9h) to +6 (06h).

Example:
If the setting is 06h, then the interval for transmission is $2^6 = 64$ (s).
If the setting is 00h, then the interval for transmission is $2^0 = 1$ (s).
If the setting is FFh, then the interval for transmission is $2^{-1} = 0.5$ (s) = 500 (ms).
If the setting is F9h, then the interval for transmission is $2^{-7} = 0.0078125$ (s) = 7.8125 (ms).

The value set in the ANCE[7:0] bits is only reflected in the SYNFP module after the SYRVLDR.ANUP bit is set to 1.
The value set in the DREQ[7:0] and SYNC[7:0] bits is only reflected in the SYNFP module after the SYRVLDR.STUP bit is set to 1.

28.2.56 SYNFP Received logMessageInterval Value Indication Register (SYRLIR)

Address(es): EPTPC0.SYRLIR E820 58C4h, EPTPC1.SYRLIR E820 5CC4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	DRESP[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC[7:0]								ANCE[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	ANCE[7:0]	Announce Message logMessageInterval Field Indication Flag	These bits indicate the logMessageInterval field value of a received Announce message.	R
b15 to b8	SYNC[7:0]	Sync Message logMessageInterval Field Indication Flag	These bits indicate the logMessageInterval field value of a received Sync message.	R
b23 to b16	DRESP[7:0]	Delay_Resp Message logMessageInterval Field Indication Flag	These bits indicate the logMessageInterval field value of a received Delay_Resp message.	R
b31 to b24	—	Reserved	These bits are read as 0.	R

The SYRLIR register indicates the logMessageInterval field values of received PTP messages.

28.2.57 offsetFromMaster Value Registers (OFMRU, OFMRL)

Address(es): EPTPC0.OFMRU E820 58C8h, EPTPC1.OFMRU E820 5CC8h zzz

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits indicate the higher-order 32 bits of the calculated offsetFromMaster value.	R

Address(es): EPTPC0.OFMRL E820 58CCh, EPTPC1.OFMRL E820 5CCCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits indicate the lower-order 32 bits of the calculated offsetFromMaster value.	R

Registers OFMRU and OFMRL indicate the calculated offsetFromMaster value.
The value is expressed as two's complements in units of nanoseconds. Note that the numeric representation*1 differs from that of the offsetFromMaster member of the current data set (currentDS).
When read, access to the registers should be in order of OFMRU and then OFMRL.

Note 1. The value of currentDS.offsetFromMaster is multiplied by 2^{16}.
Example: 2.5 (ns) = 00000000_00028000h

28.2.58 meanPathDelay Value Registers (MPDRU, MPDRL)

Address(es): EPTPC0.MPDRU E820 58D0h, EPTPC1.MPDRU E820 5CD0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits indicate the higher-order 32 bits of the calculated meanPathDelay value.	R

Address(es): EPTPC0.MPDRL E820 58D4h, EPTPC1.MPDRL E820 5CD4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits indicate the lower-order 32 bits of the calculated meanPathDelay value.	R

Registers MPDRU and MPDRL indicate the calculated meanPathDelay value.
The value is expressed as two's complements in units of nanoseconds. Note that the numeric representation*1 differs from that of the meanPathDelay member of the current data set (currentDS).
When read, access to the registers should be in order of MPDRU and then MPDRL.

Note 1. The value of currentDS.meanPathDelay is multiplied by 2^{16}.
Example: 2.5 (ns) = 00000000_00028000h

28.2.59 grandmasterPriority Field Setting Register (GMPR)

Address(es): EPTPC0.GMPR E820 58E0h, EPTPC1.GMPR E820 5CE0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	GMPR1[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	GMPR2[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	GMPR2 [7:0]	grandmasterPriority2 Field Value Setting	These bits are used to set the value of the grandmasterPriority2 fields of Announce messages.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b23 to b16	GMPR1 [7:0]	grandmasterPriority1 Field Value Setting	These bits are used to set the value of the grandmasterPriority1 fields of Announce messages.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

The GMPR register is used to specify the grandmasterPriority1 and grandmasterPriority2 field values of Announce messages generated by the SYNFP module.
The value specified by the GMPR register is only reflected in the SYNFP module after the SYRVLDR.ANUP bit is set to 1.

28.2.60 grandmasterClockQuality Field Setting Register (GMCQR)

Address(es): EPTPC0.GMCQR E820 58E4h, EPTPC0.GMCQR E820 5CE4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

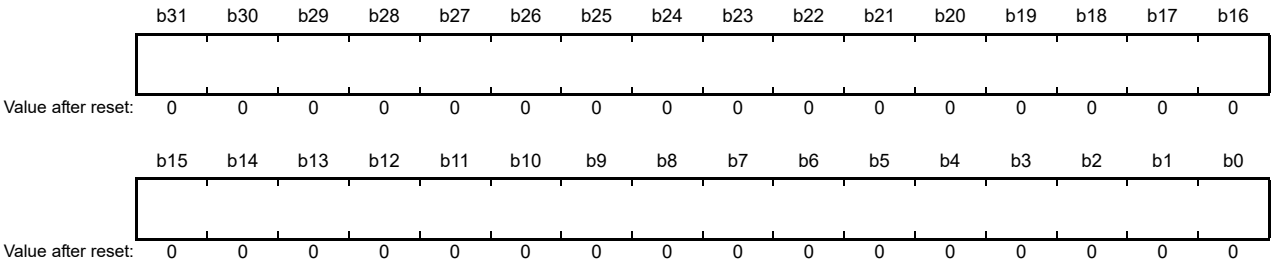
Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits are used to set the value of the grandmasterClockQuality fields of Announce messages. The correspondence between bits and the grandmasterClockQuality fields is as listed below. b31 to b24: clockClass b23 to b16: clockAccuracy b15 to b0: offsetScaledLogVariance	R/W

The GMCQR register is used to specify the grandmasterClockQuality field value of Announce messages generated by the SYNFP module.

The value specified by the GMCQR register is only reflected in the SYNFP module after the SYRVLDR.ANUP bit is set to 1.

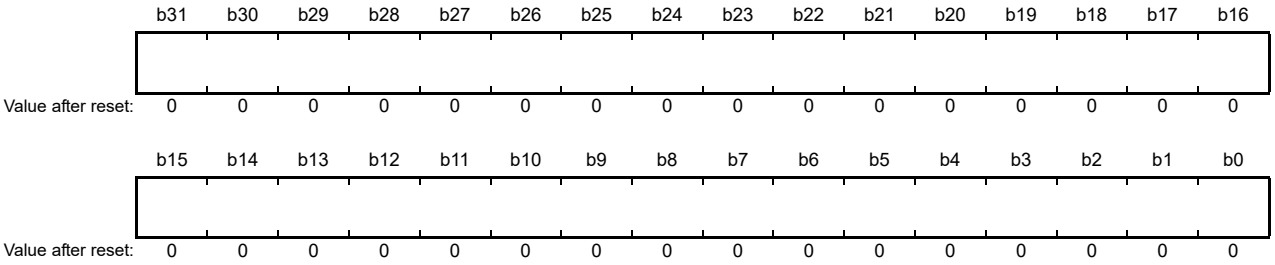
28.2.61 grandmasterIdentity Field Setting Registers (GMIDRU, GMIDRL)

Address(es): EPTPC0.GMIDRU E820 58E8h, EPTPC1.GMIDRU E820 5CE8h



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the higher-order 32 bits of the value of the grandmasterIdentity fields of Announce messages.	R/W

Address(es): EPTPC0.GMIDRL E820 58ECh, EPTPC1.GMIDRL E820 5CECh

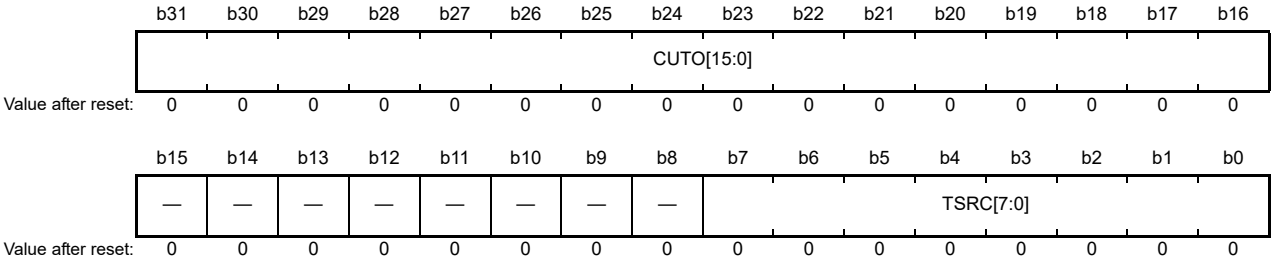


Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the lower-order 32 bits of the value of the grandmasterIdentity fields of Announce messages.	R/W

Registers GMIDRU and GMIDRL are used to specify the grandmasterIdentity field value of Announce messages generated by the SYNFP module.
The value specified by registers GMIDRU and GMIDRL is only reflected in the SYNFP module after the SYRVLDR.ANUP bit is set to 1.

28.2.62 currentUtcOffset/timeSource Field Setting Register (CUOTSR)

Address(es): EPTPC0.CUOTSR E820 58F0h, EPTPC1.CUOTSR E820 5CF0h

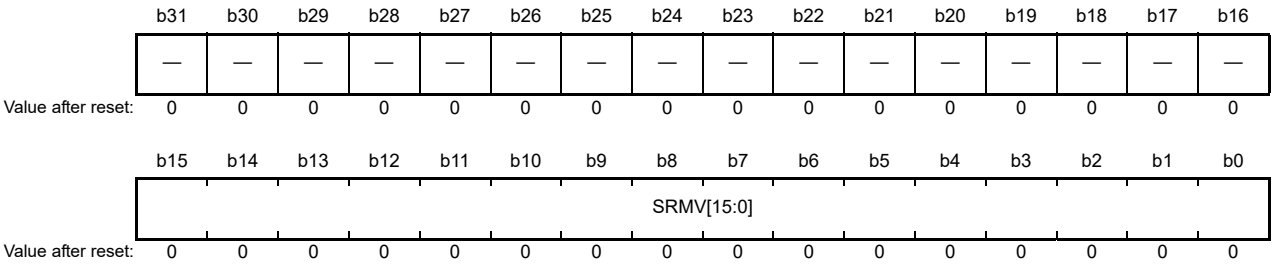


Bit	Symbol	Bit Name	Description	R/W
b7 to b0	TSRC[7:0]	timeSource Field Setting	These bits set the value of the timeSource fields of Announce messages.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b31 to b16	CUTO[15:0]	currentUtcOffset Field Setting	These bits set the value of the currentUtcOffset fields of Announce messages.	R/W

The CUOTSR register is used to specify the currentUtcOffset and timeSource field values of Announce messages generated by the SYNFP module.
The value specified by the CUOTSR register is only reflected in the SYNFP module after the SYRVLDR.ANUP bit is set to 1.

28.2.63 stepsRemoved Field Setting Register (SRR)

Address(es): EPTPC0.SRR E820 58F4h, EPTPC1.SRR E820 5CF4h



Bit	Symbol	Bit Name	Description	R/W
b15 to b0	SRMV [15:0]	stepsRemoved Field Value Setting	These bits set the value of the stepsRemoved fields of Announce messages.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R

The SRR register is used to specify the stepsRemoved field value of Announce messages generated by the SYNFP module.
The value specified by the SRR register is only reflected in the SYNFP module after the SYRVLDR.ANUP bit is set to 1.

28.2.64 PTP-primary Message Destination MAC Address Setting Registers (PPMACRU, PPMACRL)

Address(es): EPTPC0.PPMACRU E820 5900h, EPTPC1.PPMACRU E820 5D00h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	1	1	0	1	1	0	0	0	1	1	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the higher-order 24 bits of the destination MAC address for PTP-primary messages.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

Address(es): EPTPC0.PPMACRL E820 5904h, EPTPC1.PPMACRL E820 5D04h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	1	1	0	1	1	0	0	0	1	1	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the lower-order 24 bits of the destination MAC address for PTP-primary messages.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

The PPMACR registers are used to specify the destination MAC address for PTP-primary messages. In normal usage, set 01:1B:19:00:00:00 in these registers.

The value is used in the destination MAC address field when generating an Ethernet frame for a PTP-primary message. It is also used as a determining condition for received frames carrying PTP messages.

Set a value in these registers before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.65 PTP-pdelay Message MAC Address Setting Registers (PDMACRU, PDMACRL)

Address(es): EPTPC0.PDMACRU E820 5908h, EPTPC1.PDMACRU E820 5D08h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	0	0	0	0	0	0	0	1	1	0	0	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the higher-order 24 bits of the destination MAC address for PTP-pdelay messages.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

Address(es): EPTPC0.PDMACRL E820 590Ch, EPTPC1.PDMACRL E820 5D0Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the lower-order 24 bits of the destination MAC address for PTP-pdelay messages.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

The PDMACR registers are used to specify the destination MAC address for PTP-pdelay messages. In normal usage, set 01:80:C2:00:00:0E in these registers.

The value is used in the destination MAC address field when generating frames carrying PTP-pdelay messages in the Ethernet format. It is also used as a determining condition for received frames carrying PTP messages.

Set a value in these registers before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.66 PTP Message Ethertype Setting Register (PETYPER)

Address(es): EPTPC0.PETYPER E820 5910h, EPTPC1.PETYPER E820 5D10h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TYPE[15:0]															
Value after reset:	1	0	0	0	1	0	0	0	1	1	1	1	0	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	TYPE [15:0]	PTP Message Ethertype Value Setting	These bits hold the setting for the Ethertype field value for frames in the Ethernet II format.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R

The PETYPER register is used to specify the Ethertype field for frames carrying the PTP messages. In normal usage, set 0000 88F7h in this register.
The value is used in the Ethertype field when generating frames carrying PTP messages in the Ethernet II format. It is also used as a determining condition for received frames carrying PTP messages.
Set a value in these registers before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.67 PTP-primary Message Destination IP Address Setting Register (PPIPR)

Address(es): EPTPC0.PPIPR E820 5920h, EPTPC1.PPIPR E820 5D20h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the destination IP address for PTP-primary messages.	R/W

The PPIPR register is used to specify the destination IP address for PTP messages. In normal usage, set E000 0181h (224.0.1.129) in this register.

The value is used in the destination IP address field when generating frames carrying PTP-primary messages in the IPv4 format. The lower-order 23 bits are also used in the destination MAC address field for Ethernet frames. The value is also used as a determining condition for received frames carrying PTP messages.

Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.68 PTP-pdelay Message Destination IP Address Setting Register (PDIPR)

Address(es): EPTPC0.PDIPR E820 5924h, EPTPC1.PDIPR E820 5D24h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	1	1	0	1	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the destination IP address for PTP-pdelay messages.	R/W

The PDIPR register is used to specify the destination IP address for PTP-pdelay messages. In normal usage, set E000 006Bh (224.0.0.107) in this register.

The value is used in the destination IP address field when generating frames carrying PTP-pdelay messages in the IPv4 format. The lower-order 23 bits are also used in the destination MAC address field for Ethernet frames. The value is also used as a determining condition for received frames carrying PTP messages.

Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.69 PTP Event Message TOS Setting Register (PETOSR)

Address(es): EPTPC0.PETOSR E820 5928h, EPTPC1.PETOSR E820 5D28h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	EVTO[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	EVTO[7:0]	PTP Event Message TOS Field Value Setting	These bits hold the setting for the value of the TOS field within the IPv4 headers of PTP event messages.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R

The PETOSR register is used to set the TOS (type of service) field value within the IPv4 headers of PTP event messages. Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.70 PTP general Message TOS Setting Register (PGTOSR)

Address(es): EPTPC0.PGTOSR E820 592Ch, EPTPC1.PGTOSR E820 5D2Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	GETO[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	GETO[7:0]	PTP general Message TOS Field Value Setting	These bits hold the setting for the value of the TOS field within the IPv4 headers of PTP general messages.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R

The PGTOSR register is used to set the TOS (type of service) field value within the IPv4 headers of PTP general messages.
Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.71 PTP-primary Message TTL Setting Register (PPTTLR)

Address(es): EPTPC0.PPTTLR E820 5930h, EPTPC1.PPTTLR E820 5D30h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	PRTL[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	PRTL[7:0]	PTP-primary Message TTL Field Value Setting	These bits hold the setting for the value of the TTL field within the IPv4 headers of PTP-primary messages.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R

The PPTTLR register is used to set the TTL (time to live) field value within the IPv4 headers of PTP-primary messages. Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.72 PTP-pdelay Message TTL Setting Register (PDTTLR)

Address(es): EPTPC0.PDTTLR E820 5934h, EPTPC1.PDTTLR E820 5D34h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	PDTL[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	PDTL[7:0]	PTP-pdelay Message TTL Field Value	These bits hold the setting for the value of the TTL field within the IPv4 headers of PTP-pdelay messages.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R

The PDTTLR register is used to set the TTL field value within the IPv4 headers of PTP-pdelay messages.
Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.73 PTP Event Message UDP Destination Port Number Setting Register (PEUDPR)

Address(es): EPTPC0.PEUDPR E820 5938h, EPTPC1.PEUDPR E820 5D38h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	1	0	0	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	EVUPT [15:0]	PTP Event Message Destination Port Number Setting	These bits hold the setting for the value of the destination port number field within the UDP headers of PTP event messages.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R

The PEUDPR register is used to set the destination port number field value within the UDP headers of PTP event messages. In normal usage, set 013Fh (319) in this register.
Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.74 PTP general Message UDP Destination Port Number Setting Register (PGUDPR)

Address(es): EPTPC0.PGUDPR E820 593Ch, EPTPC1.PGUDPR E820 5D3Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	GEUPT[15:0]															
Value after reset:	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	GEUPT [15:0]	PTP general Message Destination Port Number	These bits hold the setting for the value of the destination port number field within the UDP headers of PTP general messages.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R

The PGUDPR register is used to set the destination port number field value within the UDP headers of PTP general messages. In normal usage, set 0140h (320) in this register.
Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.75 Frame Reception Filter Setting Register (FFLTR)

Address(es): EPTPC0.FFLTR E820 5940h, EPTPC1.FFLTR E820 5D40h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EXTPRM
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	ENB	PRT	SEL
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SEL	Receive MAC Address Select*1	This bit selects how filtering is handled when multicast frames other than PTP messages are received. b2 b0 0 0 0: Filtering is disabled (all multicast frames are received). 0 0 1: Filtering is disabled (all multicast frames are received). 0 1 0: Filtering is disabled (all multicast frames are received). 0 1 1: Filtering is disabled (all multicast frames are received). 1 0 0: Do not receive multicast frames. 1 0 1: Do not receive multicast frames. 1 1 0: Only receive multicast frames matching the MAC address setting in registers FMAC0RU and FMAC0RL. 1 1 1: Only receive multicast frames matching the MAC address setting in registers FMAC1RU and FMAC1RL).	R/W
b1	PRT	Frame Reception Enable*1		R/W
b2	ENB	Reception Filter Enable*1		R/W
b15 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b16	EXTPRM	Extended Promiscuous Mode Setting	0: Normal operation (unicast frames addressed to the EPTPC are received, filtering of PTP frames is applied, multicast filtering is applied, and all broadcast frames are received). 1: Extended promiscuous mode (all frames are received)	R/W
b31 to b17	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note 1. The setting of these bits is only effective when the EXTPRM bit is 0.

The FFLTR register is used to switch extended promiscuous mode on or off and to select how filtering is handled when multicast frames other than PTP messages are received.
The filter for the reception of multicast frames other than PTP messages is enabled by setting the ENB, PRT, and SEL bits to 110b or 111b. Frames passed by the filter are then transferred by EDMACn.
Relaying of multicast frames other than PTP messages is in accord with the TRNMR register and relaying and reception of PTP messages is in accord with the setting of registers SYRFL1R and SYRFL2R.
Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.76 Frame Reception Filter MAC Address 0 Setting Registers (FMAC0RU, FMAC0RL)

Address(es): EPTPC0.FMAC0RU E820 5960h, EPTPC1.FMAC0RU E820 5D60h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the higher-order 24 bits of the destination MAC address for received multicast frames.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

Address(es): EPTPC0.FMAC0RL E820 5964h, EPTPC1.FMAC0RL E820 5D64h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the lower-order 24 bits of the destination MAC address for received multicast frames.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

Registers FMAC0RU and FMAC0RL are used to specify the MAC address for filtering in the reception of multicast frames other than PTP messages.

Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.77 Frame Reception Filter MAC Address 1 Setting Registers (FMAC1RU, FMAC1RL)

Address(es): EPTPC0.FMAC1RU E820 5968h, EPTPC1.FMAC1RU E820 5D68h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the higher-order 24 bits of the destination MAC address for received multicast frames.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

Address(es): EPTPC0.FMAC1RL E820 596Ch, EPTPC1.FMAC1RL E820 5D6Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	—	—	These bits hold the setting for the lower-order 24 bits of the destination MAC address for received multicast frames.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R

Registers FMAC1RU and FMAC1RL are used to specify the MAC address for filtering in the reception of multicast frames other than PTP messages.

Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.78 Asymmetric Delay Setting Registers (DASYMRU, DASYMRL)

Address(es): EPTPC0.DASYMRU E820 59C0h, EPTPC1.DASYMRU E820 5DC0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	—	These bits hold the setting for the higher-order 16 bits of the asymmetric delay value. Set them to 0000h in this MCU.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R

Address(es): EPTPC0.DASYMRL E820 59C4h, EPTPC1.DASYMRL E820 5DC4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	These bits hold the setting for the lower-order 32 bits of the asymmetric delay value. Set them to 0000 0000h in this MCU.	R/W

Registers DASYMRU and DASYMRL are used to set the asymmetric delay value (delayAsymmetry).

Set the registers DASYMRU and DASYMRL to 0000 0000h in this MCU.

28.2.79 Timestamp Latency Setting Register (TSLATR)

Address(es): EPTPC0.TSLATR E820 59C8h, EPTPC1.TSLATR E820 5DC8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	INGP[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	EGP[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	EGP [15:0]	Output Port Timestamp Latency Setting	These bits hold the setting for the time stamp latency (ns) for the output ports.	R/W
b31 to b16	INGP [15:0]	Input Port Timestamp Latency Setting	These bits hold the setting for the time stamp latency (ns) for the input ports.	R/W

The TSLATR register is used to set the amount of latency in timestamp acquisition in nanoseconds.
The value should not be modified while reception or transmission of PTP messages is enabled.

EGP[15:0] Bits (Output Port Timestamp Latency Setting)

Set the corresponding fixed value in Table 28.9. The timestamp latency differs with the link transfer rate (100 Mbps or 10 Mbps) and the frequency of the STCA clock (25 or 50 MHz).

Table 28.9 Settings of the EGP[15:0] Bits (ns)

		STCA Clock Frequency		
Link Transfer Rate		20 MHz	25 MHz	50 MHz
MII	100 Mbps	590	625	695
	10 Mbps	7430	7465	7535
RMII	100 Mbps	770	805	875
	10 Mbps	9230	9265	9335

INGP[15:0] Bits (Input Port Timestamp Latency Setting)

Set the corresponding fixed value in Table 28.10. The timestamp latency differs with the link transfer rate (100 Mbps or 10 Mbps) and the frequency of the STCA clock (25 or 50 MHz).

Table 28.10 Settings of the INGP[15:0] Bits (ns)

		STCA Clock Frequency		
Link Transfer Rate		20 MHz	25 MHz	50 MHz
MII	100 Mbps	980	945	875
	10 Mbps	8180	8145	8075
RMII	100 Mbps	1060	1025	955
	10 Mbps	8980	8945	8875

28.2.80 SYNFP Operation Setting Register (SYCONFR)

Address(es): EPTPC0.SYCONFR E820 59CCh, EPTPC1.SYCONFR E820 5DCCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	TCMOD	—	—	—	FILDIS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	SBDIS	—	—	—	—	TCYC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	TCYC[7:0]	PTP Message Transmission Interval Setting	These bits are used to set the time from the completion of one transmission to the start of the next in cycles of the transmission clock. A value n in these bits means that a transmission interval of n cycles will be secured. No interval is secured if the setting is 00h. The recommended value is 28h (40 cycles).	R/W
b11 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b12	SBDIS	Sync Message Transmission Bandwidth Securing Disable	0: Securing of the bandwidth for the transmission of SYNC messages is enabled (transfer by the EDMAC is given lower priority). 1: Securing of the bandwidth for the transmission of SYNC messages is disabled (transfer by the EDMAC is given higher priority).	R/W
b15 to b13	—	Reserved	These bits are read as 0. The write value should be 0.	R
b16	FILDIS	Receive Message domainNumber Filter Disable	0: Filtering conditions for the reception of PTP messages include comparison with the domainNumber field. 1: Filtering conditions for the reception of PTP messages do not include comparison with the domainNumber field.	R/W
b19 to b17	—	Reserved	These bits are read as 0. The write value should be 0.	R
b20	TCMOD	TC Mode Setting	0: E2E TC 1: P2P TC	R/W
b31 to b21	—	Reserved	These bits are read as 0. The write value should be 0.	R

The SYCONFR register controls operation of the SYNFP module.
Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

TCYC[7:0] Bits (PTP Message Transmission Interval Setting)

The TCYC[7:0] bits are used to secure the waiting time between packets to keep a fixed transmission delay time.
The setting defines the interval from input of the transmission completed signal from the ETHERC to output of the next transmission request as a number of cycles of the transmission clock (which runs at 2.5 MHz if the link transfer rate is 10 Mbps and at 25 MHz if the rate is 100 Mbps).

SBDIS Bit (Sync Message Transmission Bandwidth Securing Disable)

This bit disables securing of bandwidth to increase accuracy of the interval for the transmission of SYNC messages.

TCMOD Bit (TC Mode Setting)

This bit sets the delay mechanism for operation as a transparent clock (TC). The bit setting changes the method of calculating the values used in the correctionField field during operation as a TC.

28.2.81 SYNFP Frame Format Setting Register (SYFORMR)

Address(es): EPTPC0.SYFORMR E820 59D0h, EPTPC1.SYFORMR E820 5DD0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FORM1	FORM0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	FORM0	Ethernet Frame Format Setting	Set this bit to 0 (Ethernet II frame format). 0: Ethernet II frame format	R/W
b1	FORM1	Ethernet/UDP Encapsulation	0: PTP directly over Ethernet 1: PTP over UDP/IPv4	R/W
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R

The SYFORMR register is used to set the format for frame generation by the SYNFP module.
Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

28.2.82 Response Message Reception Timeout Register (RSTOUTR)

Address(es): EPTPC0.RSTOUTR E820 59D4h, EPTPC1.RSTOUTR E820 5DD4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
31 to b0	—	Response Message Reception Timeout Time Setting	A response message not being received within n × 1024 (ns), where n is the setting, is judged to represent a timeout.	R/W

The RSTOUTR register is used to set the time for detection of a timeout in the reception of PTP response messages (Delay_Resp and Pdelay_Resp).
If a Delay_Resp or Pdelay_Resp message is not received within the time specified by this register after transmission of a Delay_Req or Pdelay_Req message, the SYSR.DRPTO flag becomes 1.
The value specified by this register is only reflected in the SYNFP module after the SYRVLDR.STUP bit is set to 1.

28.2.83 PTP Reset Register (PTRSTR)

Address(es): EPTPC.PTRSTR E820 4500h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RESET
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	RESET	EPTPC Software Reset	0: The EPTPC is not reset. 1: The EPTPC is reset.*1	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note 1. Do not access the EPTPC-related registers other than this register while a software reset is being issued.

The PTRSTR register is used to reset the EPTPC.
It takes 64 cycles of the peripheral module clock (Bφ) until initialization of the EPTPC is completed. After the RESET bit is set to 1, wait for 64 cycles of the Bφ before setting its value back to 0.

28.2.84 STCA Clock Select Register (STCSELR)

Address(es): EPTPC.STCSELR E820 4504h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SCLKSEL[2:0]			—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R
b2, b1	—	Reserved	These bits are read as 1. The write value should be1.	R
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b10 to b8	SCLKSEL [2:0]	STCA Clock Select	b10 b8 0 1 0: Input clock from the REF50CK0 pin 0 1 1: Input clock from the REF50CK1 pin 1 0 0: Input clock from the ET0_RX_CLK pin / Input clock from the ET0_SCLKIN pin 1 0 1: Input clock from the ET1_RX_CLK pin / Input clock from the ET1_SCLKIN pin Settings other than above are prohibited.	R/W
b31 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R

The STCSELR register selects the STCA clock signal for use in the EPTPC.
Set a value in this register before starting the EDMAC, ETHERC, or PTPEDMAC. Do not change the settings while the EPTPC is operating.

SCLKSEL[2:0] Bits (STCA Clock Select)

These bits select the STCA clock signal for use in the EPTPC.
Switching of ETn_RX_CLK/ETn_SCLKIN is selected with an Etn_EXOUT_SEL bit of section 51.3.36, Peripheral Pin Function Control Register (PMODEPFS). For details, refer to section 51, GPIO.

28.2.85 1588 Module Bypass Register (BYPASS)

Address(es): EPTPC.BYPASS E820 4508h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BYPASS1
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BYPASS0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	BYPASS0	Ether 0ch 1588 module bypass setting	0: Use the 1588 module of Ethernet channel 0. 1: Bypass the 1588 module of Ethernet channel 0.	R/W
b15 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	
b16	BYPASS1	Ether 1ch 1588 module bypass setting	0: Use the 1588 module of Ethernet channel 1. 1: Bypass the 1588 module of Ethernet channel 1.	
b31 to b17	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: Do not access the bypass register while the Ethernet is operating. The 1588 module can be bypassed by setting the bypass register if EPTPC is not used.

28.3 Operation

The EPTPC is set not to receive (analyze) and transmit (generate) PTP messages after release from the reset state. Accordingly, the EPTPC has no effect on the transmission and reception of frames by the ETHERC and EDMAC at that time.

Settings of the EPTPC registers are required for the ETHERC and EDMAC to be able to use packet filtering by MAC address in the SYNFP modules and for relaying between Ethernet ports by the PRC-TC module. That is, the transmission and reception of PTP messages requires setting of the EPTPC.

Figure 28.4 is a block diagram of the modules involved in transferring and relaying frames.

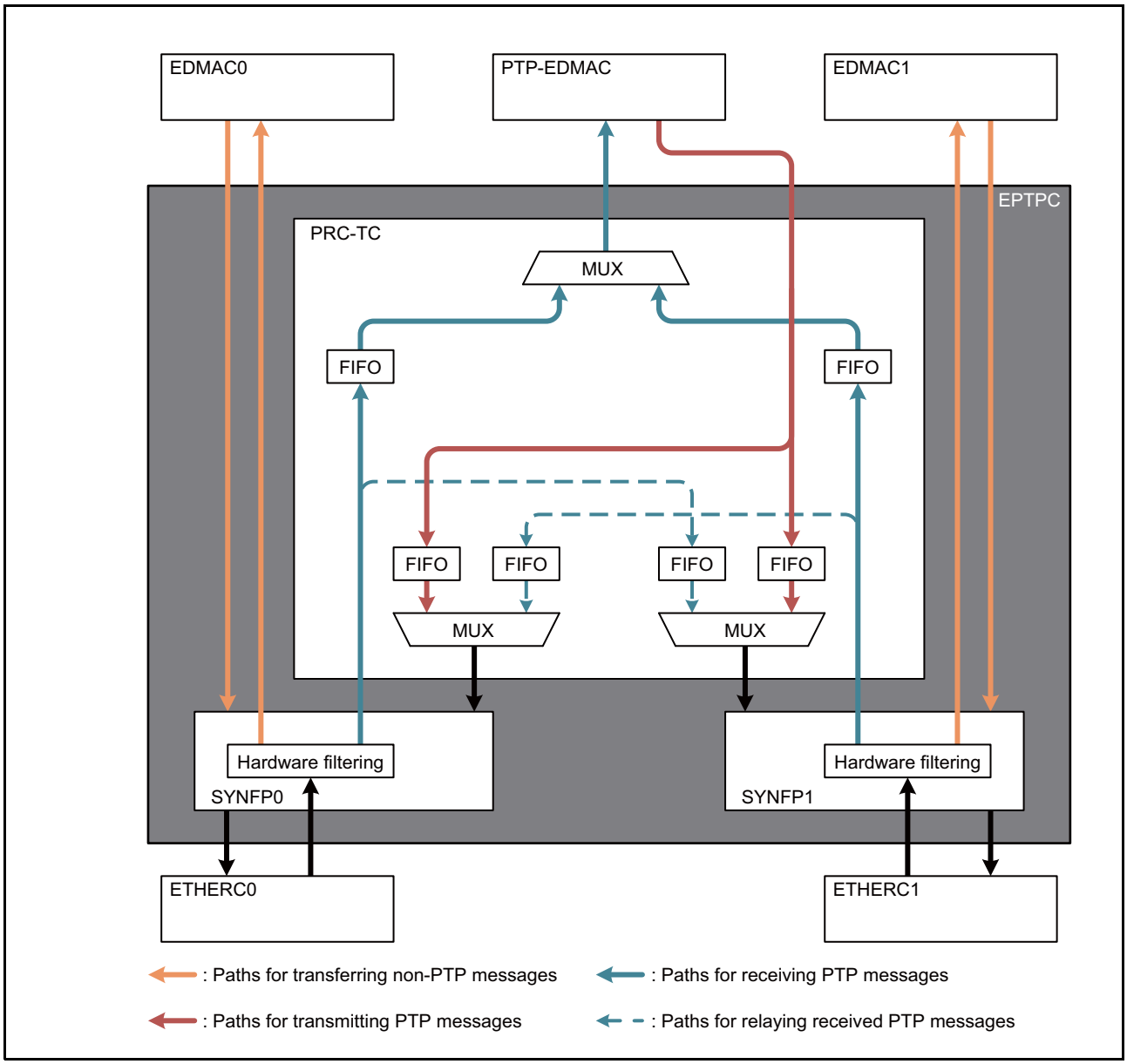


Figure 28.4 Block Diagram of the Modules Involved in Transferring and Relaying Frames

28.3.1 Transmission and Reception and Relaying of Non-PTP Messages

The EPTPC operates in extended promiscuous mode if the setting of the FFLTR.EXTPRM bit is 1 (extended promiscuous mode). In this mode, all frames received by the Ethernet ports are transferred to the EDMAC without filtering.

The EPTPC operates in normal mode if the setting of the FFLTR.EXTPRM bit is 0 (normal operation). In this mode, the SYNFP module applies its hardware filtering function to filter frames received by the Ethernet ports.

Frames for the other node can also be relayed to the Ethernet port on the other side in accord with the settings of the TRNMR.FWD0 and FWD1 bits.

The EPTPC and EDMAC transfer received unicast frames if they are for the given node. When a received frame is for the other node, the EPTPC is able to relay received frames to the other port if relaying between Ethernet ports is enabled.

Operation when multicast frames are received can be selected from the following: frames are transferred to the EDMAC, frames are not transferred to the EDMAC, or frames are transferred to the EDMAC only when the address matches the specified MAC address. The EPTPC also relays multicast frames to the other Ethernet port if relaying between Ethernet ports is enabled.

The EPTPC transfers received broadcast frames to the EDMAC for the receiving Ethernet port. The EPTPC also relays broadcast frames to the other Ethernet port if relaying between Ethernet ports is enabled.

The PRC-TC module, which handles the relaying of frames received by one Ethernet port to the other Ethernet port, has a relaying FIFO buffer for use in this process. Either of two methods for reading out from the relaying FIFO is selectable: store and forward or cut-through. With the cut-through method, reading out from the relaying FIFO starts when it holds more than one frame or more data than a specified threshold value, or when both of these conditions are satisfied. The TRNCTTDR.THVAL[10:0] bits specify the threshold value for reading out from the relaying FIFO.

With the cut-through method, when a frame is handled as an error frame due to a fault (such as malformed data being read) after a read operation starts, there is no procedure for notifying the ETHERC module of the error. Accordingly, the ETHERC may send such error frames over the line as if they were normal frames. Avoid situations of this kind by using the store-and-forward method with the relaying FIFO. With the store-and-forward method, the PRC-TC module internally discards error frames.

28.3.2 Paths for the Transfer of Non-PTP Messages

Messages received through the Ethernet port are transferred to the EDMAC. The PRC-TC module is capable of relaying messages between the Ethernet ports.

Figure 28.5 is a diagram of paths for the transmission and reception and relaying of non-PTP messages.

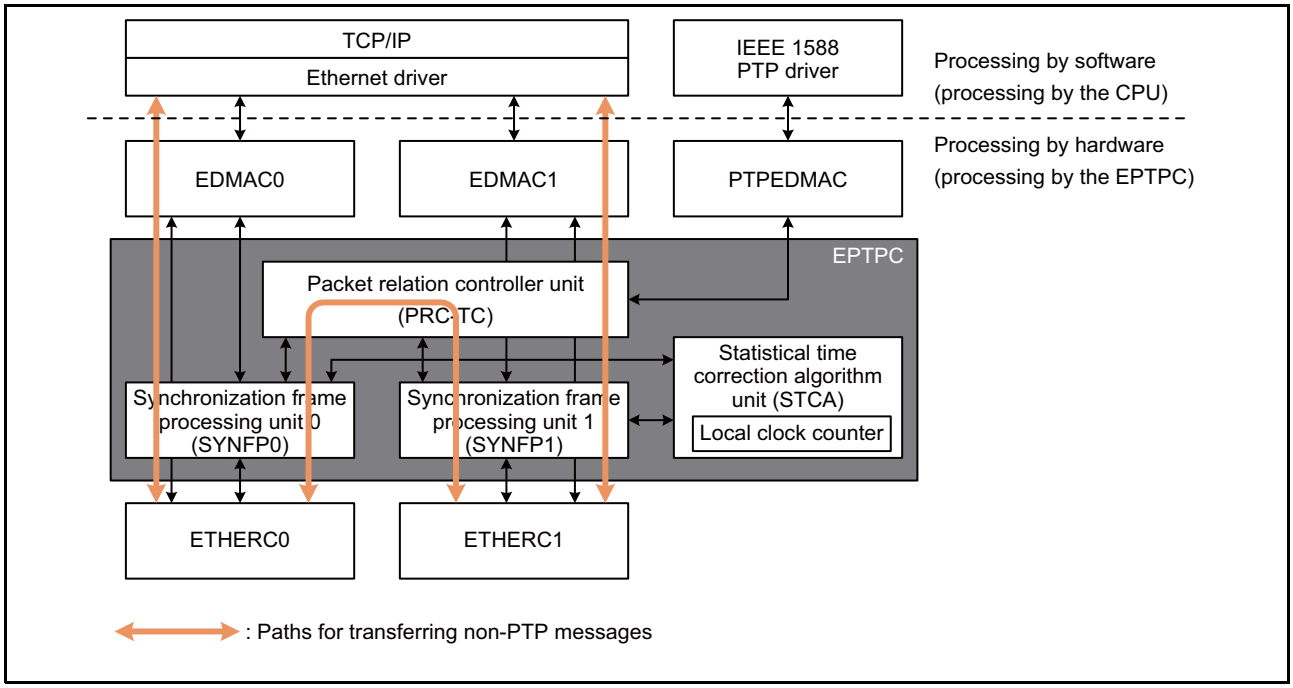


Figure 28.5 Paths for the Transmission and Reception and Relaying of Non-PTP Messages

28.3.3 Transmission and Reception and Relaying of PTP Messages

Hardware of the EPTPC automatically handles analysis of and the extraction of fields from received PTP messages, and generation and transmission of PTP messages. However, software must still handle the transmission of certain PTP messages. Table 28.11 shows the specifications for control over the transmission and reception of the various PTP messages.

Table 28.11 Control over the Transmission and Reception of PTP Messages

Message Type	Message	OC (Ordinary Clock)/BC (Boundary Clock)		TC (Transparent Clock)
		Master	Slave	
Event	Sync	Generation (automatic)	Reception (automatic)	Relaying (automatic)
	Delay_Req	Generation (automatic)	Reception (automatic)	Relaying (automatic)
	Pdelay_Req	Generation and reception (automatic)	Generation and reception (automatic)	Generation and reception (automatic)
	Pdelay_Resp	Generation and reception (automatic)	Generation and reception (automatic)	Generation and reception (automatic)
General	Announce	Generation (automatic)	Reception (software)	Reception (software)
	Follow_Up	—*1	Reception (automatic)	Relaying (automatic)
	Delay_Resp	Packet generation	Reception (automatic)	Relaying (automatic)
	Pdelay_Resp_Follow_Up	—*1	Reception (automatic)	Relaying (automatic)
	Management	Transmission and reception (software)	Transmission and reception (software)	Relaying (automatic)
	Signaling	Transmission and reception (software)	Transmission and reception (software)	Relaying (automatic)

Note 1. Control is not required as the clock in this case is a one-step clock.

28.3.4 Paths for the Transfer of PTP Messages

Paths for the transfer of PTP messages differ according to whether transfer requires processing by software or is automatically processed by hardware.

28.3.4.1 Paths for the Transfer of PTP Messages Requiring Processing by Software

Paths for the transfer of PTP messages where transfer requires processing by software are shown in Figure 28.6. The figure shows paths for message, clock-type, and process combinations for which "(software)" is indicated inTable 28.11.

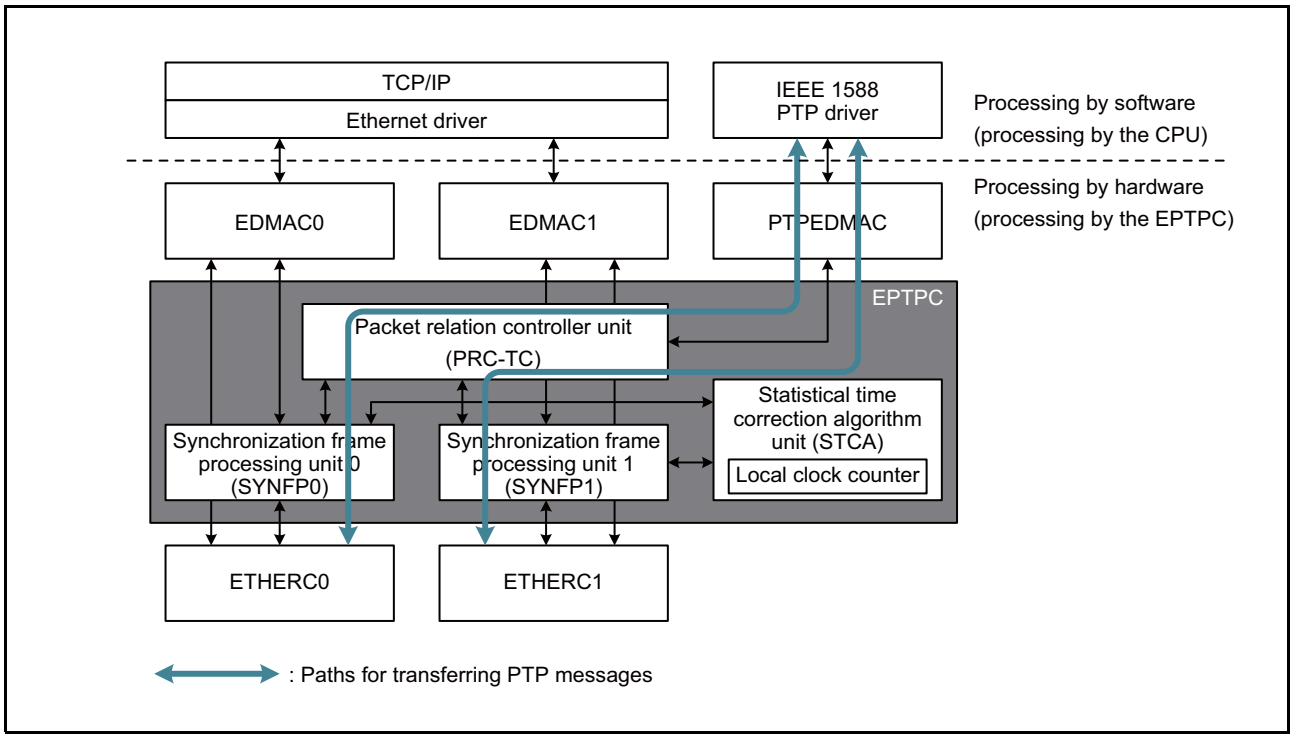


Figure 28.6 Paths for the Transfer of PTP Messages Requiring Processing by Software

28.3.4.2 Paths for the Transfer of PTP Messages Automatically Handled by Hardware

In the case of PTP messages for which hardware automatically handles the processing, the SYNFP modules handle transmission and reception, and the PRC-TC module handles relaying.

(1) Generation of and Response to PTP Messages by Hardware

Paths for transfer in the automatic generation of and response to PTP messages by the SYNFP modules are shown in Figure 28.7. The paths in the figure are used for operations “Generation (automatic)”, “Reception (automatic)”, and “Generation and reception (automatic)” indicated in Table 28.11.

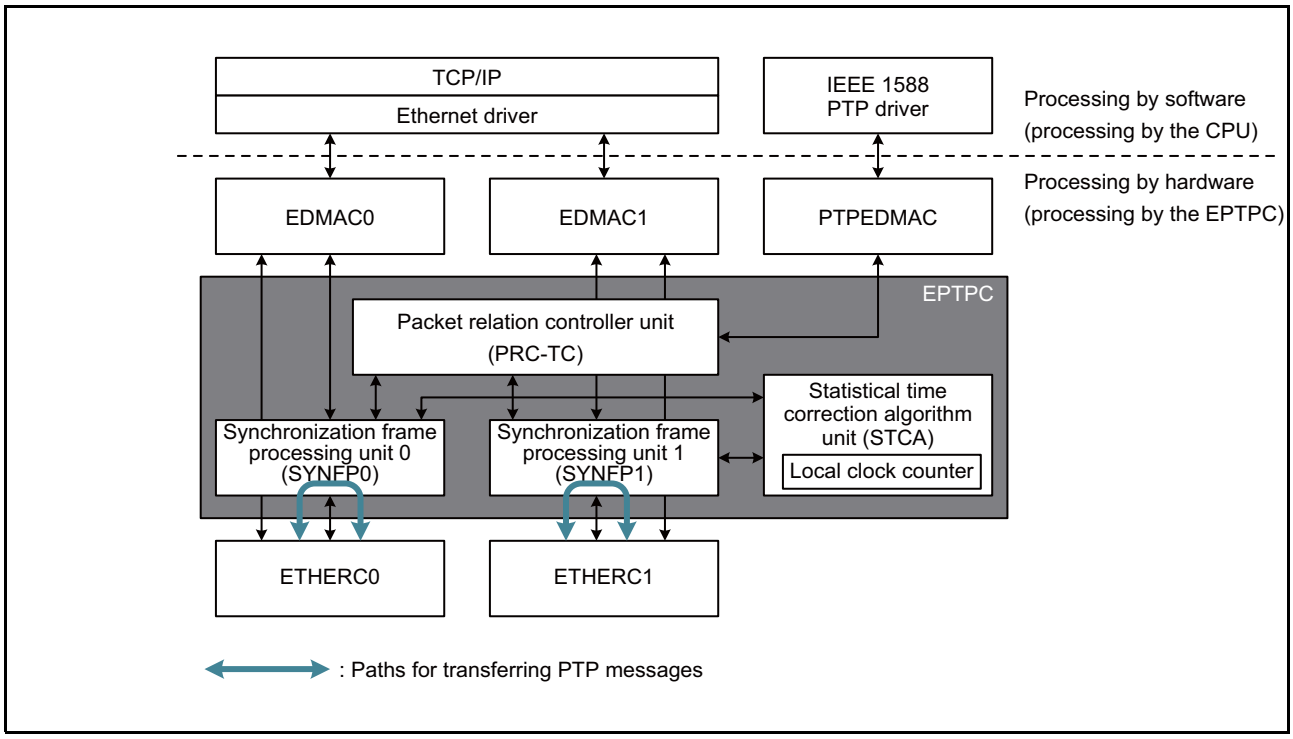


Figure 28.7 Paths for the Generation of and Response to PTP Messages by Hardware

(2) Relaying of PTP Messages by Hardware

Figure 28.8 shows paths for the relaying of PTP messages by the PRC-TC module. The paths in the figure are used for the “Relaying” operation indicated in Table 28.11.

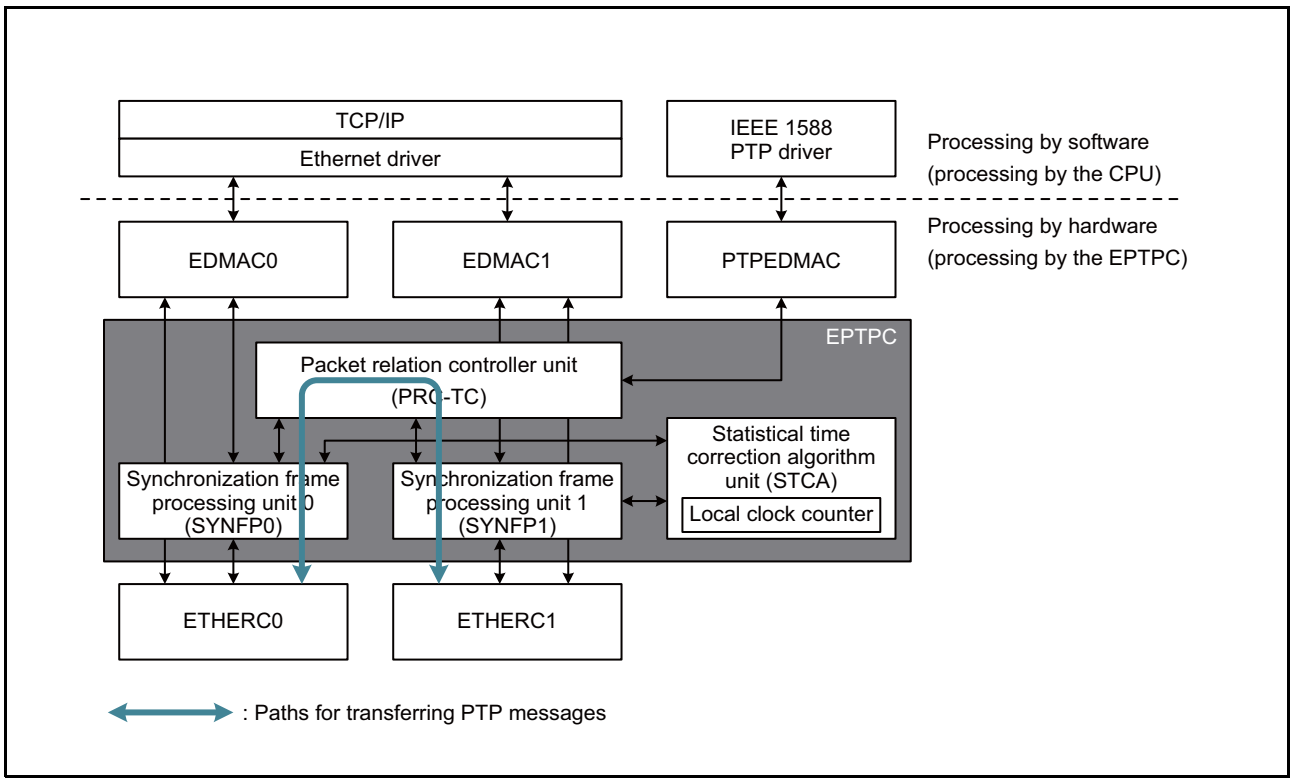


Figure 28.8 Paths for the Relaying of PTP Messages by Hardware

28.3.5 Clock Devices

The EPTPC can operate as the clock device defined in IEEE 1588.

28.3.5.1 End-to-End (E2E)

(1) Master

PTP messages are transmitted and received as listed below in operation as an end-to-end (E2E) master.

Table 28.12 List of PTP Message Processing (by an E2E Master)

Message Type	Message	Details of Processing
Event	Sync	Sync messages are transmitted at the fixed interval specified by the SYTLIR.SYNC[7:0] bits.
	Delay_Req	When this message is received, a Delay_Resp message is transmitted in response.
	Pdelay_Req	—
	Pdelay_Resp	—
General	Announce	Announce messages are transmitted at the fixed interval specified by the SYTLIR.ANCE[7:0] bits.
	Follow_Up	—
	Delay_Resp	This is transmitted as the response to a received Delay_Req messages.
	Pdelay_Resp_Follow_Up	—
	Management	Management messages are transmitted and received by software via the PTPEDMAC.
	Signaling	Signaling messages are transmitted and received by software via the PTPEDMAC.

(2) Slave

PTP messages are transmitted and received as listed below in operation as an E2E slave, and the calculated offsetFromMaster is used to correct the local time information.

Table 28.13 List of PTP Message Processing (by an E2E Slave)

Message Type	Message	Details of Processing
Event	Sync	The offsetFromMaster value is calculated when this message is received if twoStepFlag in flagField was FALSE (one-step clock).
	Delay_Req	Delay_Req messages are transmitted at random intervals from 0 to the time specified by the SYTLIR.DREQ[7:0] bits × 2.
	Pdelay_Req	—
	Pdelay_Resp	—
General	Announce	Announce messages are transmitted by software via the PTPEDMAC.
	Follow_Up	The offsetFromMaster value is calculated when this message is received if twoStepFlag in flagField of the most recently received Sync message was TRUE (two-step clock).
	Delay_Resp	The meanPathDelay value is calculated when this message is received.
	Pdelay_Resp_Follow_Up	—
	Management	Management messages are transmitted and received by software via the PTPEDMAC.
	Signaling	Signaling messages are transmitted and received by software via the PTPEDMAC.

28.3.5.2 Peer-to-Peer (P2P)

(1) Master

PTP messages are transmitted and received as listed below in operation as a peer-to-peer (P2P) master.

Table 28.14 List of PTP Message Processing (by a P2P Master)

Message Type	Message	Details of Processing
Event	Sync	Timestamps for transmission are transmitted at the fixed interval specified by the SYTLIR.SYNC[7:0] bits. When this message is received, a Pdelay_Resp message is transmitted in response.
	Delay_Req	—
	Pdelay_Req	Pdelay_Req messages are transmitted at the fixed interval specified by the SYTLIR.DREQ[7:0] bits.
	Pdelay_Resp	This is transmitted as the response to a received Pdelay_Req messages. The meanPathDelay value is calculated when this message is received if twoStepFlag in flagField was FALSE (one-step clock).
General	Announce	Announce messages are transmitted at the fixed interval specified by the SYTLIR.ANCE[7:0] bits.
	Follow_Up	—
	Delay_Resp	—
	Pdelay_Resp_Follow_Up	The meanPathDelay value is calculated when this message is received if twoStepFlag in flagField of the most recently received Pdelay_Resp message was TRUE (two-step clock).
	Management	Management messages are transmitted by software via the PTPEDMAC.
	Signaling	Signaling messages are transmitted by software via the PTPEDMAC.

(2) Slave

PTP messages are transmitted and received as listed below in operation as a P2P slave, and the calculated offsetFromMaster is used to correct the local time information.

Table 28.15 List of PTP Message Processing (by a P2P Slave)

Message Type	Message	Details of Processing
Event	Sync	The offsetFromMaster value is calculated when this message is received if twoStepFlag in flagField was FALSE (one-step clock).
	Delay_Req	—
	Pdelay_Req	Pdelay_Req messages are transmitted at the fixed interval specified by the SYTLIR.DREQ[7:0] bits. When this message is received, a Pdelay_Resp message is transmitted in response.
	Pdelay_Resp	This is transmitted as the response to a received Pdelay_Req messages. The meanPathDelay value is calculated when this message is received if twoStepFlag in flagField was FALSE (one-step clock).
General	Announce	Announce messages are transmitted by software via the PTPEDMAC.
	Follow_Up	The offsetFromMaster value is calculated when this message is received if twoStepFlag in flagField of the most recently received Sync message was TRUE (two-step clock).
	Delay_Resp	—
	Pdelay_Resp_Follow_Up	The meanPathDelay value is calculated when this message is received if twoStepFlag in flagField of the most recently received Pdelay_Resp message was TRUE (two-step clock).
	Management	Management messages are transmitted and received by software via the PTPEDMAC.
	Signaling	Signaling messages are transmitted and received by software via the PTPEDMAC.

28.3.5.3 Ordinary Clock (OC)

PTP messages are transmitted and received through one Ethernet port in operation as an ordinary clock.
An ordinary clock operates as the grand master clock or as a slave clock in the master-slave hierarchy.
For operation as an E2E master, E2E slave, P2P master, or P2P slave, see section 28.3.7, Operation as an E2E Master, section 28.3.8, Operation as an E2E Slave, section 28.3.10, Operation as a P2P Master, and section 28.3.11, Operation as a P2P Slave.

28.3.5.4 Boundary Clock (BC)

PTP messages are transmitted and received through both ports in operation as a boundary clock.
One port operates as a slave in synchronization with the root master clock and the other operates as the master that delivers time information synchronized with the master clock.
Both ports can also operate as masters.
For operation as an E2E master, E2E slave, P2P master, or P2P slave, see section 28.3.7, Operation as an E2E Master, section 28.3.8, Operation as an E2E Slave, section 28.3.10, Operation as a P2P Master, and section 28.3.11, Operation as a P2P Slave.

28.3.5.5 Transparent Clock (TC)

(1) E2E TC

In operation as an E2E transparent clock, received PTP-primary and PTP-pdelay messages are relayed.

Table 28.16 List of Packet Processing (by an E2E TC)

Message Type	Message	Details of Processing
Event	Sync	The packet residence time in the clock device is added when these messages are relayed.
	Delay_Req	
	Pdelay_Req	
	Pdelay_Resp	
General	Announce	These messages are relayed.
	Follow_Up	
	Delay_Resp	
	Pdelay_Resp_Follow_Up	
	Management	These messages are relayed. Messages are also transmitted and received by software via the PTPEDMAC.
	Signaling	

(2) P2P TC

In operation as a P2P transparent clock, PTP-primary messages other than Delay_Req and Delay_Resp are relayed.

Table 28.17 List of Packet Processing (by a P2P TC)

Message Type	Message	Details of Processing
Event	Sync	The packet residence time in the clock device is added when this message is relayed.
	Delay_Req	—
	Pdelay_Req	Pdelay_Req messages are transmitted at the fixed interval specified by the SYTLIR.DREQ[7:0] bits. When this message is received, a Pdelay_Resp message is transmitted in response.
	Pdelay_Resp	This is transmitted as the response to a received Pdelay_Req messages. The meanPathDelay value is calculated when this message is received if twoStepFlag in flagField was FALSE (one-step clock).
General	Announce	These messages are relayed.
	Follow_Up	
	Delay_Resp	—
	Pdelay_Resp_Follow_Up	The meanPathDelay value is calculated when this message is received if twoStepFlag in flagField of the most recently received Pdelay_Resp message was TRUE (two-step clock).
	Management	These messages are relayed. Messages are also transmitted and received by software via the PTPEDMAC.
	Signaling	

28.3.6 EPTPC Initialization

Transmitting and receiving PTP messages requires settings in various registers of the EPTPC.
Table 28.18 is a list of the registers for which settings are required. Set the registers corresponding to the Ethernet port used. Also set the registers listed in Table 28.19 if the UDP and IPv4 are used for the frame format of the PTP messages.

Table 28.18 List of the Registers for which Settings are Required

Register Name	Settings	Description
STCFR	Example: 0000 0002h	The value for 50 MHz is given as an example. Three other settings are also available.
STCSELR	Example: 0000 0206h	The value for the input clock from the REF50CK0 pin is given as an example.
PRMACRU0, PRMACRL0	As desired	This setting is not required if Ethernet port 0 is not used.
PRMACRU1, PRMACRL1	As desired	This setting is not required if Ethernet port 1 is not used.
SYCONFR	Example: 0000 0028h	The setting differs with the type of PTP clock operation.
SYMACRU, SYMACRL	As desired	
SYSPVRR	0000 0002h	transportSpecific and version fields
SYDOMR	As desired	
SYCIDRU, SYCIDRL	As desired	Set the same value for Ethernet ports 0 and 1.
SYPNUMR	0000 0001h or 0000 0002h	If the PTP clock operates as an OC, the setting is 0000 0001h. If it operates as a BC or TC, set 0000 0001h for one port and 0000 0002h for the other.
PPMACRU, PPMACRL	01:1B:19:00:00:00	MAC address for PTP-primary messages
PDMACRU, PDMACRL	01:80:C2:00:00:0E	MAC address for PTP-pdelay messages
DASYMRU, DASYMRL	0000 0000h	-
TSLATR	As desired	Depends on the link transfer rate and STCA clock frequency.
SYFORMR	As desired	Four settings are available.
SYLLCCTLR	0000 0003h	LLC-CTL field value for Ethernet frames
PETYPER	0000 88F7h	Ethertype for PTP messages

Table 28.19 List of the Registers for which Settings are Required (Additional Settings Required when UDP or IPv4 is to be Used)

Register Name	Settings	Description
SYIPADDRR	As desired	Local IP address
PRIPR	E000 0181h	IP address for PTP-primary messages
PETOSR	As desired	Set the highest allowable traffic class selector codepoint as the value for the differentiated service (DS) field.
PGTOSR	As desired	
PPTTLR	As desired	TTL field value for PTP-primary messages
PEUDPR	0000 013Fh	UDP port number for event messages
PGUDPR	0000 0140h	UDP port number for general messages
PDIPR	E000 006Bh	IP address for PTP-pdelay messages
PDTTLR	0000 0001h	TTL field value for PTP-pdelay messages

In operation as an OC or BC, set registers as shown below in order to transfer received Announce, Management, and Signaling messages to the PTPEDMAC.

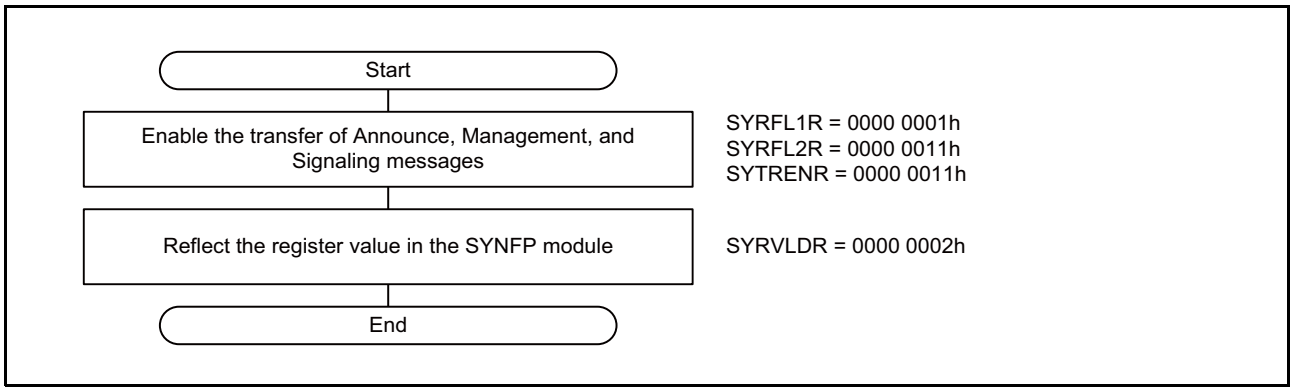


Figure 28.9 Common Settings for PTP Devices

28.3.7 Operation as an E2E Master

28.3.7.1 Preparatory Setting

Table 28.20 lists the registers for use in operation as an E2E master.
If the EPTPC operates as an OC or as a BC with both ports serving as masters, set the initial value of the time information in advance. Refer to section 28.2.17, Local Clock Counter Initial Value Registers (LCIVRU, LCIVRM, LCIVRL) for the initial value of the time information. Reflecting the values set in the registers requires setting the SYRVLDR.STUP or ANUP bit to 1.

Table 28.20 Registers for Use in Operation as an E2E Master

Register Name	SYRVLDR Register Bits Used for Loading Direction	Settings	Description
SYNFR	STUP	0000 0000h	flagField for Sync messages
SYTLIR	STUP ANUP	Example: 0000 0001h	Delay_Resp: 1s Sync: 1s Announce: 2s
ANFR	ANUP	0000 0000h	flagField for Announce messages
GMPR	ANUP	As desired	
GMCQR	ANUP	As desired	
GMIDR	ANUP	As desired	
CUOTSR	ANUP	As desired	timeSource: Internal Oscillator
SRR	ANUP	As desired	If the EPTPC operates as an OC or as a BC with both ports serving as masters, set this register to 0000 0000h. If the EPTPC operates as a BC in the combination of a slave and master, set this register to the StepsRemoved field value of Announce messages received by the slave plus one.
SYRFL1R	STUP	0000 4001h	This enables the processing of Delay_Req messages by the SYNFP module.
SYRFL2R	STUP	0000 0011h	This enables the transfer of Signaling and Management messages to the PTPEDMAC.
SYTRENR	STUP	0000 0011h	This enables the transmission of Sync and Announce messages.

28.3.7.2 Procedure for Starting Operations

Figure 28.10 shows the procedure for settings to start operation as an E2E master.

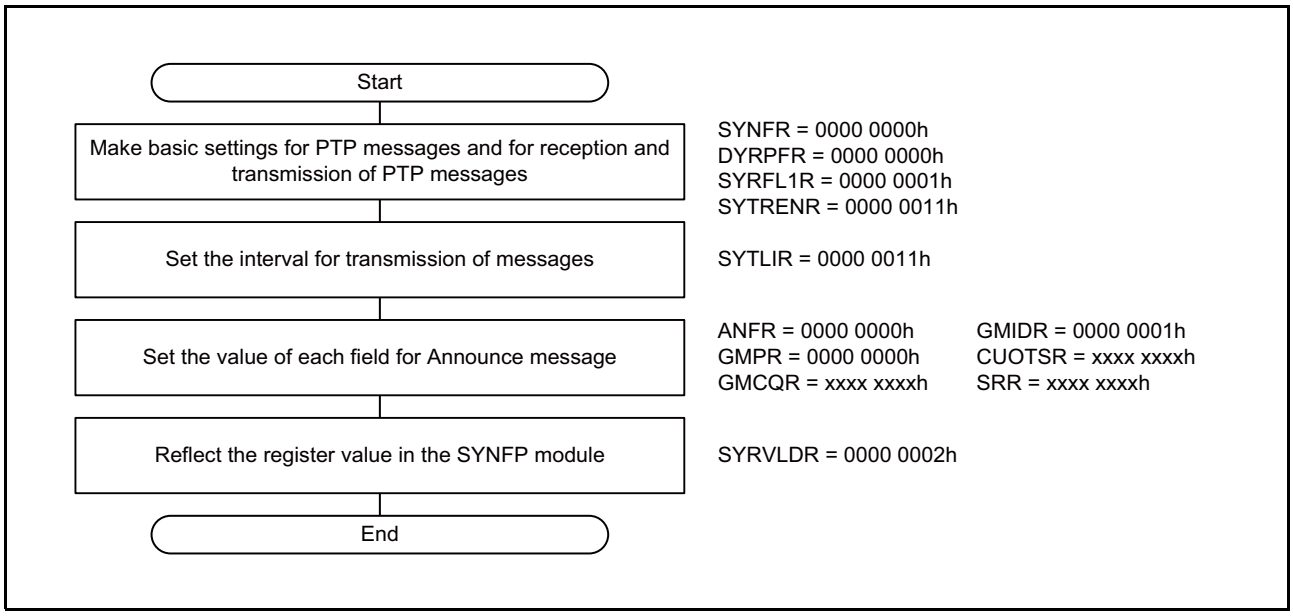


Figure 28.10 Procedure for Starting Operation as an E2E Master

28.3.7.3 Procedure for Changing the Settings

Increases in the frequency of receiving Delay_Req messages due to network conditions may lead to an overflow of the FIFO buffer which is receiving the Delay_Req messages. In such cases, change the value of the logMessageInterval field of Delay_Resp messages so that the slave sending the Delay_Req messages lengthens the interval between the messages. Figure 28.11 shows the procedure for changing the value of the logMessageInterval field.

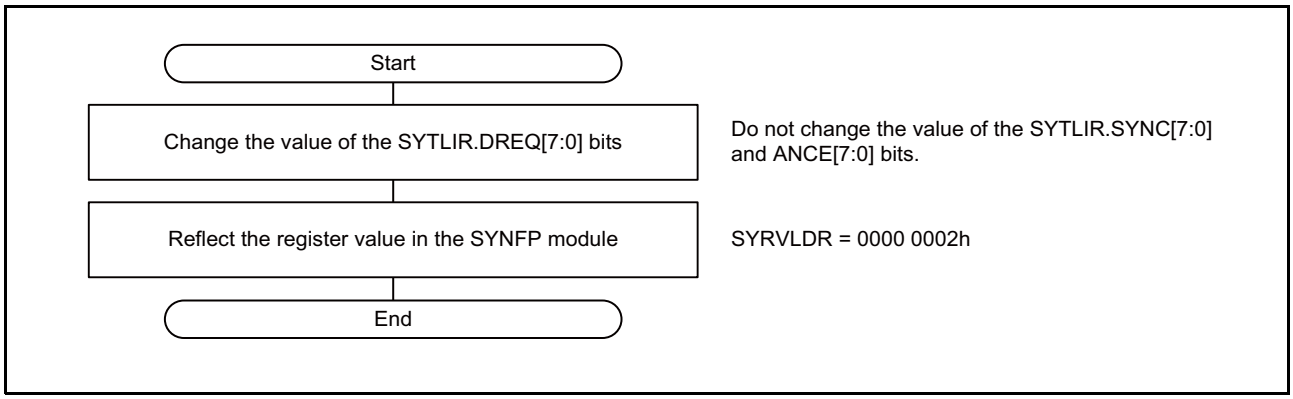


Figure 28.11 Procedure for Changing the Value of the logMessageInterval Field for Delay_Resp Messages

28.3.7.4 Procedure for Stopping Operations

Figure 28.12 shows the procedure for stopping operation as an E2E master. To confirm that the operation as the E2E master is completely stopped, read the SYSR.GENDN flag and RESDN flag to check that generating messages and sending responses are completely stopped.

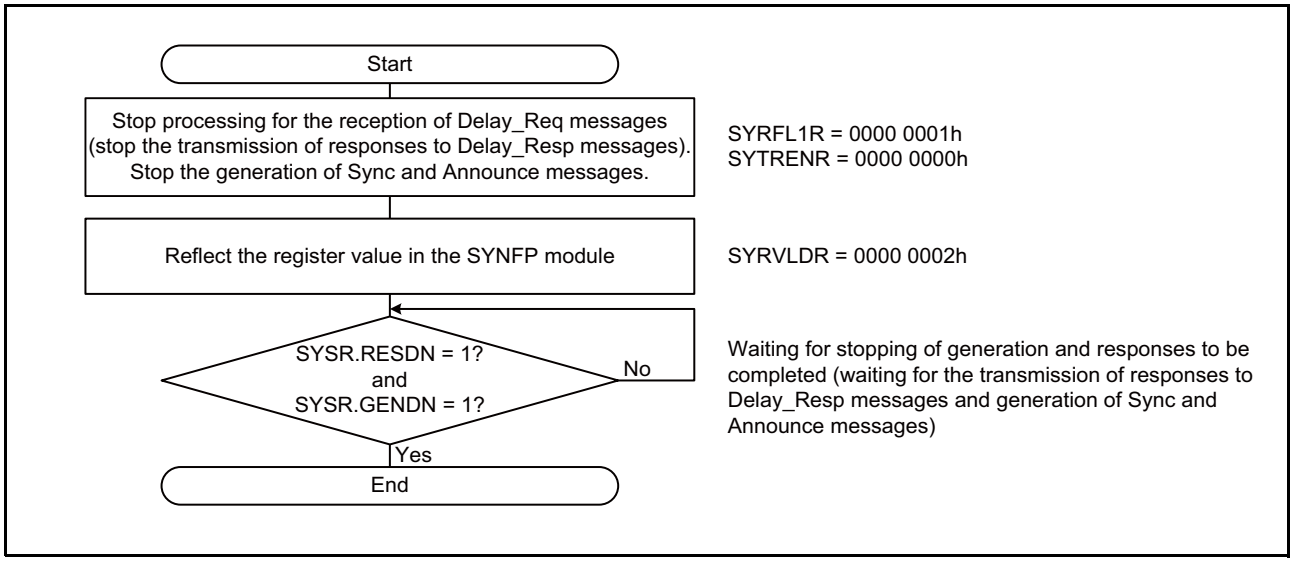


Figure 28.12 Procedure for Stopping Operation as an E2E Master

28.3.8 Operation as an E2E Slave

28.3.8.1 Preparatory Setting

Table 28.21 lists the registers for use in operation as an E2E slave.
Reflecting the value set in the register in SYNFP operations requires setting the SYRVLDR.STUP, ANUP, or BMUP bit to 1.

Table 28.21 Registers for Use in Operation as an E2E Slave

Register Name	SYRVLDR Register Bits Used for Loading Direction	Settings	Description
MTCID	BMUP	As desired	This is the clockIdentity value of the master clock that provides synchronization.
MTPID	BMUP	As desired	This is the portNumber value of the master clock that provides synchronization.
SYTLIR	ANUP BMUP	Example: 0000 0000h	Delay_Resp: 1 s*1
RSTOUTR	STUP	As desired	
SYNTOR	—	As desired	
SYRFL1R	STUP	0004 0441h	This enables the reception of Delay_Resp, Follow_Up, and Sync messages and the transfer of Announce messages to the PTPEDMAC.
SYRFL2R	STUP	0000 0011h	This enables the transfer of Signaling and Management messages to the PTPEDMAC.
SYTRENR	STUP	0000 0100h	This enables the generation of Delay_Req messages.

Note 1. In the reception of Delay_Resp messages by an E2E slave, the SYTLIR.DREQ[7:0] bits must be adjusted if the value of the SYRLIR.DRESP[7:0] flags is to be altered. The SYTLIR.DREQ[7:0] bits specify a value in the range from -7 to 6. Set the SYTLIR.DREQ[7:0] bits to -7 if the value indicated by the SYRLIR.DRESP[7:0] flags is less than or equal to -8 and to 6 if the value indicated by the SYRLIR.DRESP[7:0] flags is greater than or equal to 7.

28.3.8.2 Procedure for Starting Operations

Figure 28.13 shows the procedure for settings to start operation as an E2E slave.

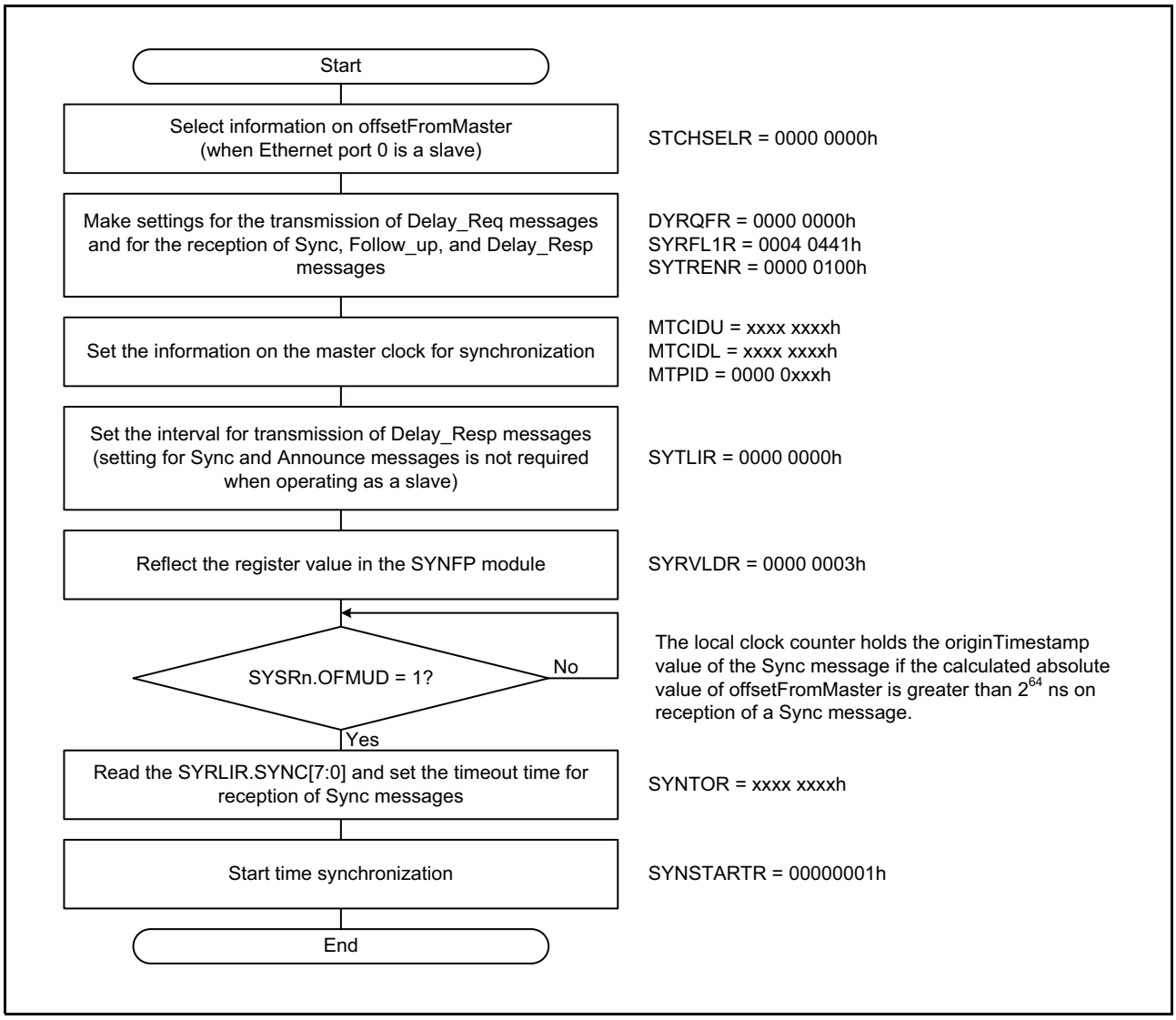


Figure 28.13 Procedure for Starting Operation as an E2E Slave

28.3.8.3 Procedure for Changing the Settings

IEEE 1588 stipulates that the average interval for the transmission of Delay_Req messages must be adjusted in response to changes in the value of the logMessageInterval field of received Delay_Resp messages. The EPTPC sets the SYSR.INTCHG flag to 1 if the logMessageInterval value of a received message differs from that of the previous message.

When this happens, set the SYTLIR.DREQ[7:0] bits to the value of the SYRLIR.DRESP[7:0] bits.

The SYTLIR.DREQ[7:0] bits specify a value in the range from -7 to 6. Set the SYTLIR.DREQ[7:0] bits to -7 if the value indicated by the SYRLIR.DRESP[7:0] flags is less than or equal to -8 and to 6 if the value indicated by the SYRLIR.DRESP[7:0] bits is greater than or equal to 7.

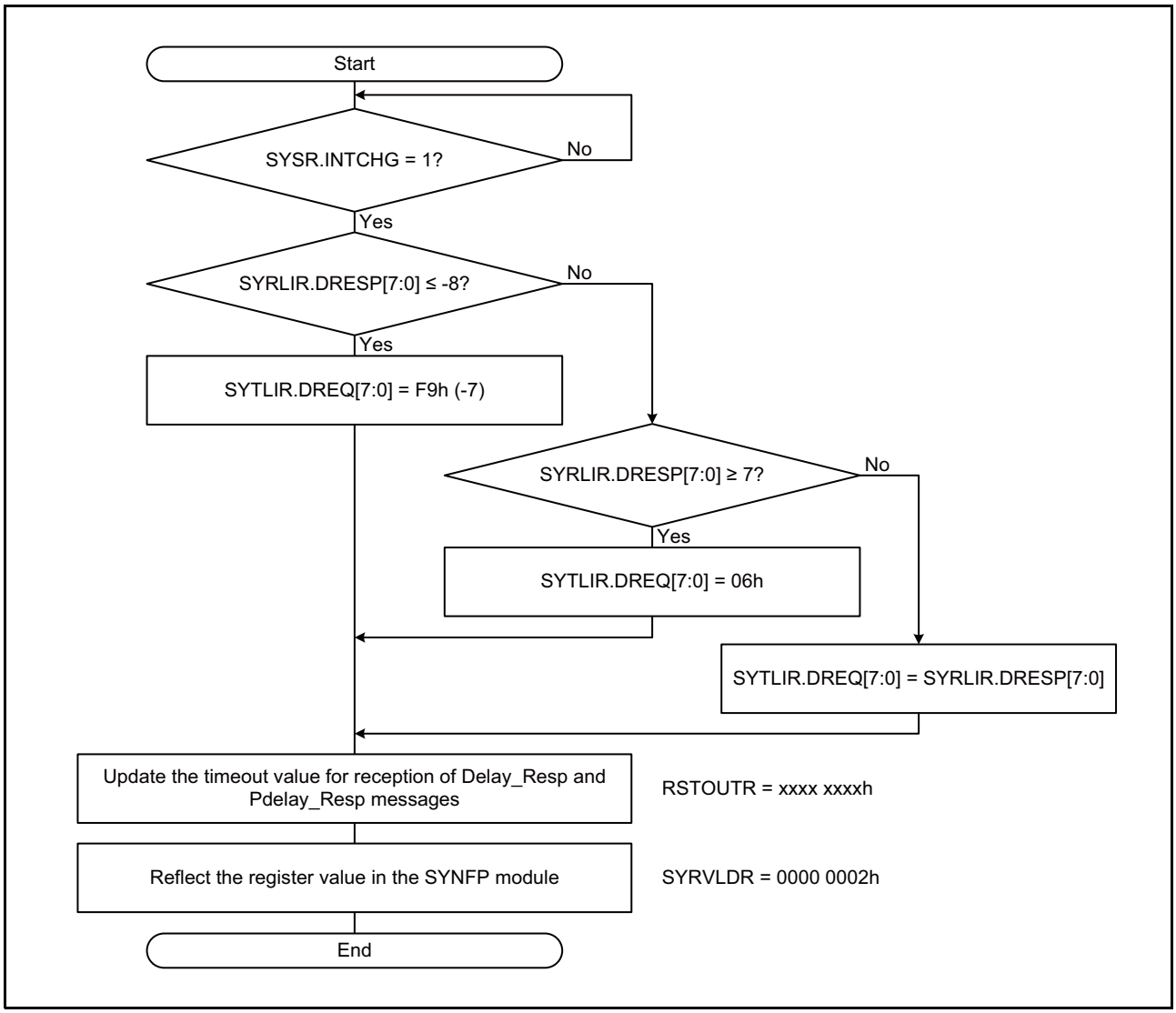


Figure 28.14 Procedure for Changing the Interval for Transmission of Delay_Req Messages

28.3.8.4 Procedure for Stopping Operations

Figure 28.15 shows the procedure for stopping operation as an E2E slave. To confirm that operation as an E2E slave is completely stopped, read the SYSR.GENDN flag to check that generation is completely stopped.

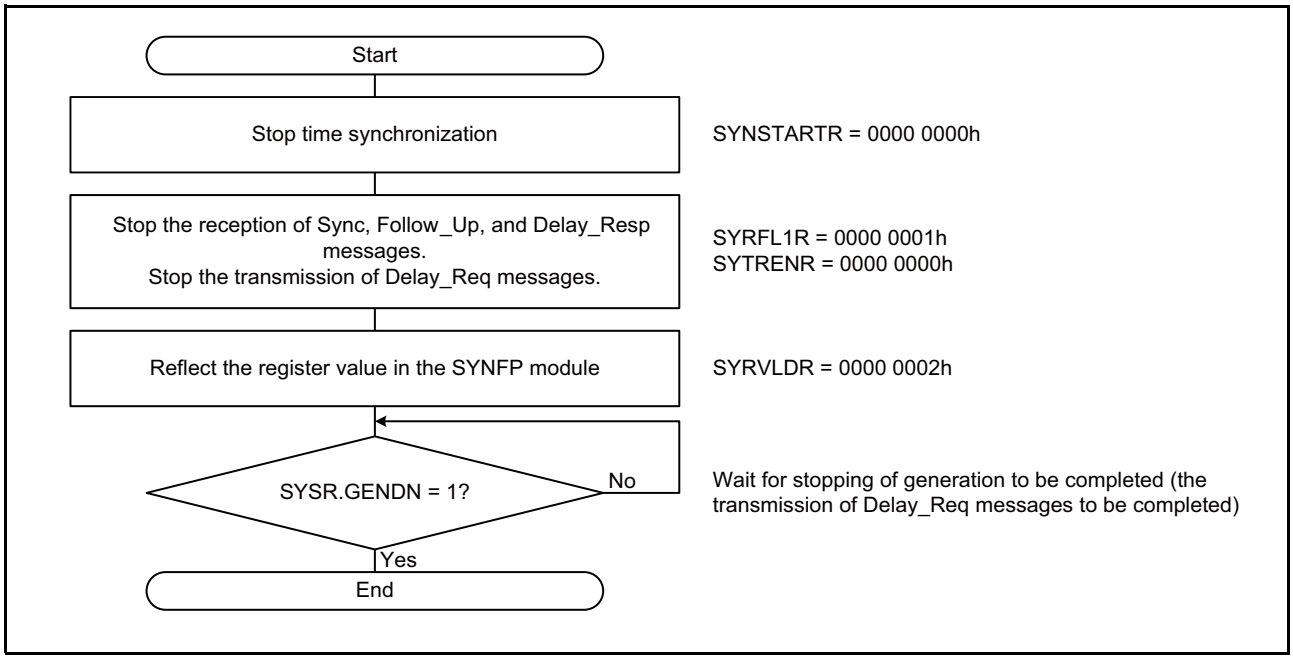


Figure 28.15 Procedure for Stopping Operation as an E2E Slave

28.3.9 P2P Operation (Common to Master and Slave)

If the EPTPC is to be operated as a P2P, the SYNFP module handles the processing of PTP-pdelay messages regardless of whether operation is to be as a master or slave. The interval for Pdelay_Req transmission and the parameters for monitoring of Pdelay_Resp messages must be set at the same time. Registers for use in operation as a P2P are listed in Table 28.22.

Table 28.22 Registers for Use in Operation as a P2P

Register Name	SYRVLDR Register Bits Used for Loading Direction	Settings	Description
MTCID	BMUP	As desired	This is the clockIdentity value of the synchronized master clock.
MTPID	BMUP	As desired	This is the portNumber value of the synchronized master clock.
SYTLIR	ANUP STUP	0000 0000h	Announce: — Sync: — Pdelay_Req: 1 s
RSTOUTR	STUP	As desired	
SYRFL1R	STUP	4440 0001h	This enables the reception of Pdelay_Req, Pdelay_Resp, and Pdelay_Resp_Follow_Up messages and the transfer of Announce messages to the PTPEDMAC.
SYRFL2R	STUP	0000 0011h	This enables the transfer of Signaling and Management messages to the PTPEDMAC.
SYTRENR	STUP	0000 1000h	This enables the generation of Pdelay_Req messages.

28.3.9.1 Procedure for Starting Operations

Figure 28.16 shows the procedure for starting operation as a P2P (sending and receiving PTP-pdelay messages).

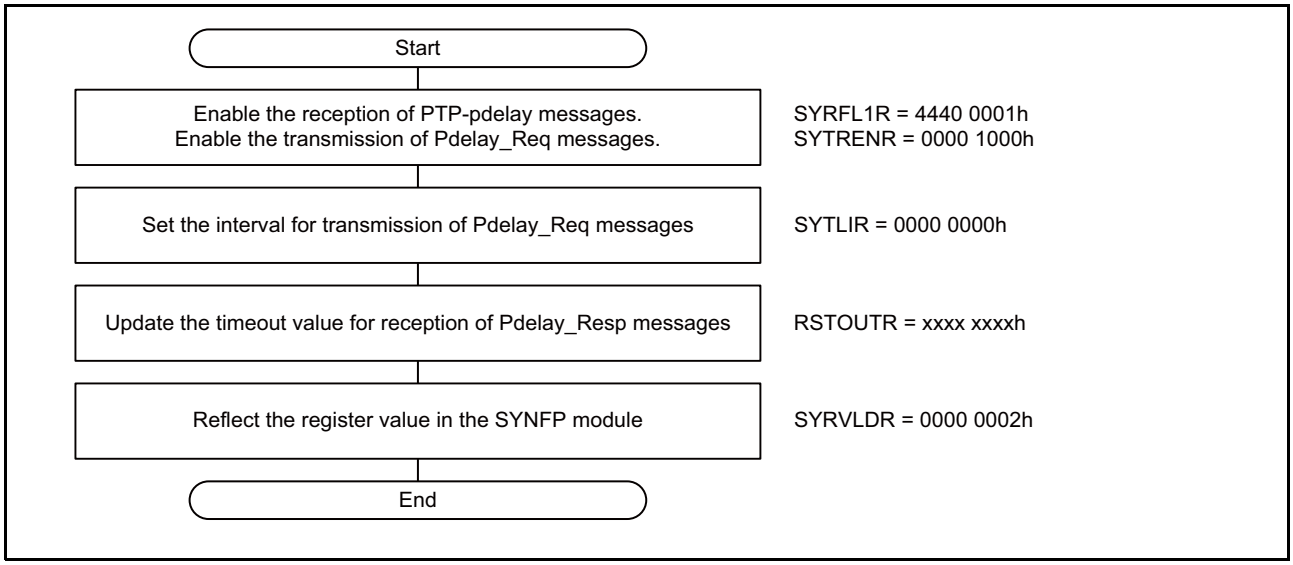


Figure 28.16 Procedure for Starting Operation as a P2P

28.3.9.2 Procedure for Stopping Operations

Figure 28.17 shows the procedure for stopping operation as a P2P (sending and receiving PTP-pdelay messages).

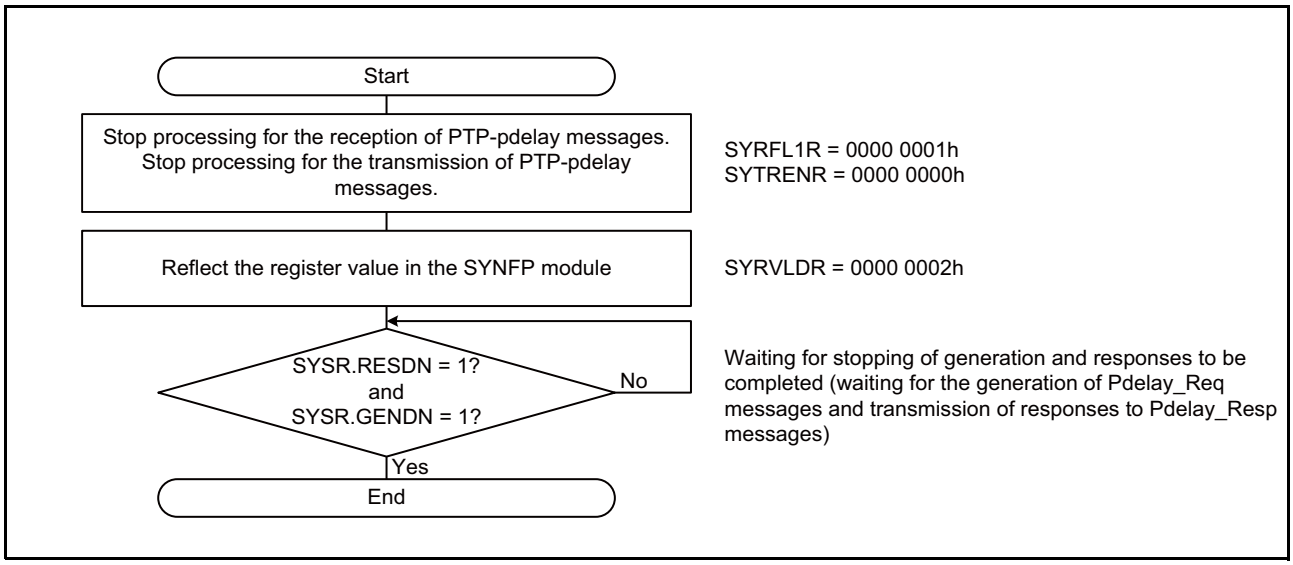


Figure 28.17 Procedure for Stopping Operation as a P2P

28.3.10 Operation as a P2P Master

When the EPTPC operates as OC or BC that uses both ports as the master, set the initial value of time information in advance as required. Refer to section 28.2.17, Local Clock Counter Initial Value Registers (LCIVRU, LCIVRM, LCIVRL) for the initial value of time information. Registers for use in operation as a P2P master are listed in Table 28.23.

Table 28.23 Registers for Use in Operation as a P2P Master

Register Name	SYRVLDR Register Bits Used for Loading Direction	Settings	Description
SYCONFR	—	0000 0028h	
ANFR	ANUP	0000 0000h	flagField for Announce messages
SYNFR	STUP	0000 0000h	flagField for Sync messages
SYTLIR	ANUP STUP	Example: 0000 0001h	Announce: 2 s Sync: 1 s Pdelay_Req: 1 s
GMPR	ANUP	As desired	Grandmaster Priority1 and Priority2
GMCQR	ANUP	As desired	Grandmaster Quality
GMIDR	ANUP	As desired	Grandmaster Identity
CUOTSR	ANUP	As desired	currentUtcOffset, timeSource
SRR	ANUP	As desired	StepsRemoved
RSTOUTR	STUP	As desired	
SYRFL1R	STUP	4440 0000h	This enables the reception of Pdelay_Req, Pdelay_Resp, and Pdelay_Resp_Follow_Up messages.
SYRFL2R	STUP	0000 0011h	This enables the transfer of Signaling and Management messages to the PTPEDMAC.
SYTRENR	STUP	0000 1011h	This enables the transmission of Pdelay_Req, Sync, and Announce messages.

28.3.10.1 Procedure for Starting Operations

When transmission of Sync and Announce messages is started during operation as a P2P (sending and receiving PTP-pdelay messages), the EPTPC operates as a P2P master. Figure 28.18 shows the procedure for starting operation as a P2P mas-ter.

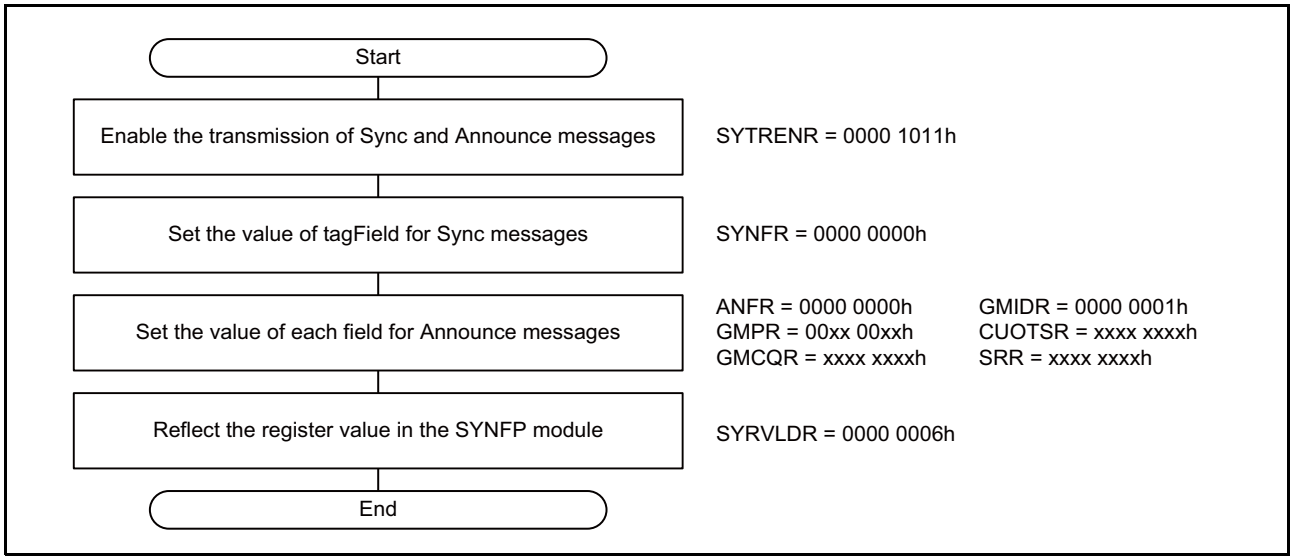


Figure 28.18 Procedure for Starting Operation as a P2P Master

28.3.10.2 Procedure for Stopping Operations

Figure 28.19 shows the procedure for stopping the transmission of Sync and Announce messages to stop operation as a P2P master.

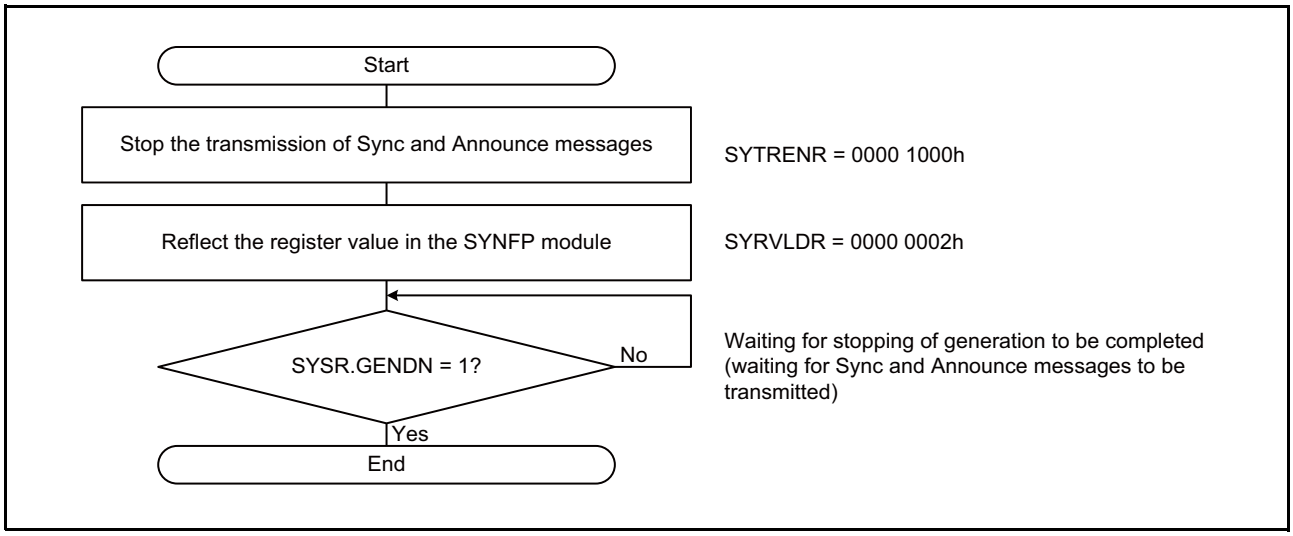


Figure 28.19 Procedure for Stopping Operation as a P2P Master

28.3.11 Operation as a P2P Slave

Setting a SYNFP module to receive Sync messages and Follow_Up messages during P2P operation obtains operation as a P2P slave. Information on the master clock for synchronization must be specified.
Table 28.24 lists the registers for use in operation as a P2P slave.

Table 28.24 Registers for Use in Operation as a P2P Slave

Register Name	SYRVLDR Register Bits Used for Loading Direction	Settings	Description
MTCID	BMUP	As desired	This is the clockIdentity value of the synchronized master clock.
MTPID	BMUP	As desired	This is the portNumber value of the synchronized master clock.
RSTOUTR	STUP	As desired	
SYRFL1R	STUP	4440 0441h	This enables the reception of Pdelay_Req, Pdelay_Resp, Pdelay_Resp_Follow_Up, Follow_Up, and Sync messages and the transfer of Announce messages to the PTPEDMAC.

28.3.11.1 Procedure for Starting Operations

Figure 28.20 shows the procedure for making the additional settings for shifting to operation as a slave during operation as a P2P (sending and receiving PTP-pdelay messages).

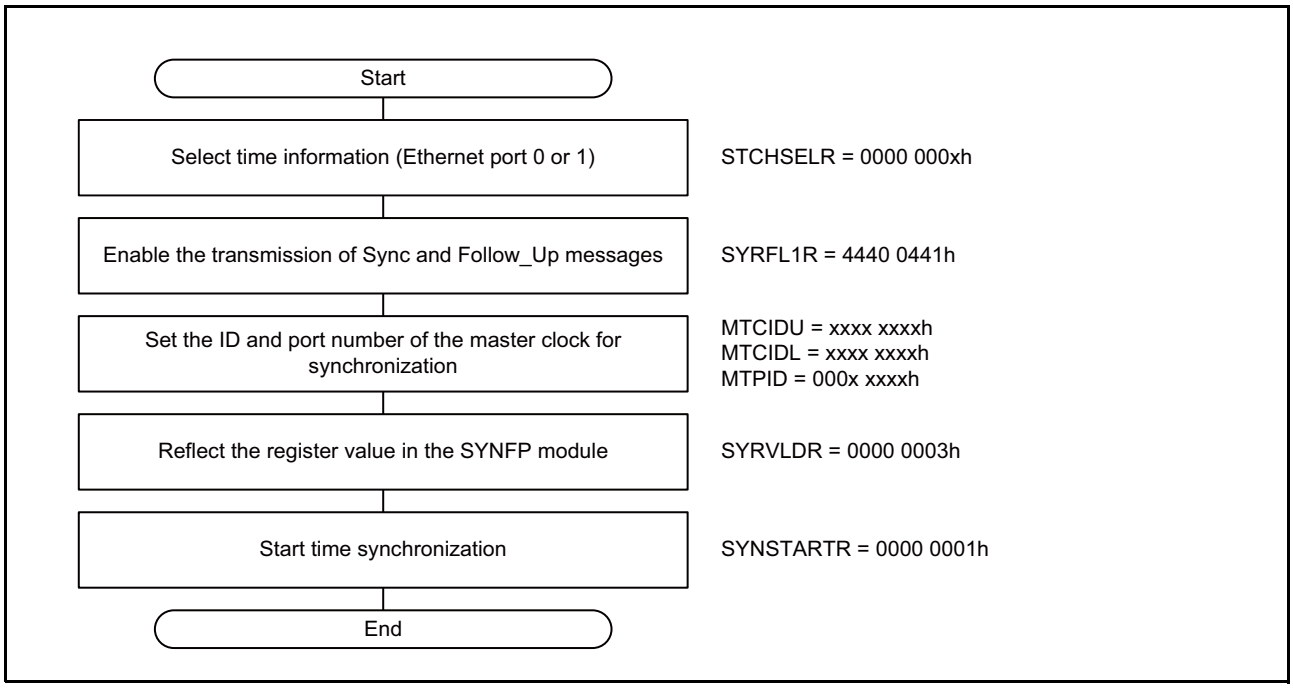


Figure 28.20 Procedure for Starting Operation as a P2P Slave

28.3.11.2 Procedure for Stopping Operations

Figure 28.21 shows the procedure for stopping the reception of Sync and Follow_Up messages to stop operation as a P2P slave.

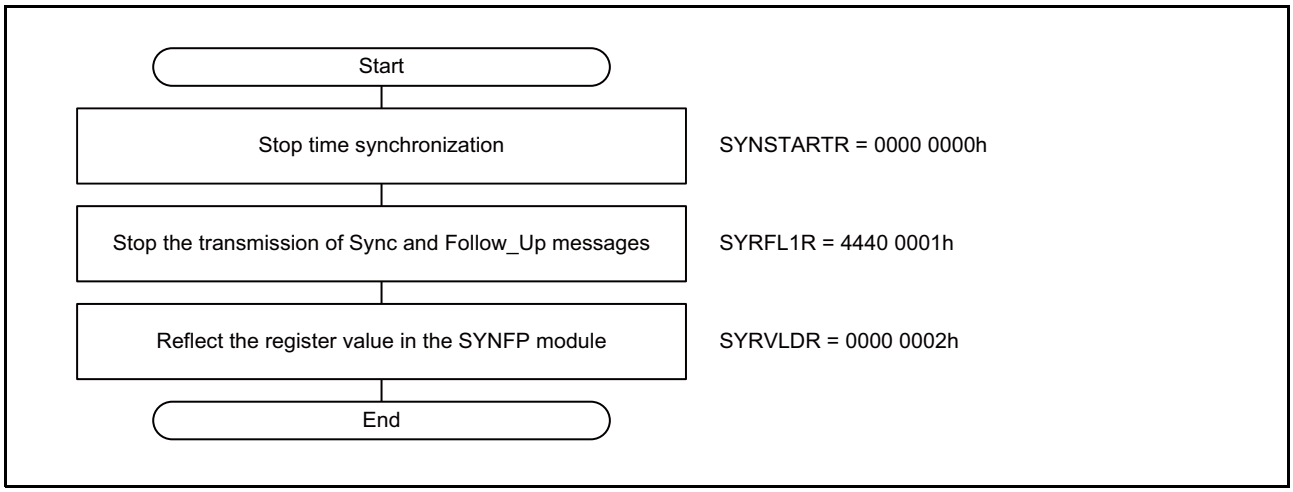


Figure 28.21 Procedure for Stopping Operation as a P2P Slave

28.3.12 Operation as an E2E TC

28.3.12.1 Preparatory Setting

Table 28.25 lists the registers for use in operation as an E2E TC.
Make the settings for both the SYNFP0 and SYNFP1 modules.

Table 28.25 Registers for Use in Operation as an E2E TC

Register Name	SYRVLDR Register Bits Used for Loading Direction	Settings	Description
SYCONFR	—	Example: 0000 0028h	Set the TCMOD bit to 0 as a preparatory setting for the PTP device.
SYRFL1R	STUP	2222 2222h	This enables the relaying of messages by the PRC-TC.
SYRFL2R	STUP	2000 0033h	This enables the transfer of Signaling and Management messages to the PTPEDMAC and the relaying of these messages by the PRC-TC.
SYTRENR	STUP	0000 0000h	No generation of messages

28.3.12.2 Procedure for Starting Operations

Figure 28.22 shows the procedure for starting operation as an E2E TC.

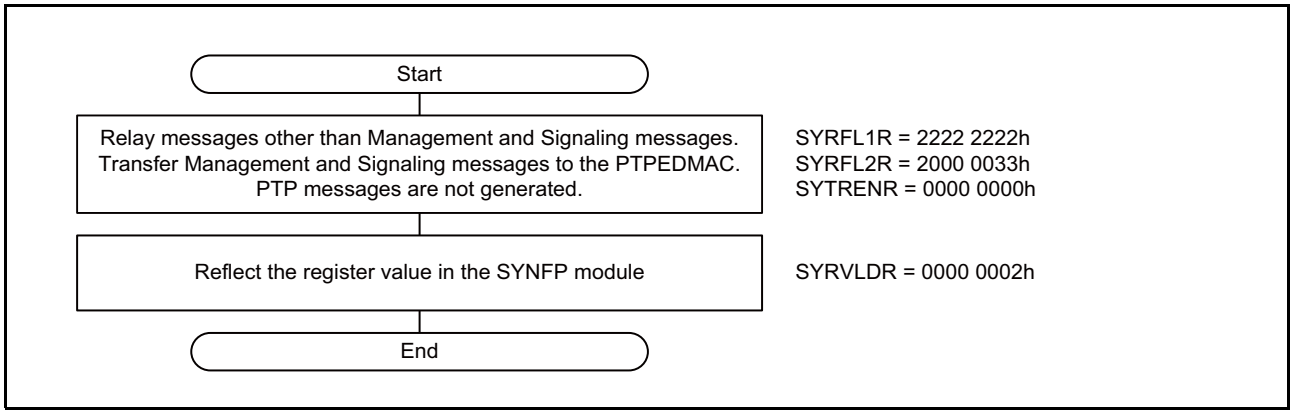


Figure 28.22 Procedure for Starting Operation as an E2E TC

28.3.13 Operation as a P2P TC

Table 28.26 lists the registers for use in operation a P2P TC.
Make the settings for both the SYNFP0 and SYNFP1 modules.

Table 28.26 Registers for Use in Operation as a P2P TC

Register Name	SYRVLDR Register Bits Used for Loading Direction	Settings	Description
SYCONFR	—	Example: 0010 0028h	Set the TCMOD bit to 1 as a preparatory setting for the PTP device.
RSTOUTR	STUP	As desired	
SYTLIR	STUP	Example: 0000 0000h	Pdelay_Req: 1 s
SYRFL1R	STUP	4440 0222h	This enables the discarding of Delay_Req and Delay_Resp messages and the relaying of Sync, Follow_Up, and Announce messages by the PRC-TC.
SYRFL2R	STUP	2000 0033h	This enables the transfer of Signaling and Management messages to the PTPEDMAC and the relaying of these messages by the PRC-TC
SYTRENR	STUP	0000 1000h	This enables the generation of Pdelay_Req messages.

28.3.13.1 Procedure for Starting Operations

Figure 28.23 shows the procedure for starting operation as a P2P TC.

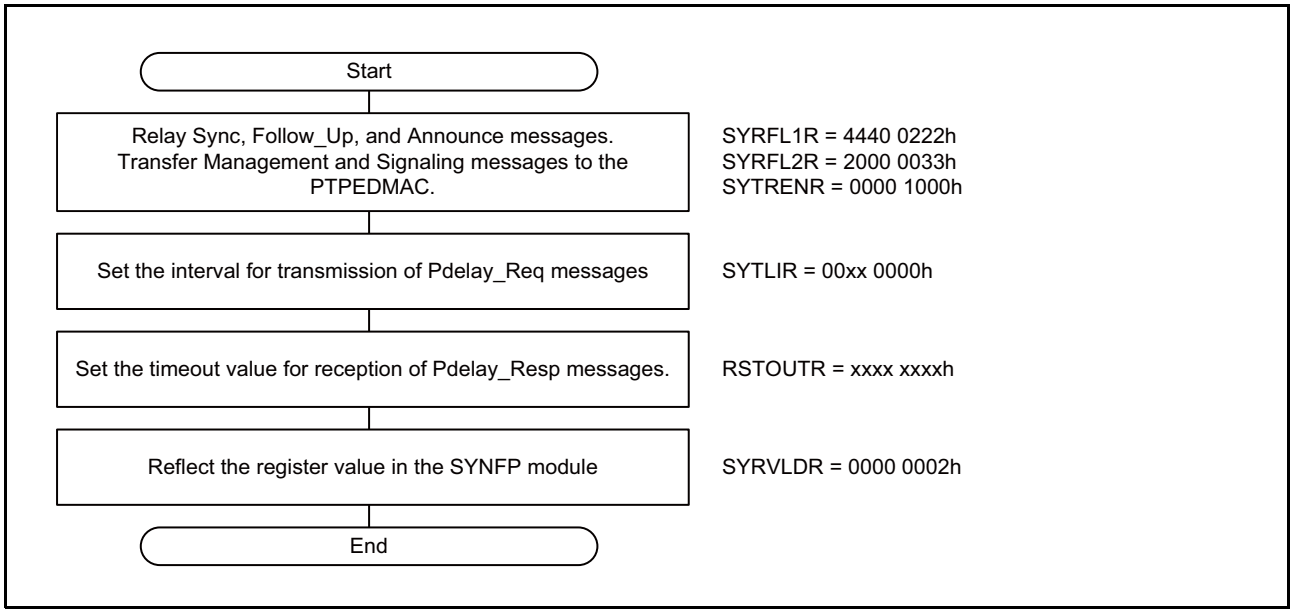


Figure 28.23 Procedure for Starting Operation as a P2P TC

28.3.14 Monitoring of Received Messages

28.3.14.1 Reception of Announce Messages

The EPTPC does not detect timeouts in the reception of Announce messages. To detect timeouts in the reception of these messages, monitor the reception of Announce messages by software.

28.3.14.2 Reception of Sync Messages

The STSR.SYNTOUT flag becomes 1 when a timeout in the reception of a Sync message occurs during the correction of time synchronization.
The SYSR.OFMUD flag becomes 1 when a Sync message is received, regardless of whether time synchronization is being corrected. Accordingly, the reception of Sync messages is detectable by reference to the SYSR.OFMUD flag even if the correction of time synchronization stops due to a timeout in the reception of a Sync message.

28.3.14.3 Reception of Delay_Resp and Pdelay_Resp Messages

The SYSR.DRPTO flag becomes 1 when a timeout in the reception of a Delay_Resp message occurs after the transmission of a Delay_Req message during operation as an E2E slave, or when a timeout in the reception of a Pdelay_Resp message occurs after the transmission of a Pdelay_Req message during operation as a P2P.
The SYSR.MPDUD flag becomes 1 when a Delay_Resp or Pdelay_Resp message is received, so the reception of these messages is still detectable in the case of a timeout in reception.

28.3.15 Correcting Time Synchronization

A slave detects differences in the clock gradient relative to the master clock. The offsetFromMaster values calculated by using the standard IEEE 1588 algorithm are used to calculate the clock gradient, so the result includes elements of network fluctuation besides frequency differences. The EPTPC has a “worst 10” function to eliminate fluctuations due to network load, etc. With these functions, the time is corrected from the calculated gradient difference values and results of correction are obtained as shown in Figure 28.25.

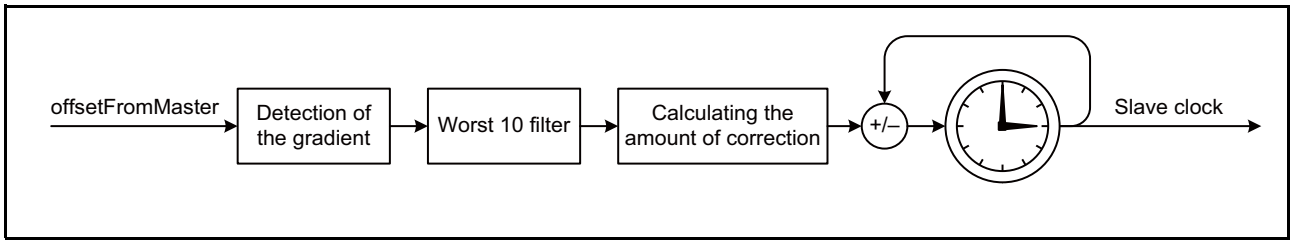


Figure 28.24 Configuration of the Time Correction Circuit

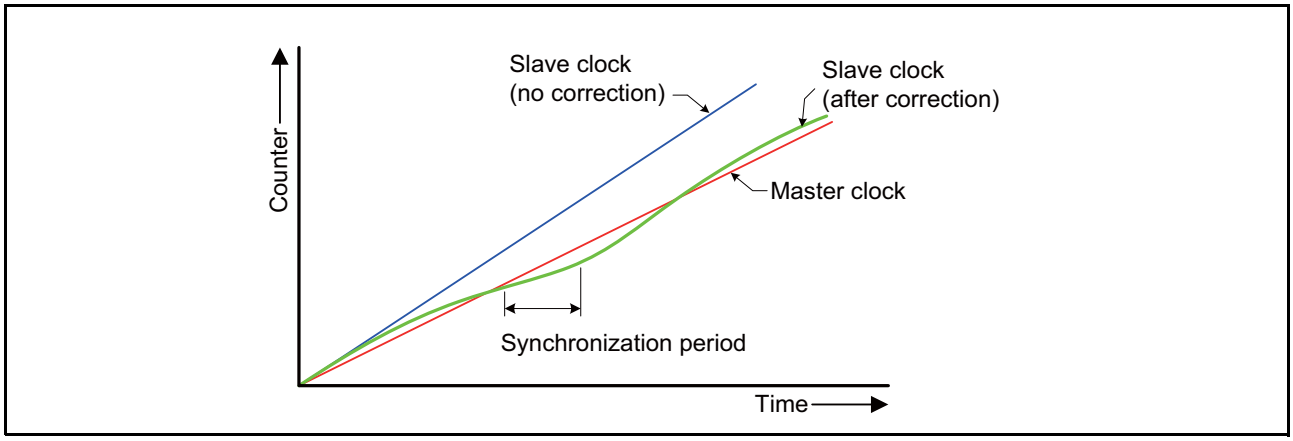


Figure 28.25 Overview of Time Correction

28.3.15.1 Judging Synchronization and Loss of Synchronization

The absolute value of offsetFromMaster reaching or exceeding the value specified in the SYNTDARU or SYNTDARL register is considered a loss of synchronization. The absolute value of offsetFromMaster being less than the absolute values of the synchronization detection threshold registers (SYNTDBRU and SYNTDBRL) is considered synchronization.

The values of the STSR.SYNCOUT and SYNC flags become 1 when synchronization is lost and obtained, respectively. Hysteresis can be obtained by setting the respective threshold registers to appropriate different values. Furthermore, the STMR.DVTH[3:0] and SYTH[3:0] bits can be used to set the consecutive number of times detection must occur for the judgment of loss of synchronization and of synchronization.

Set registers SYNTDARU and SYNTDARL to low values and the number of times detection is required to judge loss of synchronization to one in systems where control must be aborted if synchronization is lost due to fluctuations in network conditions. Set registers SYNTDARU and SYNTDARL and the number of times detection is required to large values in systems where the above condition does not apply.

Figure 28.26 shows an example of a situation where synchronization is being lost and regained. In this example, the number of consecutive times detection is required is three for both synchronization and loss of synchronization.

Note: The setting of the STSR.SYNCOUT flag is 1 when time synchronization starts, even though the condition for judging loss of synchronization has not been satisfied at this stage. For this reason, detection of loss of synchronization must be ignored immediately after time synchronization starts.

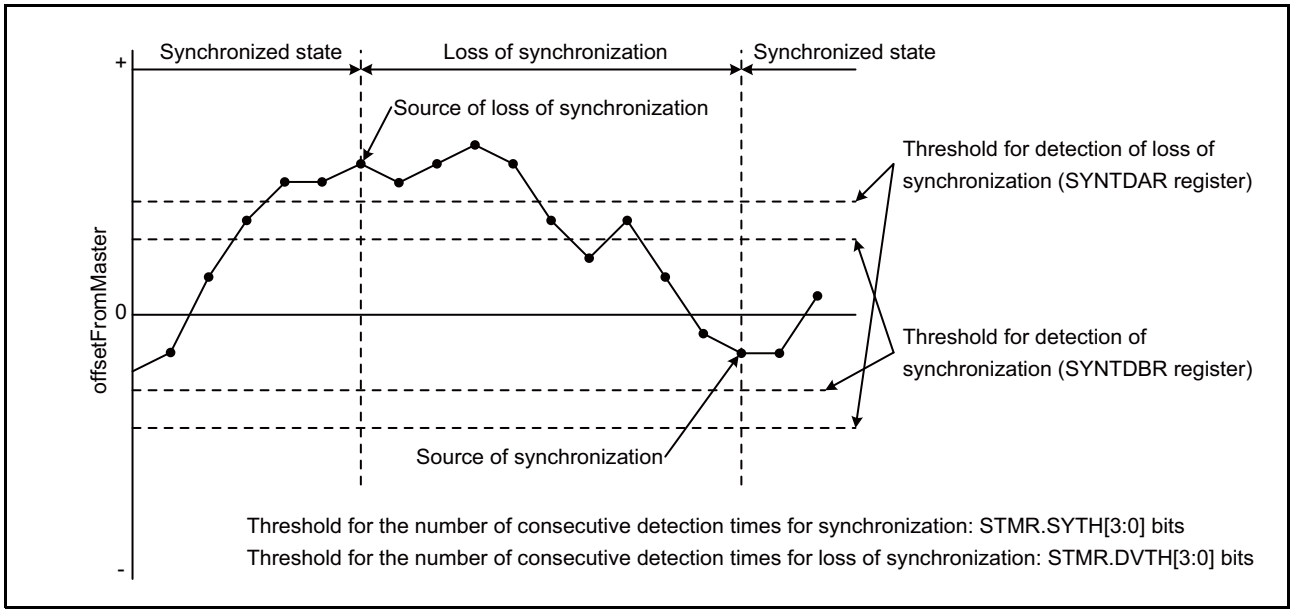


Figure 28.26 Example of a Situation where Synchronization is being Lost and Regained (the number of consecutive detection times is specified as three by the STMR.DVTH[3:0] and SYTH[3:0] bits)

28.3.15.2 Worst 10 Function

The worst 10 function is for imposing limits with regard to exceedingly large and small values among the calculated values for differences in clock gradient.

Values for difference in clock gradient are collected by observing the state of transfer over a specified interval and threshold values to impose limits are extracted from the observed values. With regard to differences in gradient, not only errors in the clock but also fluctuations in network conditions must be considered, and differences in both the positive and negative directions as shown in Figure 28.27 are collected

From among the collected values for positive and negative gradient difference, the largest values for gradient, from the first to the tenth place (worst to tenth worst), are acquired and the tenth worst is used as a threshold value. Fluctuations in the time kept by a slave clock can be suppressed by continually overwriting the tenth worst value for gradient difference with new values large enough to exceed the threshold. Furthermore, periodic collections of gradient values can also be made for updating the threshold values during operations or for the method of setting threshold values from previously measured results

However, note that while fluctuations in the time kept by a slave clock can be suppressed by the effective filtering of values for gradient difference (collecting the worst 10 values and using the tenth worst), following of the time kept by a master clock also becomes slower.

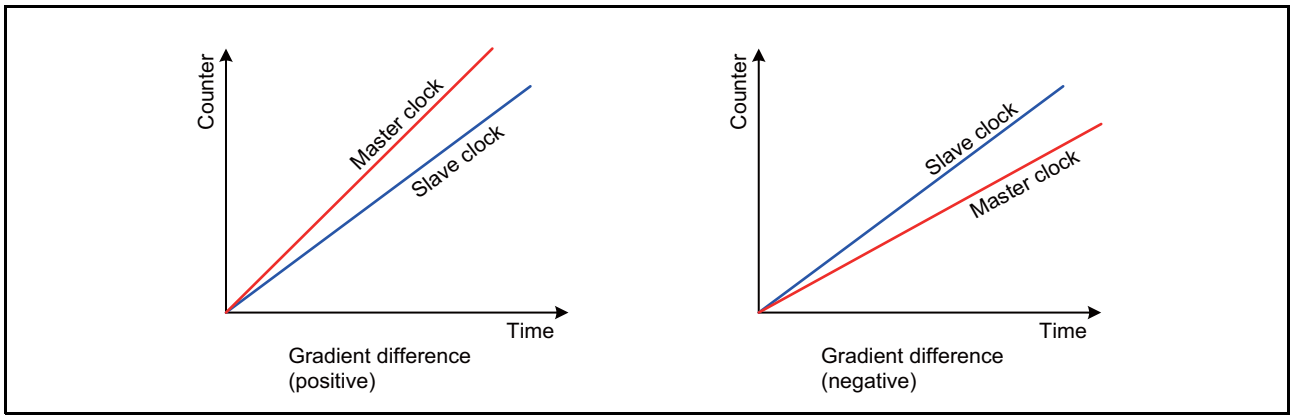


Figure 28.27 Overview of Gradient Differences

28.3.15.3 Collecting Differences in Clock Gradient and Extracting the Worst Ten Values

During slave operation, the EPTPC is capable of calculating the offsetFromMaster values from received messages and then calculating gradient differences between the local clock (acting as a slave clock) and master clock from those values. Specifically, the worst ten values are extracted from the sets of collected values for gradient difference. Either automatic filtering by the hardware or software-triggered filtering can be designated for acquisition of the sets of worst-10 values.

Figure 28.28 gives an overview of the collection of values for gradient difference.

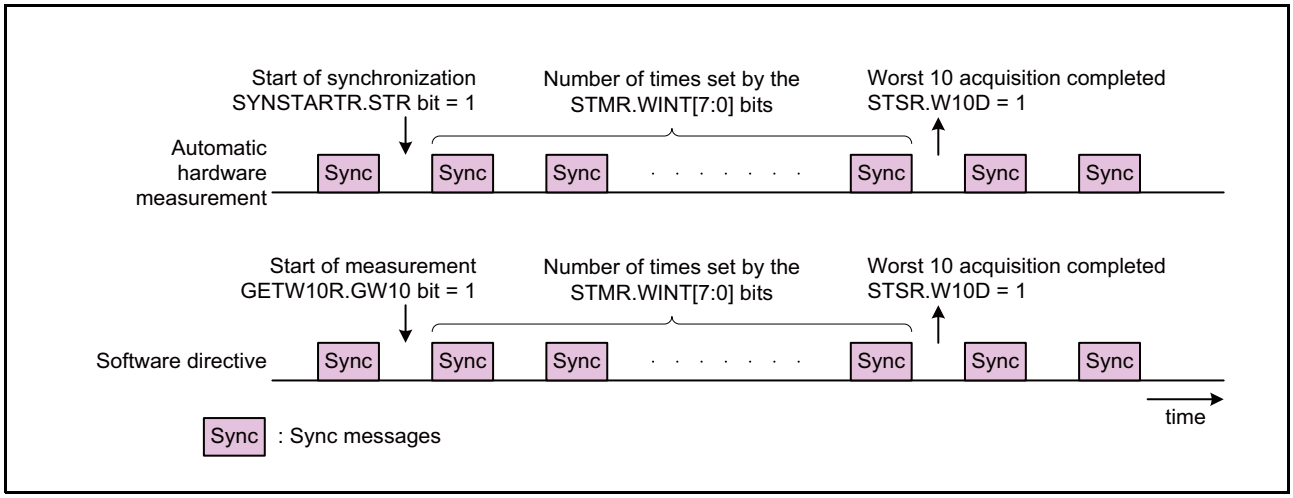


Figure 28.28 Overview of the Collection of Values for Gradient Differences

(1) Collecting Gradient Differences and Extracting the Worst Ten Values by Hardware

The EPTPC automatically collects the values of gradient difference by hardware if the STMR.W10S bit is 0.

When the SYNSTARTR.STR bit is set to 1 (starting slave time synchronization), the EPTPC collects gradient difference values for the number of times set by the STMR.WINT[7:0] bits. When the collection of gradient difference values is finished, the tenth largest values on the positive and negative gradient difference sides are stored as the tenth worst values in registers PW10VRU, PW10VRM, and PW10VRL and registers MW10RU, MW10RM, and MW10RL. When acquisition of the worst 10 values is complete, the STSR.W10D flag becomes 1.

Filtering of gradient difference values by using the stored tenth worst values then proceeds automatically.

Note that if the number of times set by STMR.WINT[7:0] bits are fewer than ten times, the double of the best of the collected values on the positive side is stored in registers PW10VRU, PW10VRM, and PW10VRL, and the half of the best of the collected values on the negative side is stored in registers MW10RU, MW10RM, and MW10RL.

(2) Collecting Gradient Differences and Extracting the Worst Ten Values by Software

The EPTPC collects values of gradient differences by software if the STMR.W10S bit is 1.
When the GETW10R.GW10 bit is set to 1 after time synchronization has started, the EPTPC collects gradient difference values for the number of times set by the STMR.WINT[7:0] bits. When the collection of gradient difference values is finished, the tenth largest values on the positive and negative gradient difference sides are stored as the tenth worst values in registers PW10VRU, PW10VRM, and PW10VRL and registers MW10RU, MW10RM, and MW10RL. When acquisition of the worst 10 values is complete, the STSR.W10D flag becomes 1.
Since filtering of gradient difference values proceeds with the values set in registers PLIMITRU, PLIMITRM, and PLIMITRL and registers MLIMITRU, MLIMITRM, and MLIMITRL as the upper and lower limits for filtering, the values stored in registers PW10VRU, PW10VRM, and PW10VRL and registers MW10RU, MW10RM, and MW10RL must be written to the PLIMITRU, PLIMITRM, and PLIMITRL and registers MLIMITRU, MLIMITRM, and MLIMITRL, respectively.
Note that if the number of times set by the STMR.WINT[7:0] bits is fewer than ten or times, the double of the best of the collected values on the positive is stored in registers PW10VRU, PW10VRM, and PW10VRL, and the half of the best of the collected values on the negative side is stored in registers MW10RU, MW10RM, and MW10RL.
The flowchart in Figure 28.29 shows an example of the procedure for software-triggered acquisition of the worst 10 values.

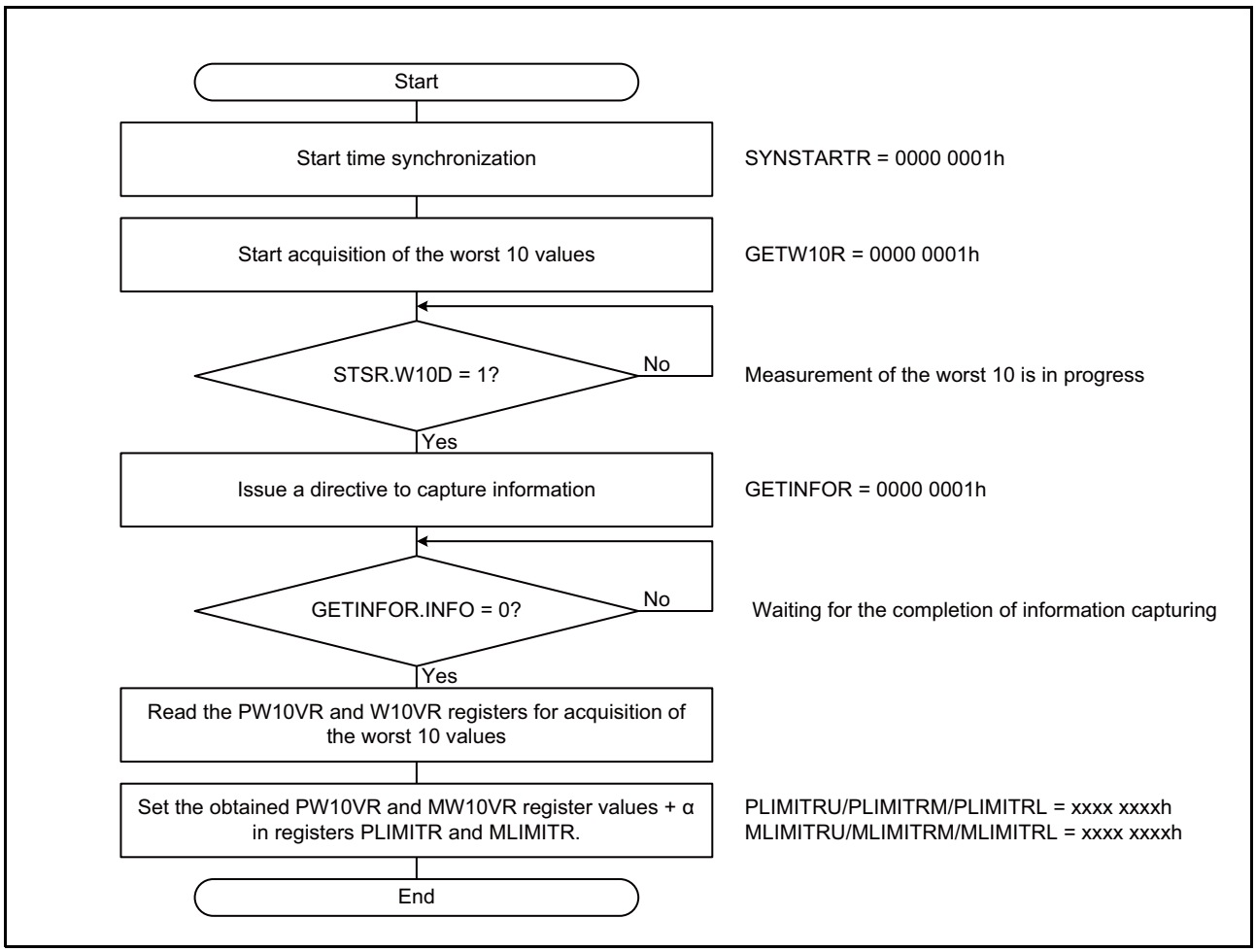


Figure 28.29 Example of the Procedure for Software-Triggered Acquisition of the Worst 10 Values

28.3.16 Local Clock Counter

The local clock counter retains the synchronized time information. The counter starts counting from 0 after the ETHERC is released from the module stop state or the EPTPC is released from the software reset state. The local clock counter can then be set to any desired value. The procedure for setting the initial value in the local clock counter is shown in Figure 28.30.

The time information kept by the local clock counter is also readable. Figure 28.31 shows the procedure for reading the time information kept by the local clock counter.

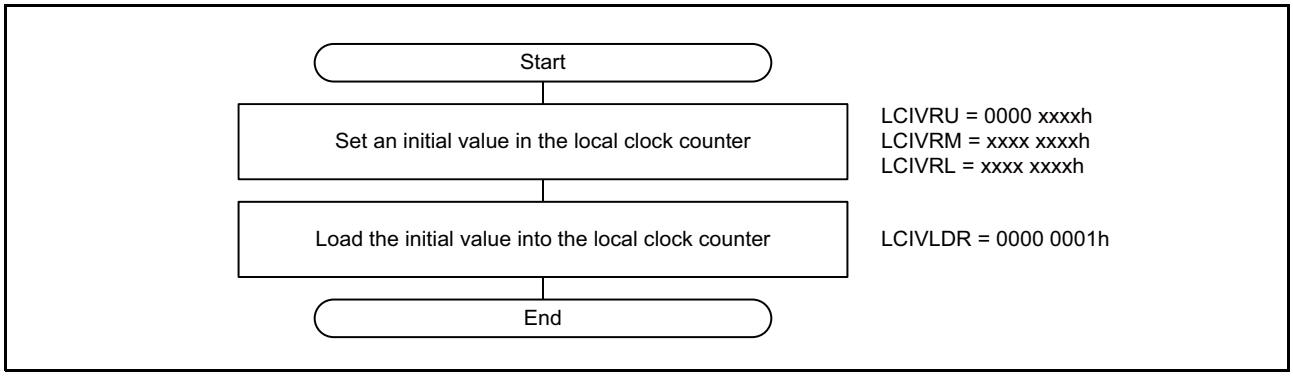


Figure 28.30 Procedure for Setting a New Initial Value in the Local Clock Counter

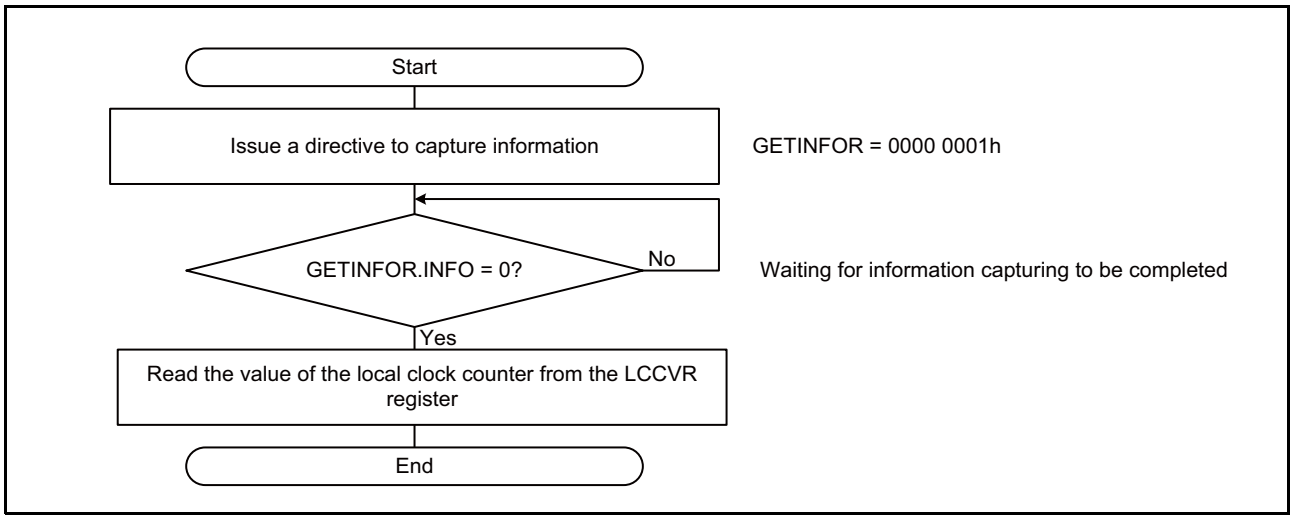


Figure 28.31 Procedure for Reading the Time Kept by the Local Clock Counter

28.3.17 Pulse Output Timer

The STCA module of the EPTPC incorporates six timers (pulse output timers 0 to 5) which operate independently of each other.

The pulse output timers produce periodic pulses, and the rising or falling edges of these pulses can be used as interrupt requests or output to the ELC as event signals. The time at which a pulse output timer starts operating (tstart), and the period (tc) and pulse width (tw) of the output pulses, can be specified.

The timing of pulse output timer operation is shown in Figure 28.32, and limits on the settings are listed in Table 28.27.

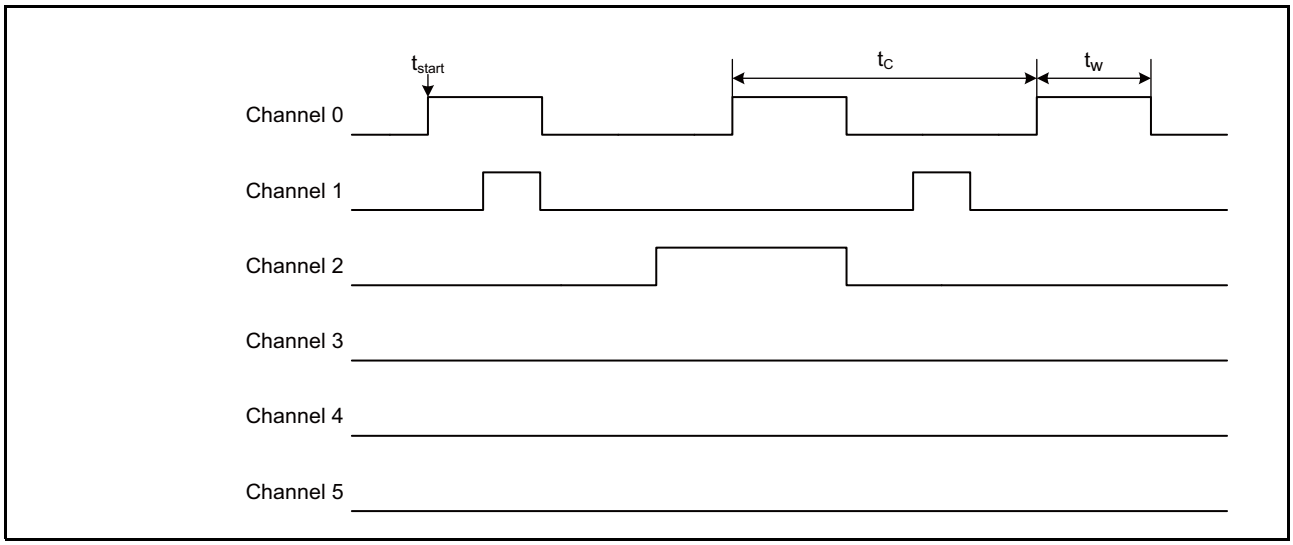


Figure 28.32 Time at which a Pulse Output Timer Starts Operating

Table 28.27 Limits on the Values that can be Specified for a Pulse Output Timer

Item	Description
Cycle (tC)	At least four cycles of the STCA clock and no greater than 1 s
Resolution of the cycle	Set in units of nanoseconds. However, the timing of rising edges is rounded by the period of the system clock (50 ns, 40 ns, or 20 ns).
Pulse width (tw)	At least two cycles of the STCA clock and no greater than 500 ms
Resolution of the pulse width	Set in units of nanoseconds. However, the timing of falling edges is rounded by the period of the system clock (50 ns, 40 ns, or 20 ns).

28.3.17.1 Procedure for Setting a Pulse Output Timer

Figure 28.33 shows the procedure for setting a pulse output timer.
Note that a timer does not produce periodic pulses if the time set in registers TMSTTRUm and TMSTTRLm (m = 0 to 5) has already passed. Set the time for a pulse output timer to start to a later time than the time when the timer is being set.

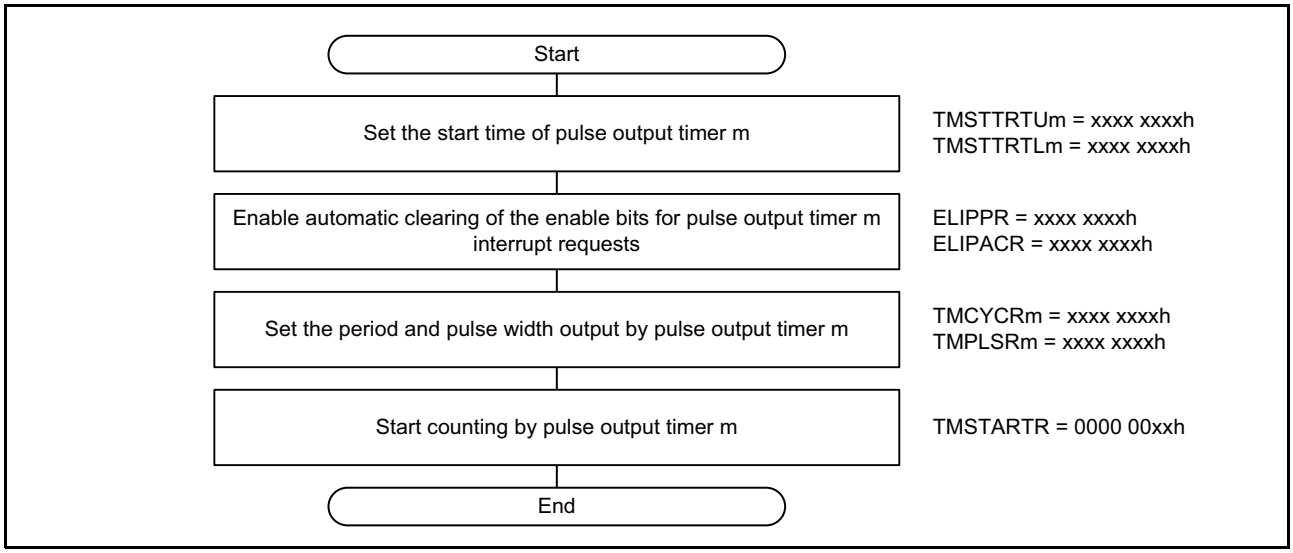


Figure 28.33 Procedure for Setting a Pulse Output Timer

28.3.17.2 Output of periodic pulses as interrupt requests or event signals

MINT interrupt requests, IPLS interrupt requests, or event output signals for the ELC can be generated by detecting rising or falling edges of the periodic pulses from the pulse output timer. The edge for detection and pulse output timer to use can be selected, and automatic clearing of enable bits for the IPLS interrupt or event output can be set. Make the required settings before setting the TMSTARTR.ENm bit to 1 (starting pulse output timer m).

(1) MINT Interrupt Request

Rising edges of the periodic pulses from the pulse output timers can be detected to generate MINT interrupt requests. Falling edges cannot be used in this way. The pulse output timers for use in generating MINT interrupt requests are selected by the MITSELR.MINTENm bits.
Automatic clearing of the enable bits for MINT interrupt requests is not available.

(2) IPLS Interrupt Request

Rising or falling edges of the periodic pulses from the pulse output timers can be detected to generate IPLS interrupt requests. The pulse output timers for use in generating IPLS interrupt requests are selected by the IPTSELR.IPTSELm bits.
Setting the ELIPACR.PLSP or PLSN bit enables automatic clearing of the enable bits for IPLS interrupt requests.

28.3.18 Priority Control in Transmission

28.3.18.1 Arbitration

In cases of contention between a message for transmission from the PTPEDMAC and a request for relaying from another channel, the PRC-TC module gives priority to the request for relaying from the other channel.
Cases of contention between multiple requests for the transmission of messages by the SYNFP module are also arbitrated in the order of priority shown in Table 28.28.

Table 28.28 Priority in Arbitration for Messages for Transmission

Messages for Transmission	Priority Order	Remark
Sync	1 Highest priority	
Delay_Req, Pdelay_Req	2	There is no device type that simultaneously transmits Delay_Req and Pdelay_Req messages.
Delay_Resp, Pdelay_Resp	3	There is no device type that simultaneously transmits Delay_Resp and Pdelay_Resp messages.
Announce	4	
Relay messages from another channel	5	The PRC-TC module determines the order of priority for these two items.
Messages for transmission from the PTPEDMAC	5	
Messages for transmission from the EDMAC	6	

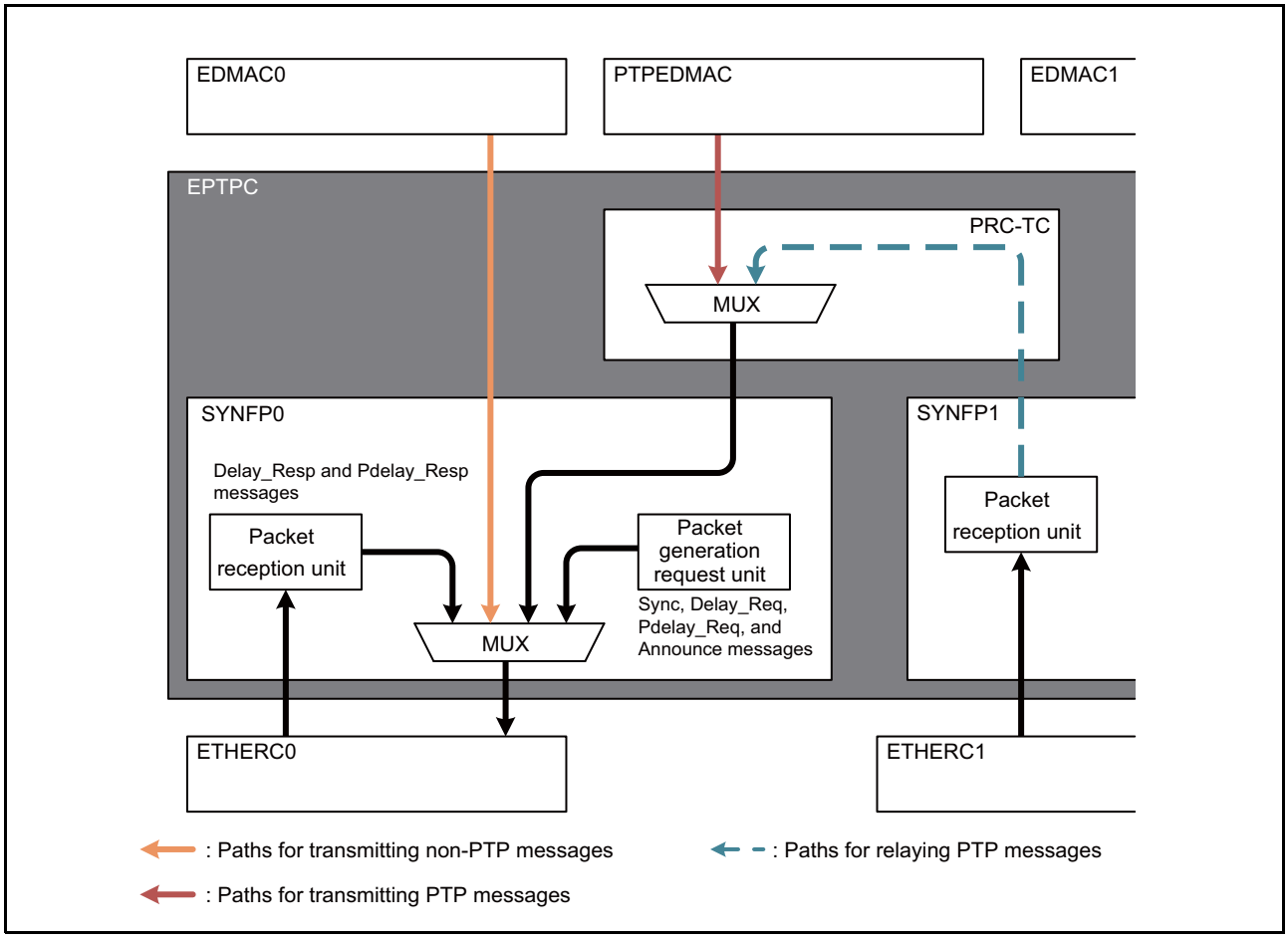


Figure 28.34 Arbitration in Message Transmission

28.3.18.2 Securing of Bandwidth for the Transmission of Sync Messages

The EPTPC secures bandwidth for the transmission of Sync messages, and is capable of handling transmission at very precise intervals.

If the transmission of a Sync message at a fixed interval proceeds at the same time as transmission by the PTPEDMAC or the relaying of messages by another channel is in progress, because transmission of the Sync message proceeds when the other processing is complete, the interval for transmission will no longer be fixed.

By securing bandwidth for the transmission of Sync messages, the transmission of messages from EDMAC0, EDMAC1, and the PTPEDMAC is limited, and the transmission of Sync messages can be handled without fluctuations. Securing of bandwidth for the transmission of Sync messages can be disabled by setting the SYCONFR.SBDIS bit to 1. Figure 28.35 gives a schematic view of securing bandwidth for the transmission of Sync messages.

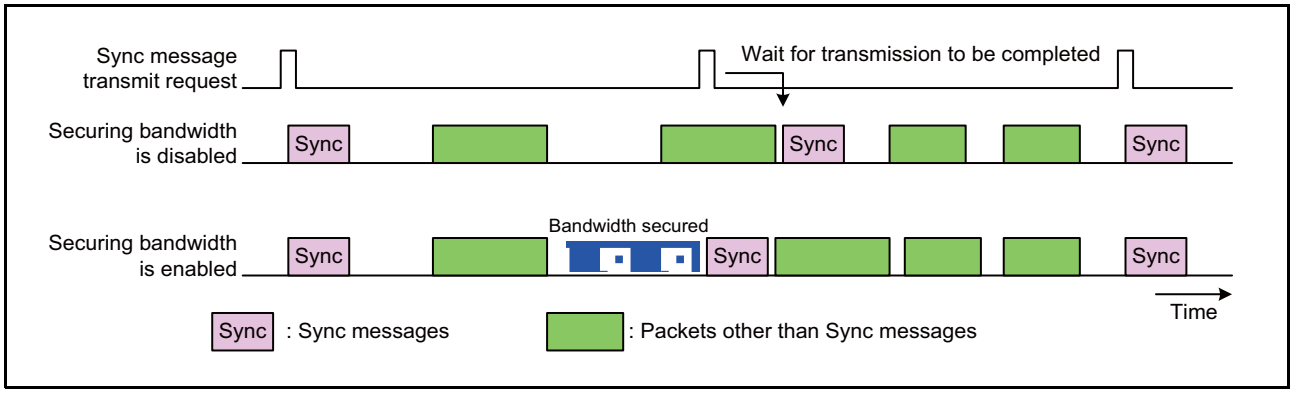


Figure 28.35 Securing of Bandwidth for the Transmission of Sync Messages

28.3.18.3 Securing of Transmission Interval

In the transmission of messages by the ETHERC, if there is a fixed delay from the time of a request for transmission to the time of transmission on the MII of Ethernet port 0 or 1, time stamp values of PTP messages can be used for accurately obtaining the size of the delay during operation as a slave. However, in the case of continuous transfer and so on, in situations where processing of messages to wait for interpacket gap times is required, delay times may fluctuate. Specifying an interval for frame transmission in SYCONFR.TCYC[7:0] bits to control the interval between the completion of transmission and the next request for transmission makes the ETHERC able to secure the reliability of time stamp values, thus avoiding the effects of interpacket gap times and obtaining a fixed delay for transmission.

28.4 Interrupts

The EPTPC has the MINT interrupt request and IPLS interrupt request.
Figure 28.36 shows the relation between the MINT interrupt request and IPLS interrupt request. Figure 28.37 shows the details on interrupt requests of the pulse output timer.

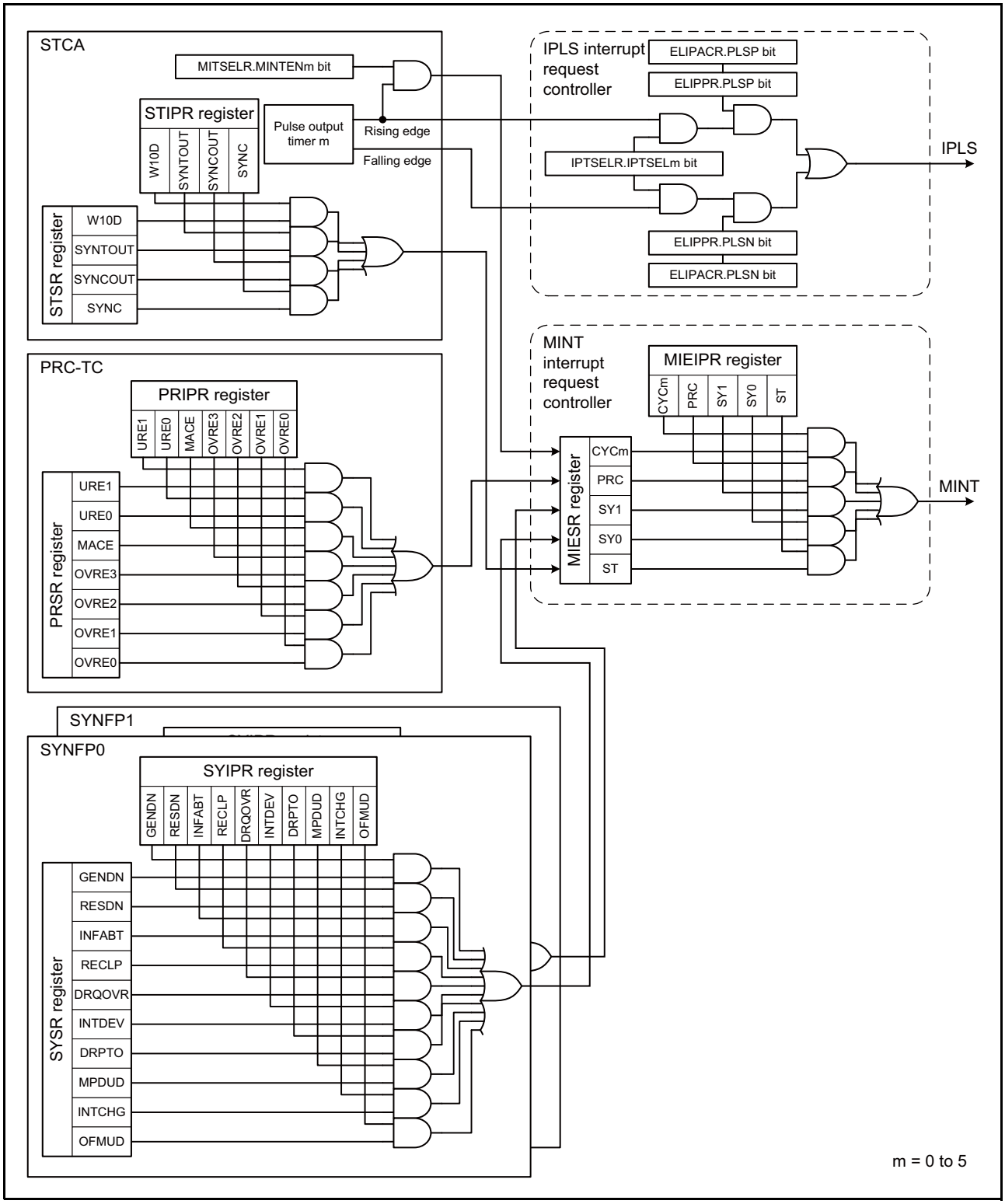


Figure 28.36 MINT Interrupt Request and IPLS Interrupt Request

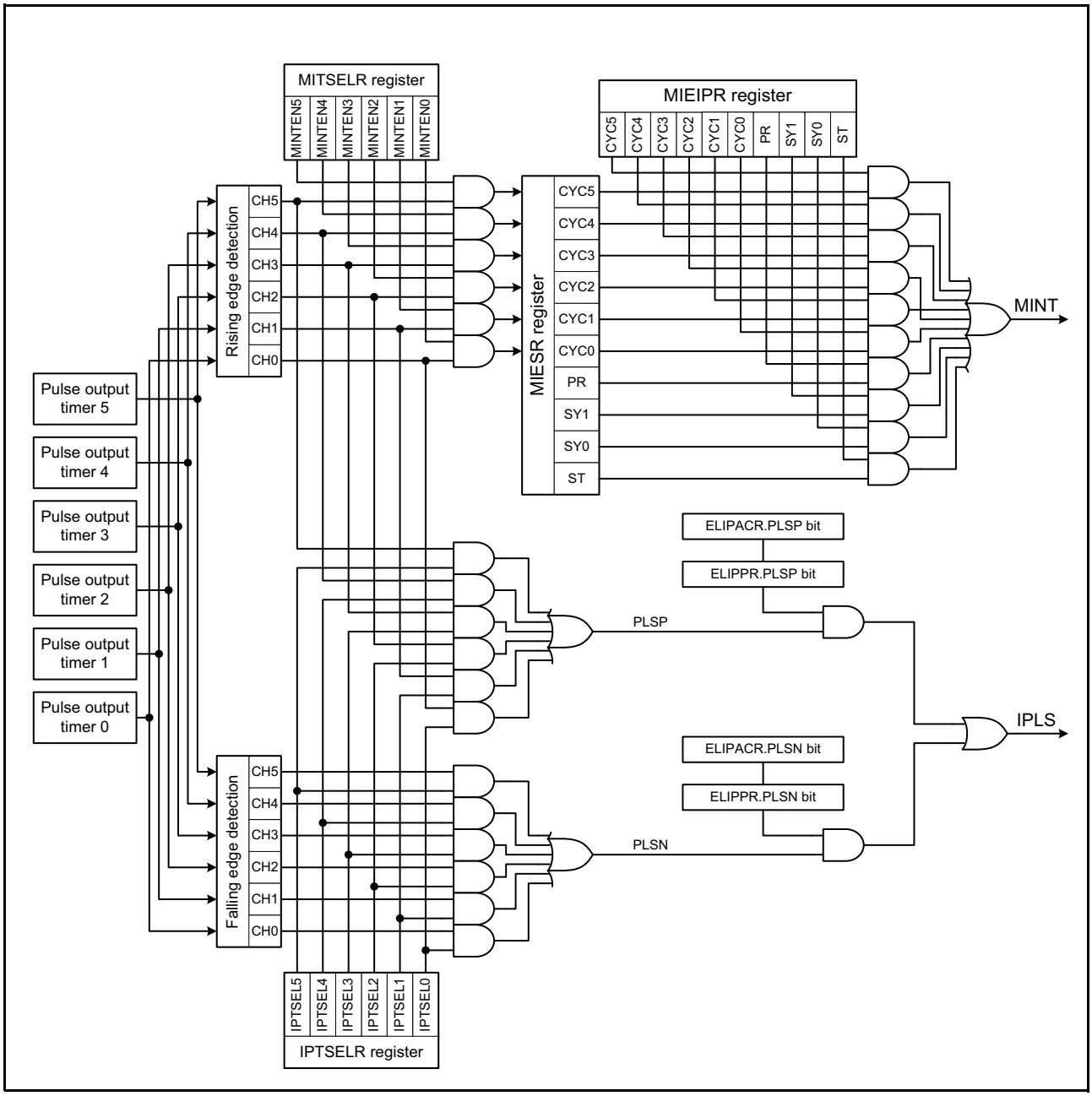


Figure 28.37 Details on Interrupt Requests of the Pulse Output Timer

28.5 Usage Notes

28.5.1 Setting of the Module-Stop Function

Operation of the EPTPC can be disabled or enabled with the standby control register 6 (STBCR6). The value following a reset is for the STBCR6 to stop. The registers of the EPTPC become accessible after 3 µs have elapsed following its release from the module-stop state and clearing of the BYPASS.BYPASSn bit to 0. For details on the module-stop function, see section 52, Power-Down Modes. Table 28.29 shows the relation between the settings of the respective bits and the accessibility of each of the EPTPC registers.
Access to the STCA and SYNFP units must proceed while the STCA and MII clock signals are being supplied to the respective units.

Table 28.29

STBCR6 Register				BYPASS Register		Allocation of Register Addresses for Access					
MSTP65 (ETHERC0, EDMAC0)	MSTP64 (ETHERC1, EDMAC1)	MSTP63 (EPTPC, PTPEDMAC)	MSTP62 (common circuit)	BYPASS0	BYPASS1	E820 4500h to E820 45FFh	E820 5000h to E820 503Fh	E820 5040h to E820 53FFh (STCA)	E820 5400h to E820 57FFh (PRC-TC)	E820 5800h to E820 5BFFh (SYNFP0)	E820 5C00h to E820 5FFFh (SYNFP1)
0	0	0	0	0	0	Accessible	Accessible	Accessible	Accessible	Accessible	Accessible
0	1	0	0	0	1	Accessible	Accessible	Accessible	Accessible	Accessible	Not accessible
1	0	0	0	1	0	Accessible	Accessible	Accessible	Accessible	Not accessible	Accessible
0	0	1	0	1	1	Accessible	Not accessible	Not accessible	Not accessible	Not accessible	Not accessible
0	1	1	0	1	1	Accessible	Not accessible	Not accessible	Not accessible	Not accessible	Not accessible
1	0	1	0	1	1	Accessible	Not accessible	Not accessible	Not accessible	Not accessible	Not accessible
1	1	1	1	1	1	Not accessible	Not accessible	Not accessible	Not accessible	Not accessible	Not accessible

Note: Settings other than those listed above are prohibited.

Writing does not proceed if you attempt access to the non-accessible registers or access while the MII and STCA clock signals are not being supplied. If you attempt reading, 0h is returned.

28.5.1.1 Release from the Module-Stop State

Before using the EPTPC, follow the steps below to release the EPTPC from the module-stop state. Do not proceed with other processing while this procedure is in progress.

(1) Clear the MSTP62, MSTP63, MSTP64, and MSTP65 bits in the STBCR6 register to 0.
(2) Clear the BYPASS0 and BYPASS1 bits in the BYPASS register to 0.
(3) Wait for 3µs.

Note: The procedure above is for cases where all functions are to be enabled. Set the respective bits as required.

28.5.1.2 Transition to the Module-Stop State

Follow the steps below to place the ETHERC, EPTPC, and EDMAC in the module-stop state. Do not proceed with other processing while this procedure is in progress.

(1) Reset the ETHERC, EPTPC, and EDMAC.*1
(2) Set the BYPASS0 and BYPASS1 bits in the BYPASS register to 1.
(3) Clear the MSTP62, MSTP63, MSTP64, and MSTP65 bits in the STBCR6 register to 1.
(4) Wait for 3 µs.

Note 1. See section 27.5.4, Points to Note when the EPTPC Is not in Use.

28.5.2 Notes on Placing the CPU on Software Standby

When you intend to place the CPU on software standby, start by placing the ETHERC, EPTPC, and EDMAC in the module stop state before executing the WFI instruction.

28.5.3 Wait Cycles for Register Access

Access to registers in the EPTPC involves the arbitration of different clock signals, specifically the peripheral module clock signal (Bφ), the STCA clock signal, and the MII clock signals such as TX_CLK. Accordingly, numbers of wait cycles for register access differ with the combination of the frequency settings of the peripheral module clock signal, of the STCA clock signal, and of the MII clock signals.
Table 28.30 gives examples of numbers of wait cycles for different combinations of clock frequency. Add 1 to 2 cycles to these values to obtain the number of access cycles.

Table 28.30 Wait Cycles for Register Access (when the frequency of the STCA clock is 50 MHz)

Address Range	STCA Clock = 50MHz							
	Peripheral Module Clock Bφ = 132 MHz				Peripheral Module Clock Bφ = 33 MHz			
	MII Clock 25 MHz (100 Mbps)		MII Clock 2.5 MHz (10 Mbps)		MII Clock 25 MHz (100 Mbps)		MII Clock 2.5 MHz (10 Mbps)	
	Read	Write	Read	Write	Read	Write	Read	Write
E820 4500h to E820 45FFh	2	2	2	2	2	2	2	2
E820 5000h to E820 503Fh	4	4	4	4	4	4	4	4
E820 5040h to E820 53FFh (STCA)	7	19 to 25*1	7	19 to 25*1	7	15 to 17*1	7	15 to 17*1
E820 5400h to E820 57FFh (PRC-TC)	8	8	8	8	8	8	8	8
E820 5800h to E820 5BFFh (SYNFP0)	8	25 to 37*2	8	119 to 225*2	8	17 to 21*2	8	41 to 69*2
E820 5C00h to E820 5FFFh (SYNFP1)	8	23 to 33*2	8	111 to 225*2	8	17 to 21*2	8	41 to 69*2

Note 1. The number of wait cycles in access to the STCA-related registers (Wstca) can be calculated to be in the following range from the periods of the peripheral module clock (tc(Bφ)) and STCA clock (tc(STCA)).
The minimum value of Wstca = Int (tc(STCA)/tc(Bφ)) × 2 + 15 (tc(Bφ) ≤ tc(STCA))
= 15 (tc(Bφ) > tc(STCA))
The maximum value of Wstca = Int (tc(STCA)/tc(Bφ)) × 4 + 17 (tc(Bφ) ≤ tc(STCA))
= 17 (tc(Bφ) > tc(STCA))
• Int(A) is the calculation of the largest integer not greater than A.
• This calculation presumes that the CPU clock and peripheral module clock have the same periods.

For example, if the frequency of the peripheral module clock is 132 MHz and that of the STCA clock is 50 MHz,
the minimum value of Wstca = Int (50 [ns]/7.58 [ns]) × 2 + 15 = 19, and
the maximum value of Wstca = Int (50 [ns]/7.58 [ns]) × 4 + 17 = 25.

Note 2. The number of wait cycles in access to the SYNFP-related registers (Wsynf) can be calculated to be in the following range from the periods of the peripheral module clock (tc(Bφ)) and MII clock (tc(MII)).
The minimum value of Wsynf = Int (tc(MII)/tc(Bφ)) × 2 + 15 (tc(Bφ) ≤ tc(MII))
= 15 (tc(Bφ) > tc(MII))
The maximum value of Wsynf = Int (tc(MII)/tc(Bφ)) × 4 + 17 (tc(Bφ) ≤ tc(MII))
= 17 (tc(Bφ) > tc(MII))
• Int(A) is the calculation of the largest integer not greater than A.
• This calculation presumes that the CPU clock and peripheral module clock have the same periods.

For example, if the frequency of the peripheral module clock is 132 MHz and the transmission rate is 10 Mbps (so the MII clock is running at 2.5 MHz),
the minimum value of Wsynf= Int (400 [ns]/7.58 [ns]) × 2 + 15 = 119, and
the maximum value of Wsynf= Int (400 [ns]/7.58 [ns]) × 4 + 17 = 225.

28.5.4 Restriction on the EPTPCa Operation when the MII Interface is in Use

The ETn_RX_CLK signal of the MII can be used for the STCA clock, however, it must be fixed to 25 MHz (100 Mbps) when in use. Use the ETn_SCLKIN signal instead if you may use the ETHERC at 10 Mbps when the MII is in use.

28.5.5 Transfer by PTPEDMAC when the Transparent Clock (TC) is in Use

When the setting of any of the following bits is 1, indicating the processing of a PTP event message, do not use PTPEDMAC to transfer data for transmission. Instead, use EDMACn of a corresponding channel for transfer (n = 0, 1).

- SYRFL1R.PDRP[1], SYRFL1R.PDRQ[1], SYRFL1R.DRP[1],or SYRFL1R.SYNC[1]

29. DMA Controller for the Ethernet Controller (EDMACa)

29.1 Overview

This MCU has three channels for the ethernet controller direct memory access controller (EDMAC): two channels for the ethernet controller (ETHERC) and one channel for the PTP controller (EPTPC). EDMAC0 and EDMAC1 control data transmission and reception for ETHERC0 and ETHERC1, respectively. The PTPEDMAC controls data transmission and reception for ETHERC0 and ETHERC1 according to the EPTPC settings. In this section, channel numbers are indicated as n (n = 0, 1).

The EDMAC controls most of the transmit/receive buffer management for communications. This reduces the load on the CPU and allows efficient data transmission and reception. The data transfers are controlled according to the information (referred to as descriptors) on the memory.

Table 29.1 lists the EDMAC specifications. Figure 29.1 shows the EDMAC configuration, and Figure 29.2 shows the configuration of descriptors and transmit/receive buffers on the memory.

Table 29.1 EDMAC Specifications

Item	Description
Data transmission and reception	• Controls data transmission and reception according to descriptors • Supports single buffer frame transmission and reception (1 buffer per frame), and multi-buffer frame transmission and reception (multiple buffers per frame)
Functions	• Minimizes system bus occupation time using block transfer (32-byte units) • Writes back the transmit/receive frame state to descriptors • Inserts padding in receive data
Low power consumption function	The EDMAC can be set to the module-stop state to reduce power consumption.

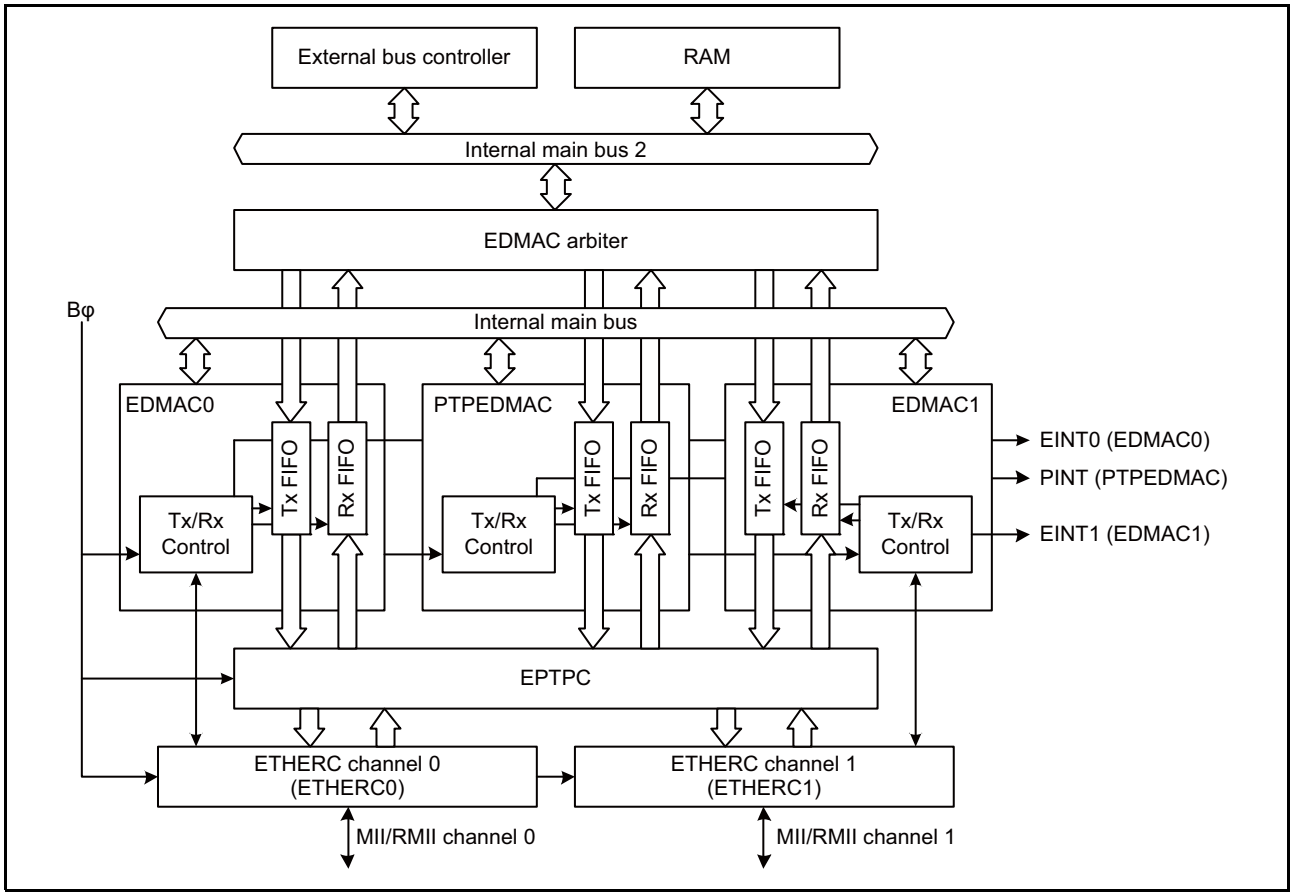


Figure 29.1 EDMAC Configuration

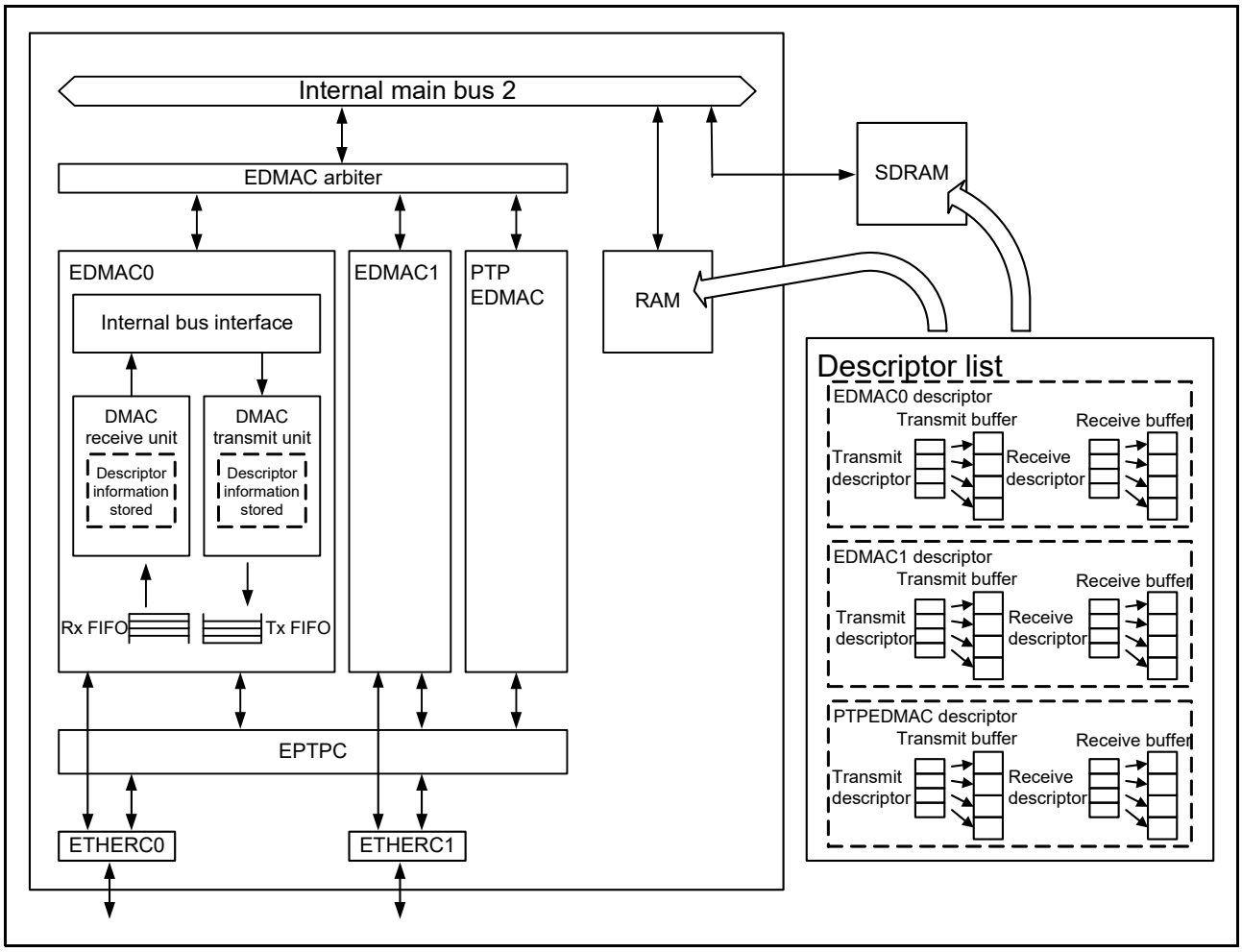


Figure 29.2 Configuration of Descriptors and Transmit/Receive Buffers on the Memory

29.2 Register Descriptions

Table 29.2 shows the register configuration.

Table 29.2 Register Configuration

Module	Register Name	Abbreviation	Address	Access size
EDMAC0	EDMAC mode register	EDMR	E820 4000h	32
	EDMAC transmit request register	EDTRR	E820 4008h	32
	EDMAC receive request register	EDRRR	E820 4010h	32
	Transmit descriptor list start address register	TDLAR	E820 4018h	32
	Receive descriptor list start address register	RDLAR	E820 4020h	32
	ETHERC/EDMAC status register	EESR	E820 4028h	32
	ETHERC/EDMAC status interrupt enable register	EESIPR	E820 4030h	32
	ETHERC/EDMAC transmit/receive status copy enable register	TRSCER	E820 4038h	32
	Missed-frame counter register	RMFCR	E820 4040h	32
	Transmit FIFO threshold register	TFTR	E820 4048h	32
	FIFO depth register	FDR	E820 4050h	32
	Receive method control register	RMCR	E820 4058h	32
	Transmit FIFO underflow counter	TFUCR	E820 4064h	32
	Receive FIFO overflow counter	RFOCR	E820 4068h	32
	Independent output signal setting register	IOSR	E820 406Ch	32
	Flow control start FIFO threshold setting register	FCFTR	E820 4070h	32
	Receive data padding insert register	RPADIR	E820 4078h	32
	Transmit interrupt setting register	TRIMD	E820 407Ch	32
	Receive buffer write address register	RBWAR	E820 40C8h	32
	Receive descriptor fetch address register	RDFAR	E820 40CCh	32
	Transmit buffer read address register	TBRAR	E820 40D4h	32
	Transmit descriptor fetch address register	TDFAR	E820 40D8h	32
EDMAC1	EDMAC mode register	EDMR	E820 4200h	32
	EDMAC transmit request register	EDTRR	E820 4208h	32
	EDMAC receive request register	EDRRR	E820 4210h	32
	Transmit descriptor list start address register	TDLAR	E820 4218h	32
	Receive descriptor list start address register	RDLAR	E820 4220h	32
	ETHERC/EDMAC status register	EESR	E820 4228h	32
	ETHERC/EDMAC status interrupt enable register	EESIPR	E820 4230h	32
	ETHERC/EDMAC transmit/receive status copy enable register	TRSCER	E820 4238h	32
	Missed-frame counter register	RMFCR	E820 4240h	32
	Transmit FIFO threshold register	TFTR	E820 4248h	32
	FIFO depth register	FDR	E820 4250h	32
	Receive method control register	RMCR	E820 4258h	32
	Transmit FIFO underflow counter	TFUCR	E820 4264h	32
	Receive FIFO overflow counter	RFOCR	E820 4268h	32
	Independent output signal setting register	IOSR	E820 426Ch	32
	Flow control start FIFO threshold setting register	FCFTR	E820 4270h	32
	Receive data padding insert register	RPADIR	E820 4278h	32
	Transmit interrupt setting register	TRIMD	E820 427Ch	32
	Receive buffer write address register	RBWAR	E820 42C8h	32
	Receive descriptor fetch address register	RDFAR	E820 42CCh	32

Table 29.2 Register Configuration

Module	Register Name	Abbreviation	Address	Access size
EDMAC1	Transmit buffer read address register	TBRAR	E820 42D4h	32
	Transmit descriptor fetch address register	TDFAR	E820 42D8h	32
PTPEDMAC	EDMAC mode register	EDMR	E820 4400h	32
	EDMAC transmit request register	EDTRR	E820 4408h	32
	EDMAC receive request register	EDRRR	E820 4410h	32
	Transmit descriptor list start address register	TDLAR	E820 4418h	32
	Receive descriptor list start address register	RDLAR	E820 4420h	32
	PTP/EDMAC status register	EESR	E820 4428h	32
	PTP/EDMAC status interrupt enable register	EESIPR	E820 4430h	32
	Missed-frame counter register	RMFCR	E820 4440h	32
	Transmit FIFO threshold register	TFTR	E820 4448h	32
	FIFO depth register	FDR	E820 4450h	32
	Receive method control register	RMCR	E820 4458h	32
	Transmit FIFO underflow counter	TFUCR	E820 4464h	32
	Receive FIFO overflow counter	RFOCR	E820 4468h	32
	Flow control start FIFO threshold setting register	FCFTR	E820 4470h	32
	Receive data padding insert register	RPADIR	E820 4478h	32
	Transmit interrupt setting register	TRIMD	E820 447Ch	32
	Receive buffer write address register	RBWAR	E820 44C8h	32
	Receive descriptor fetch address register	RDFAR	E820 44CCh	32
	Transmit buffer read address register	TBRAR	E820 44D4h	32
	Transmit descriptor fetch address register	TDFAR	E820 44D8h	32

29.2.1 EDMAC Mode Register (EDMR)

Address: EDMAC0.EDMR E820 4000h, EDMAC1.EDMR E820 4200h, PTPEDMAC.EDMR E820 4400h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	DE	DL[1:0]		—	—	—	SWR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SWR	Software Reset	When 1 is written, the corresponding channels of the EDMAC and ETHERC are reset. Note that for the PTPEDMAC, the ETHERC are not reset. Registers TDLAR, RDLAR, RMFCR, TFUCR, and RFOCR are not reset with this bit. The read value is 0.	R/W
b3 to b1	—	Reserved	The read value is 0. The write value should be 0.	R/W
b5, b4	DL[1:0]	Transmit/Receive Descriptor Length	b5 b4 0 0: 16 bytes 0 1: 32 bytes 1 0: 64 bytes 1 1: 16 bytes	R/W
b6	DE	Big Endian Mode/Little Endian Mode *1	0: Big endian mode 1: Little endian mode	R/W
b31 to b7	—	Reserved	The read value is 0. The write value should be 0.	R/W

Note 1. This setting applies to data for the transmit/receive buffer. It does not apply to transmit/receive descriptors and registers.

The EDMR register controls EDMAC operation.
Set the EDMR register during the initialization process after a reset. When rewriting this register outside of the initialization process, set the SWR bit to 1 to reset the EDMAC and ETHERC, and then set this register again. If the ETHERC and EDMAC are reset during data transmission or reception, abnormal data may be sent on the line. Do not rewrite this register while the ETHERC transmit or receive function is enabled. It takes 64 cycles of the peripheral module clock (Bφ) to initialize the ETHERC and EDMAC. Complete the initialization process before accessing registers in the ETHERC and EDMAC.

29.2.2 EDMAC Transmit Request Register (EDTRR)

Address: EDMAC0.EDTRR E820 4008h, EDMAC1.EDTRR E820 4208h, PTPEDMAC.EDTRR E820 4408h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TR	Transmit Request	When 1 is written, the EDMAC reads the corresponding descriptor and transmits frames where the TD0.TACT bit is 1. The TR bit becomes 0 after all the valid frames are transmitted. Writing 0 to this bit has no effect.	R/W
b31 to b1	—	Reserved	The read value is 0. The write value should be 0.	R/W

The EDTRR register controls EDMAC transmission.

After the EDMAC has transmitted one frame, it reads the next descriptor. When the TD0.TACT bit in the descriptor is 1, the EDMAC continues transmission. When the TD0.TACT bit is 0, the EDMAC sets the TR bit to 0 and stops transmission.

29.2.3 EDMAC Receive Request Register (EDRRR)

Address: EDMAC0.EDRRR E820 4010h, EDMAC1.EDRRR E820 4210h, PTPEDMAC.EDRRR E820 4410h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	RR	Receive Request	0: Receive function is disabled.*1 1: Receive descriptor is read, and the receive function is enabled.	R/W
b31 to b1	—	Reserved	The read value is 0. The write value should be 0.	R/W

Note 1. If the receive function is disabled during frame reception, write-back to the receive descriptor is not performed successfully. Subsequent pointers for reading a receive descriptor become abnormal and the EDMAC cannot operate normally. In this case, to enable the EDMAC receive function again, execute a software reset by setting the EDMR.SWR bit to 1.
To disable the EDMAC receive function without resetting the EDMAC, set the ETHERCn.ECMR.RE bit to 0. Next, after the EDMAC has completed reception and write-back to the receive descriptor has been confirmed, set the RR bit to 0.

The EDRRR register controls EDMAC reception.

When the RR bit becomes 1, the EDMAC reads the receive descriptor. When the RD0.RACT bit is 1, the EDMAC waits for a receive request from the ETHERC.

When the EDMAC has received data for the receive buffer size, it reads the next descriptor and waits to receive a frame. If the RD0.RACT bit is 0, the EDMAC sets the RR bit to 0 and stops reception.

29.2.4 Transmit Descriptor List Start Address Register (TDLAR)

Address: EDMAC0.TDLAR E820 4018h, EDMAC1.TDLAR E820 4218h, PTPEDMAC.TDLAR E820 4418h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	The start address of the receive descriptor list is set. Set the start address according to the descriptor length selected by the EDMR.DL[1:0] bits. 16-byte boundary: Lower 4 bits = 0000b 32-byte boundary: Lower 5 bits = 00000b 64-byte boundary: Lower 6 bits = 000000b	R/W

The TDLAR register sets the start address of the transmit descriptor list.
Align each descriptor on the corresponding boundary to the descriptor length selected by the EDMR.DL[1:0] bits.
Do not rewrite the TDLAR register during transmission. Rewrite the TDLAR register while the EDTRR.TR bit is 0.

29.2.5 Receive Descriptor List Start Address Register (RDLAR)

Address: EDMAC0.RDLAR E820 4020h, EDMAC1.RDLAR E820 4220h, PTPEDMAC.RDLAR E820 4420h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	—	—	The start address of the receive descriptor list is set. Set the start address according to the descriptor length selected by the EDMR.DL[1:0] bits. 16-byte boundary: Lower 4 bits = 0000b 32-byte boundary: Lower 5 bits = 00000b 64-byte boundary: Lower 6 bits = 000000b	R/W

The RDLAR register sets the start address of the receive descriptor list.
Allocate each descriptor on the corresponding boundary to the descriptor length selected by the EDMR.DL[1:0] bits.
Do not rewrite the RDLAR register during reception. Rewrite the RDLAR register while the EDRRR.RR bit is 0.

29.2.6 ETHERC/EDMAC Status Register (EDMACn.EESR)

Address: EDMAC0.EESR E820 4028h, EDMAC1.EESR E820 4228h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	TWB	—	—	—	TABT	RABT	RFCOF	—	ECI	TC	TDE	TFUF	FR	RDE	RFOF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	CND	DLC	CD	TRO	RMAF	—	—	RRF	RTLF	RTSF	PRE	CERF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CERF	CRC Error Flag	0: CRC error has not been detected. 1: CRC error has been detected.	R/W
b1	PRE	PHY-LSI Receive Error Flag	0: PHY-LSI receive error has not been detected. 1: PHY-LSI receive error has been detected.	R/W
b2	RTSF	Frame-Too-Short Error Flag	0: Frame-too-short error has not been received. 1: Frame-too-short error has been received.	R/W
b3	RTLF	Frame-Too-Long Error Flag	0: Frame-too-long error has not been received. 1: Frame-too-long error has been received.	R/W
b4	RRF	Alignment Error Flag	0: Alignment error has not been received. 1: Alignment error has been received.	R/W
b6, b5	—	Reserved	The read value is 0. The write value should be 0.	R/W
b7	RMAF	Multicast Address Frame Receive Flag	0: Multicast address frame has not been received. 1: Multicast address frame has been received.	R/W
b8	TRO	Transmit Retry Over Flag	0: Transmit retry-over condition has not been detected. 1: Transmit retry-over condition has been detected.	R/W
b9	CD	Late Collision Detect Flag	0: Late collision has not been detected. 1: Late collision has been detected during frame transmission.	R/W
b10	DLC	Loss of Carrier Detect Flag	0: Loss of carrier has not been detected. 1: Loss of carrier has been detected during frame transmission.	R/W
b11	CND	Carrier Not Detect Flag	0: A carrier has been detected when transmission starts. 1: A carrier has not been detected during preamble transmission.	R/W
b15 to b12	—	Reserved	The read value is 0. The write value should be 0.	R/W
b16	RFOF	Receive FIFO Overflow Flag	0: Overflow has not occurred. 1: Overflow has occurred.	R/W
b17	RDE	Receive Descriptor Empty Flag	0: The EDMAC detects that the receive descriptor valid bit (RD0.RACT) is 1. 1: The EDMAC detects that the receive descriptor valid bit (RD0.RACT) is 0.	R/W
b18	FR	Frame Receive Flag	0: Frame has not been received. 1: Frame has been received. Update of the receive descriptor is complete.	R/W
b19	TFUF	Transmit FIFO Underflow Flag	0: Underflow has not occurred. 1: Underflow has occurred.	R/W
b20	TDE	Transmit Descriptor Empty Flag	0: The EDMAC detects that the transmit descriptor valid bit (TD0.TACT) is 1. 1: The EDMAC detects that the transmit descriptor valid bit (TD0.TACT) is 0.	R/W
b21	TC	Frame Transfer Complete Flag	0: Transfer have not been completed, or no transfer has been requested. 1: All frames indicated by the transmit descriptor have been completely transferred to the transmit FIFO.	R/W

Bit	Symbol	Bit Name	Description	R/W
b22	ECI	ETHERC Status Register Source Flag	0: ETHERC status interrupt source has not been detected. 1: ETHERC status interrupt source has been detected.	R *1
b23	—	Reserved	The read value is 0. The write value should be 0.	R
b24	RFCOF	Receive Frame Counter Overflow Flag	0: Receive frame counter has not overflowed. 1: Receive frame counter has overflowed.	R/W
b25	RABT	Receive Abort Detect Flag	0: Frame reception has not been aborted or no reception has been requested. 1: Frame reception has been aborted.	R/W
b26	TABT	Transmit Abort Detect Flag	0: Frame transmission has not been aborted or no transmission has been requested. 1: Frame transmission has been aborted.	R/W
b29 to b27	—	Reserved	The read value is 0. The write value should be 0.	R/W
b30	TWB	Write-Back Complete Flag	0: Write-back has not been completed, or no transmission has been requested. 1: Write-back to the transmit descriptor has been completed.	R/W
b31	—	Reserved	The read value is 0. The write value should be 0.	R/W

Note 1. The ECI flag is read-only. When the source in the ETHERCn.ECSR register is cleared, the ECI flag is also cleared.

The EDMACn.EESR register indicates the ETHERC and EDMAC communication status.
Each flag in the EESR register can be output as an interrupt request signal (EINTn) from the EDMAC. Each flag, excluding the ECI flag, becomes 0 by writing 1. The value of each flag is not changed by writing 0. Each interrupt source can be enabled by the corresponding bit in the EDMACn.EESIPR register.

CERF Flag (CRC Error Flag)

The CERF flag becomes 1 when an error is detected in checking the frame check sequence (FCS) field of the receive frame.

PRE Flag (PHY-LSI Receive Error Flag)

The PRE flag indicates the RX_ER signal output from the PHY-LSI is high.

RTSF Flag (Frame-Too-Short Error Flag)

The RTSF flag indicates that a received frame was less than 64 bytes.

RTLF Flag (Frame-Too-Long Error Flag)

The RTLF flag indicates that a received frame is greater than the upper limit of the receive frame length set in the ETHERCn.RFLR register. The excess data is discarded.

RRF Flag (Alignment Error Flag)

The RRF flag indicates that a frame is not an integral number of octets. The last word that is not an integral number of octets is not transferred.

RMAF Flag (Multicast Address Frame Receive Flag)

The RMAF flag indicates that a multicast frame has been received.

TRO Flag (Transmit Retry Over Flag)

This flag indicates that a collision occurred again during the 15th retry of frame transmission.

CD Flag (Late Collision Detect Flag)

The CD flag indicates that a late collision has been detected during frame transmission.

DLC Flag (Loss of Carrier Detect Flag)

The DLC flag indicates that a loss of carrier has been detected during frame transmission.

CND Flag (Carrier Not Detect Flag)

The CND flag becomes 1 when a carrier has not been detected during preamble transmission.

RFOF Flag (Receive FIFO Overflow Flag)

The RFOF flag indicates that the receive FIFO has overflowed during frame reception.

RDE Flag (Receive Descriptor Empty Flag)

The RDE flag indicates that the read receive descriptor is invalid.
When this flag becomes 1, set the RD0.RACT bit in the receive descriptor to 1 and set the EDRRR.RR bit to 1 to resume reception.

FR Flag (Frame Receive Flag)

The FR flag indicates that a frame has been received and the receive descriptor has been updated. The FR flag becomes 1 every time a frame is received.

TFUF Flag (Transmit FIFO Underflow Flag)

The TFUF flag indicates that no data remains in the transmit FIFO during frame transmission. Incomplete data is sent on the line.

TDE Flag (Transmit Descriptor Empty Flag)

The TDE flag indicates that the TD0.TACT bit of the transmit descriptor is 0 while the previous transmit descriptor indicates that the frame is not complete (TD0.TFP[1:0] bits are 10b or 00b) in multi-buffer frame trans-mission. As a result, an incomplete frame may be sent.
When this flag becomes 1, perform a software reset and then set the EDTRR.TR bit to 1 to resume transmission. Transmission starts from the address stored in the TDLAR register.

TC Flag (Frame Transfer Complete Flag)

The TC flag indicates that all the data specified by the transmit descriptor has been transmitted from the ETHERC. This flag becomes 1 when one frame has been transmitted in single-buffer frame transmission or when the last data of a frame is transmitted in multi-buffer frame transmission and the TD0.TACT bit in the next transmit descriptor is 0. After frame transmission is completed, the EDMAC writes the transfer status back to the descriptor.

ECI Flag (ETHERC Status Register Source Flag)

The ECI flag becomes 1 when an interrupt request is generated by the ETHERC.ECSR register.

ADE Flag (Address Error Flag)

The ADE flag indicates that the memory address that the EDMAC tried to transfer is invalid.

RFCOF Flag (Receive Frame Counter Overflow Flag)

The RFCOF flag indicates that the next frame reception starts while the number of frames stored in the receive FIFO reaches the maximum number of frames (16 frames). Note that the received frame is discarded while the RFCOF flag is 1.

RABT Flag (Receive Abort Detect Flag)

The RABT flag indicates that the ETHERC has aborted frame reception due to a CRC error, PHY-LSI receive error, frame-too-short error, frame-too-long error, or other error.

TABT Flag (Transmit Abort Detect Flag)

The TABT flag indicates that the ETHERC has aborted frame transmission due to transmit retry over, loss of carrier, no carrier detection, or other error.

TWB Flag (Write-Back Complete Flag)

The TWB flag indicates the EDMAC has completed writing back to the descriptor after frame transmission. Note that this flag becomes 1 after each frame transmission when the TRIMD.TIM bit is 0. This flag becomes 1 only when the TRIMD.TIS bit is 1.

29.2.7 PTP/EDMAC Status Register (PTPEDMAC.EESR)

Address: PTPEDMAC.EESR E820 4428h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	TWB	—	—	—	TABT	—	RFCOF	—	—	TC	TDE	TFUF	FR	RDE	RFOF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	MACE	RPORT	—	—	PVER	TYPE[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	TYPE[3:0]	PTP v2 Message Type Flag	b3 b0 0 0 0 0: Sync 0 0 0 1: Delay_Req 0 0 1 0: Pdelay_Req 0 0 1 1: Pdelay_Resp 1 0 0 0: Follow_Up 1 0 0 1: Delay_Resp 1 0 1 0: Pdelay_Resp_Follow_Up 1 0 1 1: Announce 1 1 0 0: Signaling 1 1 0 1: Management Settings other than above are reserved.	R/W
b4	PVER	PTP v2 Packet Flag	0: The current packet is not a PTP v2 packet. 1: The current packet is a PTP v2 packet.	R/W
b6, b5	—	Reserved	The read value is 0. The write value should be 0.	R/W
b7	RPORT	Receive Port Flag	0: Port 0 1: Port 1	R/W
8	MACE	MAC Address Mismatch Flag	0: The source MAC address of transmit frame data matches the set value. 1: The source MAC address of transmit frame data does not match the set value.	R/W
b15 to b9	—	Reserved	The read value is 0. The write value should be 0.	R/W
b16	RFOF	Receive FIFO Overflow Flag	0: Overflow has not occurred. 1: Overflow has occurred.	R/W
b17	RDE	Receive Descriptor Empty Flag	0: The EDMAC detects that the receive descriptor valid bit (RD0.RACT) is 1. 1: The EDMAC detects that the receive descriptor valid bit (RD0.RACT) is 0.	R/W
b18	FR	Frame Receive Flag	0: Frame has not been received. 1: Frame has been received. The receive descriptor has been updated.	R/W
b19	TFUF	Transmit FIFO Underflow Flag	0: Underflow has not occurred. 1: Underflow has occurred.	R/W
b20	TDE	Transmit Descriptor Empty Flag	0: The EDMAC detects that the transmit descriptor valid bit (TD0.TACT) is 1. 1: The EDMAC detects that the transmit descriptor valid bit (TD0.TACT) is 0.	R/W
b21	TC	Frame Transfer Complete Flag	0: Transfer has not been completed, or transfer has not been requested. 1: All frames indicated by the transmit descriptor have been completely transferred to the transmit FIFO.	R/W
b23, b22	—	Reserved	The read value is 0. The write value should be 0.	R/W
b24	RFCOF	Receive Frame Counter Overflow Flag	0: Receive frame counter has not overflowed. 1: Receive frame counter has overflowed.	R/W

Bit	Symbol	Bit Name	Description	R/W
b25	—	Reserved	The read value is 0. The write value should be 0.	R/W
b26	TABT	Transmit Abort Detect Flag	0: Frame transmission has not been aborted or transmission has not been requested. 1: Frame transmission has been aborted.	R/W
b29 to b27	—	Reserved	The read value is 0. The write value should be 0.	R/W
b30	TWB	Write-Back Complete Flag	0: Write-back has not been completed, or transmission has not been requested. 1: Write-back to the transmit descriptor has been completed.	R/W
b31	—	Reserved	The read value is 0. The write value should be 0.	R/W

The PTPEDMAC.EESR register indicates the PTPEDMAC communication status.
Each flag in the EESR register can be output as an interrupt request signal (PINT) from the PTPEDMAC. Each flag becomes 0 by writing 1. The value of each flag is not changed by writing 0. Each interrupt source, excluding the TYPE[3:0] flag, can be enabled by setting the corresponding bit in the PTPEDMAC.EESIPR register.

TYPE[3:0] Flags (PTP v2 Message Type Flag)

The TYPE[3:0] flags indicate the type of received PTP message.

PVER Flag (PTP v2 Packet Flag)

The PVER flag indicates whether the received packet is a PTP v2 packet.

RPORT Flag (Receive Port Flag)

The RPORT flag indicates which Ethernet port has been used for receiving PTP messages.

MACE Flag (MAC Address Mismatch Flag)

The MACE flag indicates the source MAC address is different from the set value.

RFOF Flag (Receive FIFO Overflow Flag)

The RFOF flag indicates that the receive FIFO has overflowed during frame reception.

RDE Flag (Receive Descriptor Empty Flag)

The RDE flag indicates that the read receive descriptor is invalid.
When this flag becomes 1, set the RD0.RACT bit in the receive descriptor to 1 and set the EDRRR.RR bit to 1 to resume reception.

FR Flag (Frame Receive Flag)

The FR flag indicates that a frame has been received and the receive descriptor has been updated. The FR flag becomes 1 every time a frame is received.

TFUF Flag (Transmit FIFO Underflow Flag)

The TFUF flag indicates that no data remains in the transmit FIFO during frame transmission. Incomplete data is sent on the line.

TDE Flag (Transmit Descriptor Empty Flag)

The TDE flag indicates that the TD0.TACT bit of the transmit descriptor is 0 while the previous transmit descriptor indicates that the frame is not complete (TD0.TFP[1:0] bits are 10b or 00b) in multi-buffer frame transmission. As a result, an incomplete frame may be sent.
When this flag becomes 1, perform a software reset and then set the EDTRR.TR bit to 1 to resume transmission. Transmission starts from the address stored in the TDLAR register.

TC Flag (Frame Transfer Complete Flag)

The TC flag indicates that all the data specified by the transmit descriptor has been transmitted from the ETHERC. This flag becomes 1 when one frame has been transmitted in single-buffer frame transmission or when the last data of a frame is transmitted in multi-buffer frame transmission and the TD0.TACT bit in the next transmit descriptor is 0. After frame transmission is completed, the PTPEDMAC writes the transfer status back to the descriptor.

ADE Flag (Address Error Flag)

The ADE flag indicates that the memory address that the PTPEDMAC tried to transfer is invalid.
RFCOF Flag (Receive Frame Counter Overflow Flag)
The RFCOF flag indicates that the next frame reception starts while the number of frames stored in the receive FIFO reaches the maximum number of frames (16 frames). Note that the received frame is discarded while the RFCOF flag is 1.

TABT Flag (Transmit Abort Detect Flag)

The TABT flag indicates that the ETHERC has aborted frame transmission due to transmit retry over, loss of carrier, no carrier detection, or other error.

TWB Flag (Write-Back Complete Flag)

The TWB flag indicates the PTPEDMAC has completed writing back to the descriptor after frame transmission. Note that this flag becomes 1 after each frame transmission when the TRIMD.TIM bit is 0. This flag becomes 1 only when the TRIMD.TIS bit is 1.

29.2.8 ETHERC/EDMAC Status Interrupt Enable Register (EDMACn.EESIPR)

Address: EDMAC0.EESIPR E820 4030h, EDMAC1.EESIPR E820 4230h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	TWBIP	—	—	—	TABTIP	RABTIP	RFCOFIP	—	ECIIP	TCIP	TDEIP	TFUFIP	FRIP	RDEIP	RFOFIP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	CNDIP	DLCIP	CDIP	TROIP	RMAFIP	—	—	RRFIP	RTLFIP	RTSFIP	PREIP	CERFIP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CERFIP	CRC Error Interrupt Request Enable	0: CRC error interrupt request is disabled. 1: CRC error interrupt request is enabled.	R/W
b1	PREIP	PHY-LSI Receive Error Interrupt Request Enable	0: PHY-LSI receive error interrupt request is disabled. 1: PHY-LSI receive error interrupt request is enabled.	R/W
b2	RTSFIP	Frame-Too-Short Error Interrupt Request Enable	0: Frame-too-short error interrupt request is disabled. 1: Frame-too-short error interrupt request is enabled.	R/W
b3	RTLFIP	Frame-Too-Long Error Interrupt Request Enable	0: Frame-too-long error interrupt request is disabled. 1: Frame-too-long error interrupt request is enabled.	R/W
b4	RRFIP	Alignment Error Interrupt Request Enable	0: Alignment error interrupt request is disabled. 1: Alignment error interrupt request is enabled.	R/W
b6, b5	—	Reserved	The read value is 0. The write value should be 0.	R/W
b7	RMAFIP	Multicast Address Frame Receive Interrupt Request Enable	0: Multicast address frame receive interrupt request is disabled. 1: Multicast address frame receive interrupt request is enabled.	R/W
b8	TROIP	Transmit Retry Over Interrupt Request Enable	0: Transmit retry over interrupt request is disabled. 1: Transmit retry over interrupt request is enabled.	R/W
b9	CDIP	Late Collision Detect Interrupt Request Enable	0: Late collision detect interrupt request is disabled. 1: Late collision detect interrupt request is enabled.	R/W
b10	DLCIP	Loss of Carrier Detect Interrupt Request Enable	0: Loss of carrier detect interrupt request is disabled. 1: Loss of carrier detect interrupt request is enabled.	R/W
b11	CNDIP	Carrier Not Detect Interrupt Request Enable	0: Carrier not detect interrupt request is disabled. 1: Carrier not detect interrupt request is enabled.	R/W
b15 to b12	—	Reserved	The read value is 0. The write value should be 0.	R/W
b16	RFOFIP	Receive FIFO Overflow Interrupt Request Enable	0: Overflow interrupt request is disabled. 1: Overflow interrupt request is enabled.	R/W
b17	RDEIP	Receive Descriptor Empty Interrupt Request Enable	0: Receive descriptor empty interrupt request is disabled. 1: Receive descriptor empty interrupt request is enabled.	R/W
b18	FRIP	Frame Receive Interrupt Request Enable	0: Frame reception interrupt request is disabled. 1: Frame reception interrupt request is enabled.	R/W
b19	TFUFIP	Transmit FIFO Underflow Interrupt Request Enable	0: Underflow interrupt request is disabled. 1: Underflow interrupt request is enabled.	R/W
b20	TDEIP	Transmit Descriptor Empty Interrupt Request Enable	0: Transmit descriptor empty interrupt request is disabled. 1: Transmit descriptor empty interrupt request is enabled.	R/W
b21	TCIP	Frame Transfer Complete Interrupt Request Enable	0: Frame transmission complete interrupt request is disabled. 1: Frame transmission complete interrupt request is enabled.	R/W
b22	ECIIP	ETHERC Status Register Source Interrupt Request Enable	0: ETHERC status interrupt request is disabled. 1: ETHERC status interrupt request is enabled.	R/W
b23	—	Reserved	The read value is 0. The write value should be 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b24	RFCOFIP	Receive Frame Counter Overflow Interrupt Request Enable	0: Receive frame counter overflow interrupt request is disabled. 1: Receive frame counter overflow interrupt request is enabled	R/W
b25	RABTIP	Receive Abort Detect Interrupt Request Enable	0: Receive abort detect interrupt request is disabled. 1: Receive abort detect interrupt request is enabled.	R/W
b26	TABTIP	Transmit Abort Detect Interrupt Request Enable	0: Transmit abort detect interrupt request is disabled. 1: Transmit abort detect interrupt request is enabled.	R/W
b29 to b27	—	Reserved	The read value is 0. The write value should be 0.	R/W
b30	TWBIP	Write-Back Complete Interrupt Request Enable	0: Write-back complete interrupt request is disabled. 1: Write-back complete interrupt request is enabled.	R/W
b31	—	Reserved	The read value is 0. The write value should be 0.	R/W

The EDMACn.EESIPR register enables interrupt requests corresponding to bits in the EDMACn.EESR register. When a bit in this register is 1, the corresponding interrupt request is enabled.

29.2.9 PTP/EDMAC Status Interrupt Enable Register (PTPEDMAC.EESIPR)

Address: PTPEDMAC.EESIPR E820 4430h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	TWBIP	—	—	—	TABTIP	—	RFCOFIP	—	—	TCIP	TDEIP	TFUFIP	FRIP	RDEIP	RFOFIP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	MACEIP	RPORTIP	—	—	PVERIP	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	The read value is 0. The write value should be 0.	R/W
b4	PVERIP	PTP v2 Packet Receive Interrupt Request Enable	0: PTP v2 packet receive interrupt request is disabled. 1: PTP v2 packet receive interrupt request is enabled.	R/W
b6, b5	—	Reserved	The read value is 0. The write value should be 0.	R/W
b7	RPORTIP	Receive Port Interrupt Request Enable	0: Interrupt request at a fame reception on port 1 is disabled. 1: Interrupt request at a fame reception on port 1 is enabled.	R/W
b8	MACEIP	MAC Address Mismatch Interrupt Request Enable	0: This bit disables an interrupt request generated when the source MAC address of transmit frame data does not match the set value. 1: This bit enables an interrupt request generated when the source MAC address of transmit frame data does not match the set value.	R/W
b15 to b9	—	Reserved	The read value is 0. The write value should be 0.	R/W
b16	RFOFIP	Receive FIFO Overflow Interrupt Request Enable	0: Overflow interrupt request is disabled. 1: Overflow interrupt request is enabled.	R/W
b17	RDEIP	Receive Descriptor Empty Interrupt Request Enable	0: Receive descriptor empty interrupt request is disabled. 1: Receive descriptor empty interrupt request is enabled.	R/W
b18	FRIP	Frame Receive Interrupt Request Enable	0: Frame receive interrupt request is disabled. 1: Frame receive interrupt request is enabled	R/W
b19	TFUFIP	Transmit FIFO Underflow Interrupt Request Enable	0: Underflow interrupt request is disabled. 1: Underflow interrupt request is enabled.	R/W
b20	TDEIP	Transmit Descriptor Empty Interrupt Request Enable	0: Transmit descriptor empty interrupt request is disabled. 1: Transmit descriptor empty interrupt request is enabled.	R/W
b21	TCIP	Frame Transfer Complete Interrupt Request Enable	0: Frame transmission complete interrupt request is disabled. 1: Frame transmission complete interrupt request is enabled.	R/W
b23, b22	—	Reserved	The read value is 0. The write value should be 0.	R/W
b24	RFCOFIP	Receive Frame Counter Overflow Interrupt Request Enable	0: Receive frame counter overflow interrupt request is disabled. 1: Receive frame counter overflow interrupt request is enabled.	R/W
b25	—	Reserved	The read value is 0. The write value should be 0.	R/W
b26	TABTIP	Transmit Abort Detect Interrupt Request Enable	0: Transmit abort detect interrupt request is disabled. 1: Transmit abort detect interrupt request is enabled.	R/W
b29 to b27	—	Reserved	The read value is 0. The write value should be 0.	R/W
b30	TWBIP	Write-Back Complete Interrupt Request Enable	0: Write-back complete interrupt request is disabled. 1: Write-back complete interrupt request is enabled.	R/W
b31	—	Reserved	The read value is 0. The write value should be 0.	R/W

The PTPEDMAC.EESIPR register enables interrupt requests corresponding to bits in the PTPEDMAC.EESR register. When a bit in this register is 1, the corresponding interrupt request is enabled.

29.2.10 ETHERC/EDMAC Transmit/Receive Status Copy Enable Register (EDMACn.TRSCER)

Address: EDMAC0.TRSCER E820 4038h, EDMAC1.TRSCER E820 4238h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	RMAFCE	—	—	RRFCE	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	The read value is 0. The write value should be 0.	R/W
b4	RRFCE	RRF Flag Copy Enable	0: The EESR.RRF flag status is reflected in the RD0.RFE bit of the receive descriptor. 1: The EESR.RRF flag status is not reflected in the RD0.RFE bit of the receive descriptor.	R/W
b6, b5	—	Reserved	The read value is 0. The write value should be 0.	R/W
b7	RMAFCE	RMAF Flag Copy Enable	0: The EESR.RMAF flag status is reflected in the RD0.RFE bit of the receive descriptor. 1: The EESR.RMAF flag status is not reflected in the RD0.RFE bit of the receive descriptor.	R/W
b31 to b8	—	Reserved	The read value is 0. The write value should be 0.	R/W

The EDMACn.TRSCER register selects whether the receive status indicated by flags EDMACn.EESR.RMAF and RRF is reflected in the RFE bit of the receive descriptor as a summary. The bits in the TRSCER register correspond to bits in the EESR register that have the same number. When setting the RMAFCE or RRFCE bit to 0, the corresponding receive status (EESR.RMAF or RRF flag) is reflected in the RFE bit of the receive descriptor. When setting the RMAFCE or RRFCE bit to 1, the corresponding receive status is not reflected in the RFE bit.

29.2.11 Missed-Frame Counter Register (RMFCR)

Address: EDMAC0.RMFCR E820 4040h, EDMAC1.RMFCR E820 4240h, PTPEDMAC.RMFCR E820 4440h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	MFC[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	MFC[15:0]	Missed-Frame Counter	These bits indicate the number of frames that are discarded and not transferred to the receive buffer during reception.	R/W
b31 to b16	—	Reserved	The read value is 0. The write value should be 0.	R/W

The RMFCR register indicates that the number of frames that could not be stored in the receive FIFO and were therefore discarded during reception. When the receive FIFO overflows, it stops receiving data, and the remain-ing frames are discarded. At the same time, the RMFCR register value is incremented. When the RMFCR register value reaches FFFFh, incrementing is halted. When any value is written to the RMFCR register, the counter value becomes 0.
For the frame that has not been completely received, after data in the receive FIFO is transferred to the receive buffer, the RACT bit in the receive descriptor 0 (RD0) becomes 0 (descriptor disabled), the RFS9 bit becomes 1 (receive FIFO overflow), and the EDMACn.EESR.RFOF or PTPEDMAC.EESR.RFOF flag becomes 1 (overflow has occurred).

29.2.12 Transmit FIFO Threshold Register (TFTR)

Address: EDMAC0.TFTR E820 4048h, EDMAC1.TFTR E820 4248h, PTPEDMAC.TFTR E820 4448h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TFT[10:0]										
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b10 to b0	TFT[10:0]	Transmit FIFO Threshold	b10 b0 000h: Store and forward mode 001h to 00Ch: Setting prohibited 00Dh to 200h: The threshold is the set value multiplied by 4. Example: 00Dh: 52 bytes 040h: 256 bytes 100h: 1024 bytes 200h: 2048 bytes 201h to 7FFh: Setting prohibited	R/W
b31 to b11	—	Reserved	The read value is 0. The write value should be 0.	R/W

Note 1. When starting transmission before one frame data has been completely written, take care to prevent an underflow.
Note 2. To prevent a transmit underflow, using the initial value (store and forward mode) is recommended.

The TFTR register sets the transmit FIFO threshold at which the first transmission starts. The actual threshold is the set value multiplied by 4.

The ETHERC starts transmission when the amount of data in the transmit FIFO exceeds the number of bytes set in the TFTR register, when the transmit FIFO is full, or when one frame data has been completely written. Set the TFTR register while the EDTRR.TR bit is 0.

29.2.13 FIFO Depth Register (FDR)

Address: EDMAC0.FDR E820 4050h, EDMAC1.FDR E820 4250h, PTPEDMAC.FDR E820 4450h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	TFD[4:0]					—	—	—	RFD[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	RFD[4:0]	Receive FIFO Depth	b4 b0 01111: 4096 bytes Settings other than above are prohibited.	R/W
b7 to b5	—	Reserved	The read value is 0. The write value should be 0.	R/W
b12 to b8	TFD[4:0]	Transmit FIFO Depth	b12 b8 00111: 2048 bytes Settings other than above are prohibited.	R/W
b31 to b13	—	Reserved	The read value is 0. The write value should be 0.	R/W

The FDR register sets the transmit and receive FIFO depths.
Set this register to 0000 070Fh before starting transmission and reception.

29.2.14 Receive Method Control Register (RMCR)

Address: EDMAC0.RMCR E820 4058h, EDMAC1.RMCR E820 4258h, PTPEDMAC.RMCR E820 4458h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RNR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	RNR	Receive Request Reset	0: EDRRR.RR bit (receive request bit) is set to 0 when one frame has been received. 1: EDRRR.RR bit (receive request bit) is not set to 0 when one frame has been received.	R/W
b31 to b1	—	Reserved	The read value is 0. The write value should be 0.	R/W

The RMCR register sets how to control the EDRRR.RR bit when receiving a frame.
Since the EDRRR.RR bit becomes 0 when one frame has been received while the RNR bit is 0, the RR bit needs to be set to 1 by software to receive the subsequent frame.
Since the EDRRR.RR bit does not become 0 when one frame has been received while the RNR bit is 1, the EDMAC reads the next receive descriptor and continues receiving frames. It is recommended to set the RNR bit to 1 when receiving data continuously.
Set the RMCR register while the EDRRR.RR bit is 0.

29.2.15 Transmit FIFO Underflow Counter (TFUCR)

Address: EDMAC0.TFUCR E820 4064h, EDMAC1.TFUCR E820 4264h, PTPEDMAC.TFUCR E820 4464h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	UNDER[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	UNDER[15:0]	Transmit FIFO Underflow Count	These bits indicate how many times the transmit FIFO has underflowed. The counter stops when the counter value reaches FFFFh.	R/W
b31 to b16	—	Reserved	The read value is 0. The write value should be 0.	R/W

The TFUCR register indicates how many times the transmit FIFO has underflowed.
When writing any value to the TFUCR register, the counter value becomes 0.

29.2.16 Receive FIFO Overflow Counter (RFOCR)

Address: EDMAC0.RFOCR E820 4068h, EDMAC1.RFOCR E820 4268h, PTPEDMAC.RFOCR E820 4468h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OVER[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	OVER[15:0]	Receive FIFO Overflow Count	These bits indicate how many times the receive FIFO has overflowed. The counter stops when the counter value reaches FFFFh.	R/W
b31 to b16	—	Reserved	The read value is 0. The write value should be 0.	R/W

The RFOCR register indicates how many times the receive FIFO has overflowed.
When writing any value to the RFOCR register, the counter value becomes 0.

29.2.17 Independent Output Signal Setting Register (IOSR)

Address: EDMAC0.IOSR E820 406Ch, EDMAC1.IOSR E820 426Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ELB
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ELB	External Loopback Mode	0: The ETn_EXOUT pin outputs low. 1: The ETn_EXOUT pin outputs high.	R/W
b31 to b1	—	Reserved	The read value is 0. The write value should be 0.	R/W

The IOSR register selects the output level of the ETHERC external output pin (ETn_EXOUT) in external loopback mode.

The ELB bit value is output to the ETn_EXOUT pin of the MCU as is, which can be used to set loopback mode for the PHY-LSI. To use the loopback function of the PHY-LSI through this register, the PHY-LSI needs to be provided with the pin to be connected to the ETn_EXOUT pin.

29.2.18 Flow Control Start FIFO Threshold Setting Register (FCFTR)

Address: EDMAC0.FCFTR E820 4070h, EDMAC1.FCFTR E820 4270h, PTPEDMAC.FCFTR E820 4470h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	RFFO[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RFDO[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	RFDO[2:0]	Receive FIFO Data PAUSE Output Threshold	b2 b0 0 0 0: When 224 (256 - 32) bytes of data is stored in the receive FIFO. 0 0 1: When 480 (512 - 32) bytes of data is stored in the receive FIFO. : 1 1 0: When 1760 (1792 - 32) bytes of data is stored in the receive FIFO. 1 1 1: When 2016 (2048 - 32) bytes of data is stored in the receive FIFO.	R/W
b15 to b3	—	Reserved	The read value is 0. The write value should be 0.	R/W
b18 to b16	RFFO[2:0]	Receive FIFO Frame PAUSE Output Threshold	b18 b16 0 0 0: When 2 receive frames have been stored in the receive FIFO. 0 0 1: When 4 receive frames have been stored in the receive FIFO. 0 1 0: When 6 receive frames have been stored in the receive FIFO. : 1 1 0: When 14 receive frames have been stored in the receive FIFO. 1 1 1: When 16 receive frames have been stored in the receive FIFO.	R/W
b31 to b19	—	Reserved	The read value is 0. The write value should be 0.	R/W

The FCFTR register sets the ETHERC flow control (sets the threshold for automatically transmitting a PAUSE frame). The threshold can be set using the data size (RFDO[2:0] bits) and the number of frames (RFFO[2:0] bits) stored in the receive FIFO. The flow control starts when the stored data size or number of stored frames reaches its threshold.

29.2.19 Receive Data Padding Insert Register (RPADIR)

Address: EDMAC0.RPADIR E820 4078h, EDMAC1.RPADIR E820 4278h, PTPEDMAC.RPADIR E820 4478h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PADS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	PADR[5:0]					
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PADR[5:0]	Padding Slot	b5 b0 00h: Padding is inserted at the head of received data. 01h: Padding is inserted between the 1st byte and 2nd byte of received data. : 3Eh: Padding is inserted between the 62nd byte and 63rd byte of received data. 3Fh: Padding is inserted between the 63rd byte and 64th byte of received data	R/W
b15 to b6	—	Reserved	The read value is 0. The write value should be 0.	R/W
b17, b16	PADS[1:0]	Padding Size	b17 b16 0 0: No padding is inserted. 0 1: 1 byte is inserted. 1 0: 2 bytes are inserted. 1 1: 3 bytes are inserted.	R/W
b31 to b18	—	Reserved	The read value is 0. The write value should be 0.	R/W

The RPADIR register sets insertion of padding for received data. The padding value is 00h.
Set the EDMR.SWR bit to 1 to reset before rewriting the RPADIR register.

29.2.20 Transmit Interrupt Setting Register (TRIMD)

Address: EDMAC0.TRIMD E820 407Ch, EDMAC1.TRIMD E820 427Ch, PTPEDMAC.TRIMD E820 447Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	TIM	—	—	—	TIS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TIS	Transmit Interrupt Enable	0: Transmit Interrupt is disabled. 1: Transmit Interrupt is enabled. Set the EESR.TWB flag to 1 in the mode selected by the TIM bit to notify an interrupt.	R/W
b3 to b1	—	Reserved	The read value is 0. The write value should be 0.	R/W
b4	TIM	Transmit Interrupt Mode	0: Transmission complete interrupt mode An interrupt occurs when a frame has been transmitted. 1: Write-back complete interrupt mode An interrupt occurs when write-back to the transmit descriptor has been completed while the TWBI bit is 1.	R/W
b31 to b5	—	Reserved	The read value is 0. The write value should be 0.	R/W

The TRIMD register sets transmit interrupt mode and enables/disables the transmit interrupt.
When the condition set in this register is satisfied, the EESR.TWB flag becomes 1, and an interrupt request is output when the EESIPR.TWBIP bit is 1.

29.2.21 Receive Buffer Write Address Register (RBWAR)

Address: EDMAC0.RBWAR E820 40C8h, EDMAC1.RBWAR E820 42C8h, PTPEDMAC.RBWAR E820 44C8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The RBWAR register indicates the last address that the EDMAC has written data to when writing to the receive buffer. Refer to the address indicated by the RBWAR register to recognize which address in the receive buffer the EDMAC is writing data to. Note that the address that the EDMAC is outputting to the receive buffer may not match the read value of the RBWAR register during data reception.
The RBWAR register is read only. Do not write to this register.

29.2.22 Receive Descriptor Fetch Address Register (RDFAR)

Address: EDMAC0.RDFAR E820 40CCh, EDMAC1.RDFAR E820 42CCh, PTPEDMAC.RDFAR E820 44CCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The RDFAR register indicates the start address of the last fetched receive descriptor when the EDMAC fetches descriptor information from the receive descriptor.
Refer to the address indicated by the RDFAR register to recognize which receive descriptor information the EDMAC is using for the current processing. Note that the address of the receive descriptor that the EDMAC fetches may not match the read value of the RDFAR register during data reception.
The RDFAR is read only. Do not write to this register.

29.2.23 Transmit Buffer Read Address Register (TBRAR)

Address: EDMAC0.TBRAR E820 40D4h, EDMAC1.TBRAR E820 42D4h, PTPEDMAC.TBRAR E820 44D4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The TBRAR register indicates the last address that the EDMAC has read data from when reading data from the transmit buffer.
Refer to the address indicated by the TBRAR register to recognize which address in the transmit buffer the EDMAC is reading from. Note that the address that the EDMAC is outputting to the transmit buffer may not match the read value of the TBRAR register.
The TBRAR register is read only. Do not write to this register.

29.2.24 Transmit Descriptor Fetch Address Register (TDFAR)

Address: EDMAC0.TDFAR E820 40D8h, EDMAC1.TDFAR E820 42D8h, PTPEDMAC.TDFAR E820 44D8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The TDFAR register indicates the start address of the last fetched transmit descriptor when the EDMAC fetches descriptor information from the transmit descriptor.
Refer to the address indicated by the TDFAR register to recognize which transmit descriptor information the EDMAC is using for the current processing. Note that the address of the transmit descriptor that the EDMAC fetches may not match the read value of the TDFAR register.
The TDFAR is read only. Do not write to this register.

29.3 Operation

The EDMAC transfers data according to the information written in the descriptor. Two types of descriptors are provided: transmit descriptor and receive descriptor. A descriptor includes the buffer size, address, and transmit/receive status. The EDMAC transmits or receives data continuously using sequentially arranged descriptors (descriptor list).

29.3.1 Descriptor Lists and Data Buffers

To transfer data using the EDMAC, create the transmit and receive descriptor lists on the memory, set the start address of the transmit descriptor list to the TDLAR register, and set the start address of the receive descriptor list to the RDLAR register. Also, transmit and receive buffers corresponding to the each descriptor are required.
Align the descriptor list on the appropriate address boundary according to the descriptor length set by the EDMR.DL[1:0] bits. The transmit buffer can be aligned on a long word boundary, word boundary, or byte boundary. When the valid transmit buffer size is 16 bytes or less, align it on a 32-byte boundary. Align the receive buffer on a 32-byte boundary. Set different addresses for the transmit and receive descriptors and buffers for EDMAC0, EDMAC1, and the PTPEDMAC.

29.3.1.1 Transmit Descriptor

Figure 29.3 shows the relation between a transmit descriptor and transmit buffer. A transmit descriptor consists of TD0 to TD2. The transmit frame and transmit buffer configuration can be selected from one buffer per frame (single-buffer frame transmission) or multiple buffers per frame (multi-buffer frame transmission) by setting the transmit descriptor.

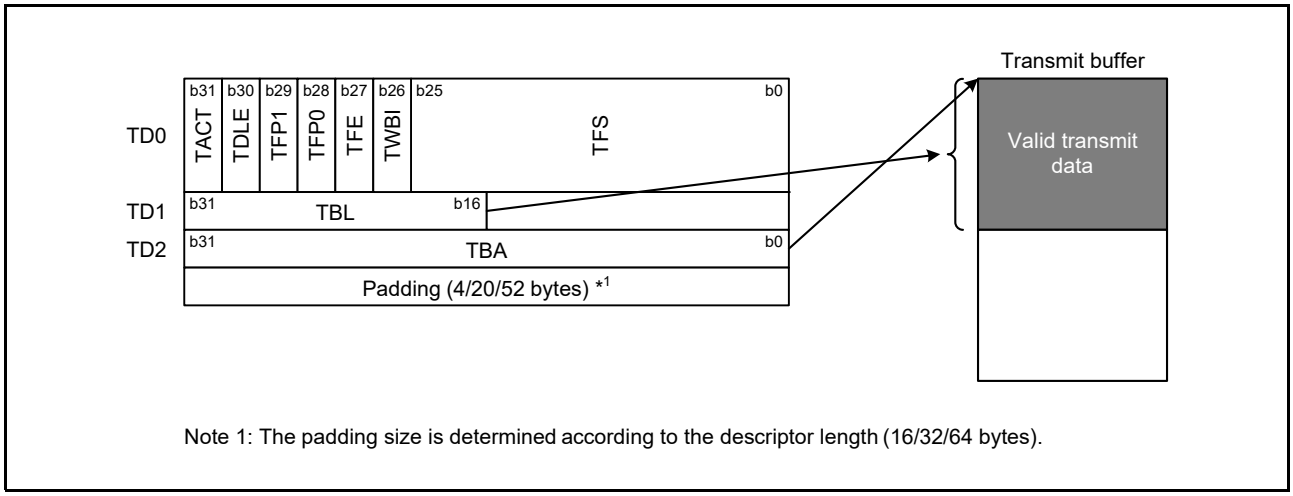


Figure 29.3 Relation between a Transmit Descriptor and Transmit Buffer

(1) Transmit Descriptor 0 (TD0)

Bit	Symbol	Bit Name	Description	R/W
b25 to b0	TFS	Transmit Frame Status	Set all bits to 0 when creating a descriptor. After write-back, the bits indicate the following: [for EDMACn] TFS25 to TFS9: Reserved TFS8: Transmit abort is detected (the value is equivalent to the EESR.TABT flag). TFS7 to TFS4: Reserved TFS3: No carrier is detected (the value is equivalent to the EESR.CND flag). TFS2: Loss of carrier is detected (the value is equivalent to the EESR.DLC flag). TFS1: Late collision during transmission is detected (the value is equivalent to the EESR.CD flag). TFS0: Transmit retry over (the value is equivalent to the EESR.TRO flag) When each bit becomes 1, it indicates that the corresponding error has occurred during frame transmission. When any of the TFS bits becomes 1, the TFE bit also becomes 1. When any bit from TFS3 to TFS0 becomes 1, the TFS8 bit also becomes 1. [for the PTPEDMAC] TFS25 to TFS9: Reserved TFS8: Transmit abort is detected (the value is equivalent to the EESR.TABT flag). TFS7 to TFS1: Reserved TFS0: The transmission source MAC address of the transmit frame data does not match the set value (the value is equivalent to the EESR.MACE flag). When each bit becomes 1, it indicates that the corresponding error has occurred during frame transmission. When any of the TFS bits becomes 1, the TFE bit also becomes 1. When TFS0 becomes 1, TFS8 also becomes 1.	R/W
b26	TWBI	Write-Back Complete Interrupt Enable	0: Interrupt does not occur when write-back to this descriptor has been completed. 1: Interrupt occurs when write-back to this descriptor has been completed.	R/W
b27	TFE	Transmit Frame Error	0: Frame transmission is successfully completed. 1: An error occurs during frame transmission (transmission aborted).	R/W
b29, b28	TFP[1:0]	Transmit Frame Position	b29 b28 0 0: Transmit buffer indicated by this descriptor is the middle of a transmit frame (frame information is incomplete). 0 1: Transmit buffer indicated by this descriptor is the end of a transmit frame (frame information is complete). 1 0: Transmit buffer indicated by this descriptor is the head of a transmit frame (frame information is incomplete.) 1 1: Transmit buffer indicated by this descriptor is all of a transmit frame (one buffer per frame).	R/W
b30	TDLE	Transmit Descriptor List End	When this bit is 1, it indicates that this descriptor is the last descriptor of the descriptor list.	R/W
b31	TACT	Transmit Descriptor Valid	This bit indicates that this descriptor is valid.	R/W

Note: Bits for write-back are underlined.

TD0 indicates the transmit frame settings, and the status after transmission.

TWBI Bit (Write-Back Complete Interrupt Enable)

This bit setting is valid, when the TRIMD.TIM bit is 1.

To generate an interrupt request, the TRIMD.TIS and EESIPR.TWBIP bits must be set to 1.

TFE Bit (Transmit Frame Error)

When the TFE bit is 1, it indicates that one or more of the TFS bits is 1.

TFP[1:0] Bits (Transmit Frame Position)

The TFP[1:0] bits indicate which part of a transmit frame corresponds to the transmit buffer indicated by this descriptor. The TFP[1:0] and TD1.TBL bit settings must be logically consistent in the previous and next descriptors.

TACT Bit (Transmit Descriptor Valid)

The TACT bit indicates that this descriptor is valid. The TACT bit is set to 1 by software. This bit becomes 0 when the transmit frame has been transferred or when the transmission is aborted.

(2) Transmit Descriptor 1 (TD1)

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	Reserved	The read value is 0. The write value should be 0.	R/W
b31 to b16	TBL	Transmit Buffer Length	Set a valid byte length of the corresponding transmit buffer. Set a value equal to or greater than 1.	R/W

TD1 is used to set a valid byte length of the transmit buffer.

(3) Transmit Descriptor 2 (TD2)

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	TBA	Transmit Buffer Address	Set the start address of the transmit buffer. When the TD1.TBL bit value is 1 to 16 bytes, align it on a 32-byte boundary.	R/W

TD2 is used to set the start address of the transmit buffer.

29.3.1.2 Receive Descriptor

Figure 29.4 shows the relation between a receive descriptor and receive buffer. The receive frame and receive buffer configuration can be selected from one buffer per frame (single-buffer frame transmission) or multiple buffers per frame (multi-buffer frame transmission) by setting the receive descriptor. If the receive buffer length (RBL) is set to 0, operation indicated by the descriptor is not guaranteed.

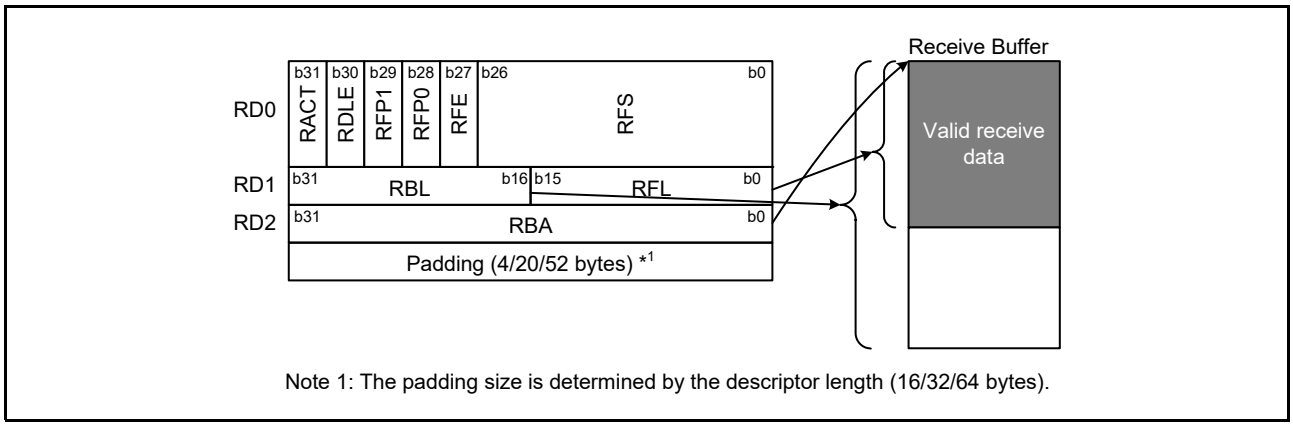


Figure 29.4 Relation between Receive Descriptor and Receive Buffer

(1) Receive Descriptor 0 (RD0)

Bit	Symbol	Bit Name	Description	R/W
b26 to b0	RFS	Receive Frame Status	Set all bits to 0 when creating a descriptor. After write-back, the bits indicate the following: [for EDMACn] RFS26 to RFS10: Reserved RFS9: Receive FIFO overflow (the value is equivalent to the EESR.RFOF flag) RFS8: Receive abort is detected (the value is equivalent to the EESR.RABT flag). RFS7: Multicast address frame is received (the value is equivalent to the EESR.RMAF flag). RFS6 and RFS5: Reserved RFS4: Alignment error is detected (the value is equivalent to the EESR.RRF flag). RFS3: Frame-too-long error (the value is equivalent to the EESR.RTLF flag) RFS2: Frame-too-short error (the value is equivalent to the EESR.RTSF flag) RFS1: PHY-LSI receive error (the value is equivalent to the EESR.PRE flag) RFS0: CRC error (the value is equivalent to the EESR.CERF flag) When each bit becomes 1, it indicates that the corresponding error has occurred during frame reception. When any of the RFS bits becomes 1, the RFE bit also becomes 1 (set the TRSCER register to select whether bits RFS7 and RFS4 are reflected in the RFE bit). When any bit from RFS3 to RFS0 becomes 1, the RFS8 bit also becomes 1. [for the PTPEDMAC] RFS26 to RFS10: Reserved RFS9: Receive FIFO overflow (the value is equivalent to the EESR.RFOF flag) RFS8: Reserved RFS7: Receive port (the value is equivalent to the EESR.RPORT flag) RFS4: PTPV2 packet is received (the value is equivalent to the EESR.PVER flag). The PTPEDMAC can receive only PTP packets. If a non-PTP packet is received, the packet is not transferred to the PTPEDMAC, and it is discarded. RFS3 to RFS0: Type of the received PTP message (the value is equivalent to the EESR.TYPE[3:0] flags) Each bit indicates the status of the received frame.	R/W
b27	RFE	Receive Frame Error	[for EDMACn] 0: No error has occurred in the received frame. 1: An error has occurred in the received frame. [for the PTPEDMAC] Reserved	R/W
b29, b28	RFP[1:0]	Receive Frame Position	b29 b28 0 0: Receive buffer indicated by this descriptor is the middle of a receive frame (frame information is incomplete). 0 1: Receive buffer indicated by this descriptor is the end of a receive frame (frame information is complete). 1 0: Receive buffer indicated by this descriptor is the head of a receive frame (frame information is incomplete). 1 1: Receive buffer indicated by this descriptor is all of a receive frame (one buffer per frame).	R/W
b30	RDLE	Receive Descriptor List End	When this bit is 1, it indicates that this descriptor is the last one of the descriptor list.	R/W
b31	RACT	Receive Descriptor Valid	Indicates that this descriptor is valid.	R/W

Note: Bits for write-back are underlined.

RD0 indicates the receive frame status.

RFE Bit (Receive Frame Error)

When the RFE bit is 1, it indicates that one or more of the RFS bits is 1 (set the TRSCER register to select whether bits RFS7 and RFS4 of EDMACn are reflected in the RFE bit).

RFP[1:0] Bits (Receive Frame Position)

The RFP[1:0] bits indicate which part of a receive frame corresponds to the receive buffer indicated by this descriptor.

RACT Bit (Receive Descriptor Valid)

The RACT bit indicates that this descriptor is valid. The RACT bit is set to 1 by software. This bit becomes 0 when all data has been transferred to the receive buffer indicated by RD2 or when the receive buffer becomes full.

(2) Receive Descriptor 1 (RD1)

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	RFL	Receive Frame Length	These bits indicate the length (number of bytes) of the receive frame stored in the buffer. The number of bytes for padding set by the RPADIR register is not included. These bits are written back to the descriptor corresponding to the end of a frame.	R/W
b31 to b16	RBL	Receive Buffer Length	These bits indicate the byte length of the corresponding receive buffer. Set an integral multiple of 32 as the buffer length.	R/W

Note: Bits for write-back are underlined.

RD1 indicates the receive buffer length. When reception is completed, the receive frame length is written back.

(3) Receive Descriptor 2 (RD2)

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RBA	Receive Buffer Address	These bits indicate the start address of the receive buffer. Align the buffer address on a 32-byte boundary.	R/W

RD2 indicates the start address of the receive buffer.

29.3.2 Transmission

When setting the EDTRR.TR bit to 1 while the ETHERCn.ECMR.TE bit is 1, the EDMAC reads the descriptor following the previously used descriptor in the transmit descriptor list (or the descriptor indicated by the TDLAR register after a reset). When the TACT bit is 1 in the transmit descriptor (TD0), the EDMAC sequentially reads transmit data from the start address of the transmit buffer indicated by transmit descriptor 2 (TD2) and transfers them to the ETHERC via the transmit FIFO. The ETHERC creates a transmit frame and starts transmission to the MII/RMII. When all data indicated by the TD1.TBL bit is transferred, write-back is performed according to the TD0.TFP[1:0] bits as follows:

•When the TD0.TFP[1:0] bits are 00b or 10b (frame information is incomplete), the TD0.TACT bit is written back.

•When the TD0.TFP[1:0] bits are 01b or 11b (frame information is complete), bits TD0.TACT, TD0.TFS, and TD0.TFE are written back.

When the TD0.TACT bit in the read descriptor is 1, the EDMAC continues reading descriptors and transmit frames. When the TD0.TACT bit in the read descriptor is 0, the EDMAC sets the EDTRR.TR bit to 0 and stops transmission.

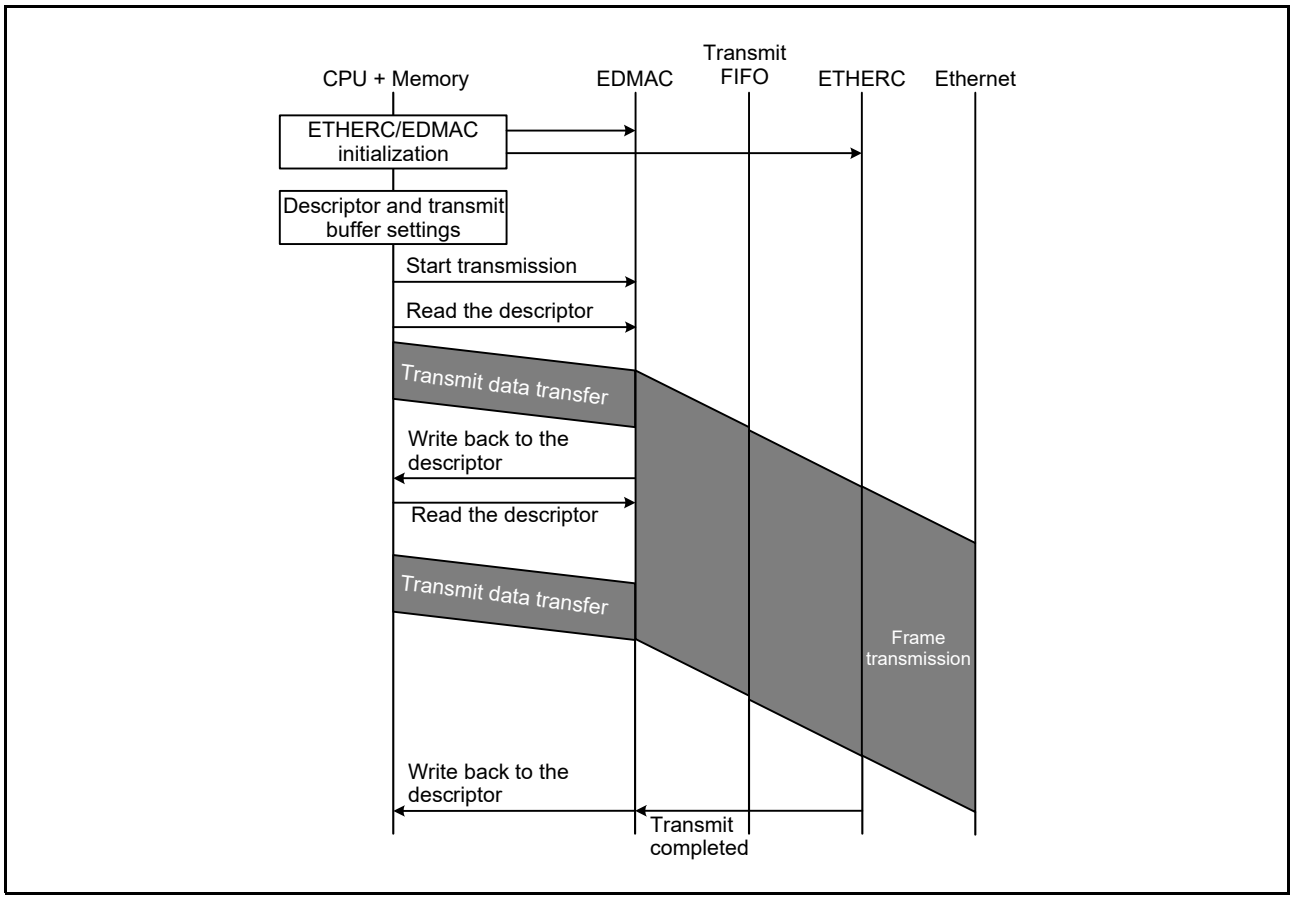


Figure 29.5 Example of Transmission Flow

29.3.3 Reception

When setting the EDRRR.RR bit to 1 while the ETHERCn.ECMR.RE bit is 1, the EDMAC reads the receive descriptor following the previously used descriptor (or descriptor indicated by the RDLAR register after a reset) and then waits for reception. While the RD0.RACT bit is 1, if the data stored to the receive FIFO is 32 bytes or more, or if the end byte of the frame is stored in the receive buffer, the EDMAC transfers data from the receive FIFO to the receive buffer indicated by receive descriptor 2 (RD2). If the data length of the received frame is longer than the buffer length set by the RBL bit in receive descriptor 1 (RD1), the EDMAC writes back 10b or 00b to the RD0.RFP[1:0] bits and 0 to the RD0.RACT bit when the receive buffer becomes full, and then the EDMAC reads the next data. After that, the EDMAC transfers data to another receive buffer. When the frame reception is completed or when the frame reception is aborted by an error, the EDMAC writes back 11b or 01b to the RD0.RFP[1:0] bits, 0 to the RD0.RACT bit, and the receive frame length to the RD1.RFL bit.*1 When the RMCR.RNR bit is 1, the EDMAC reads the next descriptor and waits for reception. When the RNR bit is 0, the EDMAC sets the EDRRR.RR bit to 0 and stops reception.

Note 1. If the reception is interrupted when the data stored in the FIFO is less than 16 times, the data in the FIFO are discarded without being transferred.
At this time, the corresponding error flag does not also set to 1.

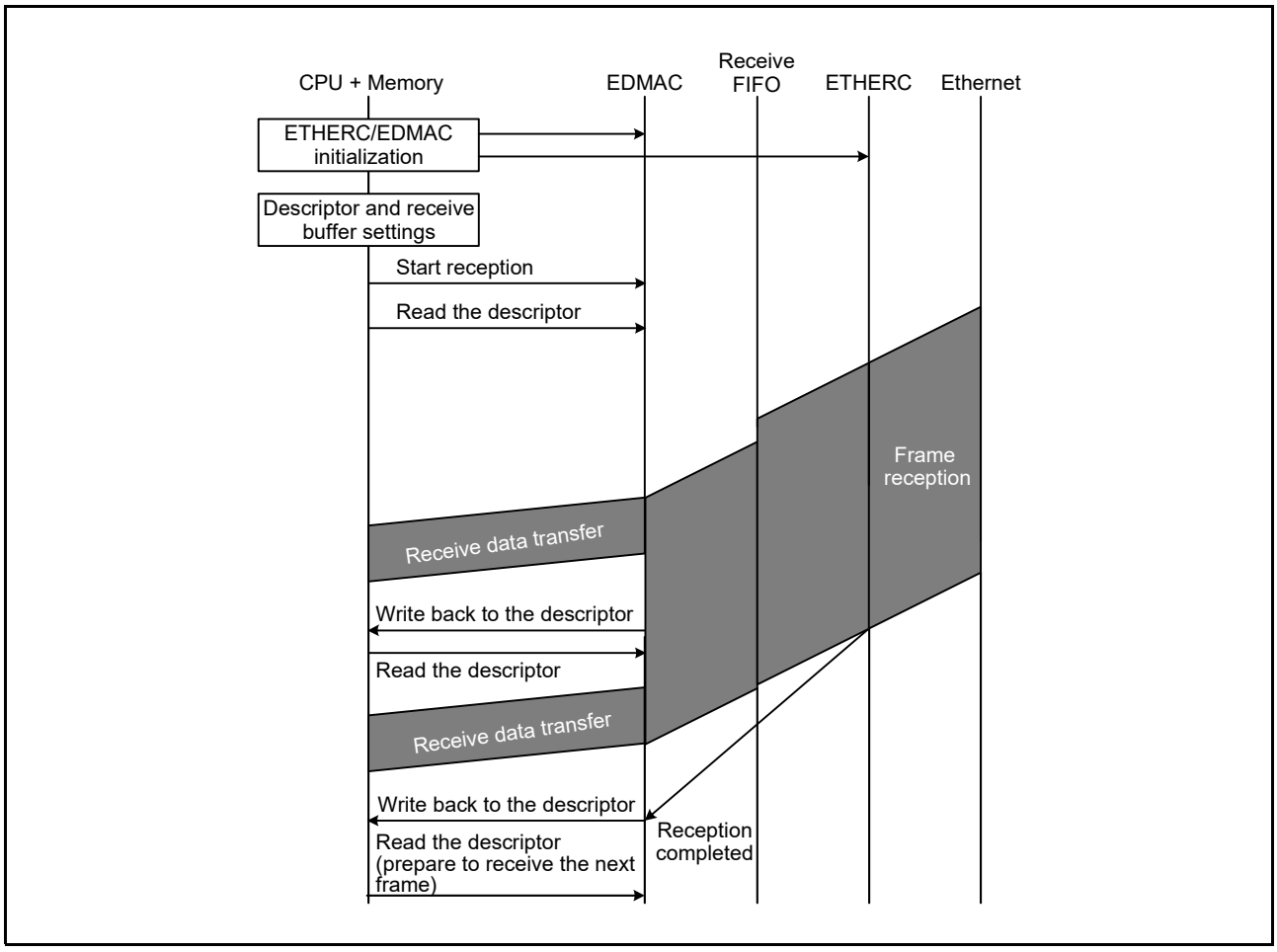


Figure 29.6 Example of Reception Flow

29.3.4 Multi-Buffer Frame Transmission

29.3.4.1 Error Processing While Transmitting a Multi-Buffer Frame

If an error occurs during multi-buffer frame transmission, the EDMAC performs the processing shown in Figure 29.7. In the figure, when the TACT bit of transmit descriptor 0 (TD0) is 0, the descriptor indicates that all data in the buffer has been successfully transmitted. When the TACT bit is 1, the descriptor indicates that data in the buffer has not yet been transmitted. If a frame transmit error *1 occurs in the head or middle of the frame while the TD0.TACT bit is 1, the EDMAC stops data transmission from the transmit FIFO and EDMAC data transfer, and sets the TD0.TACT bit to 0. After that, the EDMAC reads the next descriptor to see if the descriptor indicates the middle of the fame (TD0.TFP[1:0] bits are 00b) or the end of the frame (TD0.TFP[1:0] bits are 01b). When the descriptor indicates the middle of the frame, the EDMAC sets the TD0.TACT bit to 0 and reads the next descriptor. When the descriptor indicates the end of the frame, the EDMAC not only sets the TD0.TACT bit to 0, but also writes back to bits TD0.TFE and TD0.TFS. After an error occurs, data in the buffer is not transmitted until write-back to the descriptor for the end of the frame. When the corresponding transmit error interrupt is enabled in the EESIPR register, an interrupt request is generated immediately after write-back to the descriptor for the end of the frame.

Note 1. For EDMACn, a transmit retry-over condition, late collision, or loss of carrier is detected, or a carrier is not detected. For the PTPEDMAC, the MAC address does not match the set value.

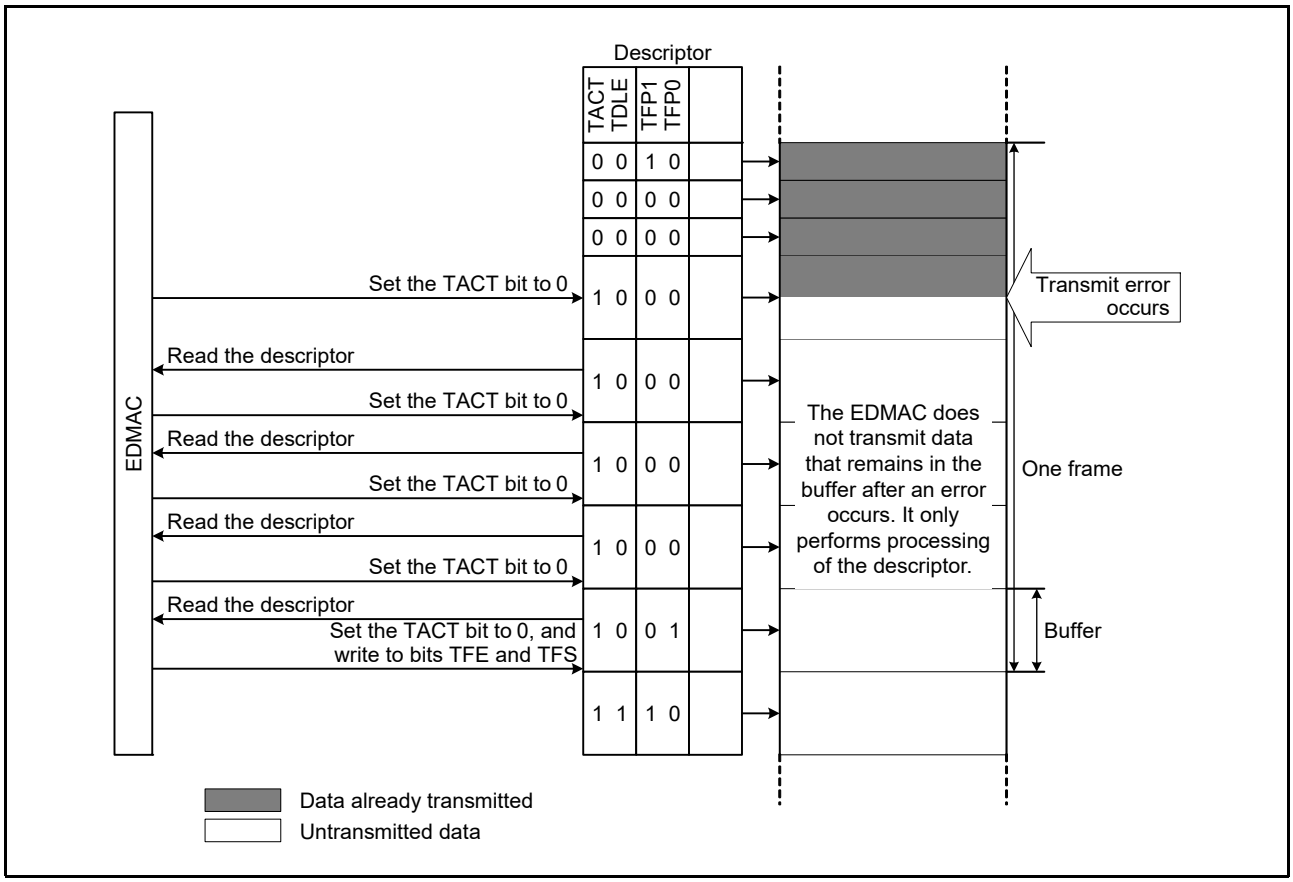


Figure 29.7 EDMAC Operation After a Transmit Error Occurs

29.3.4.2 Error Processing While Receiving a Multi-Buffer Frame

If an error occurs during multi-buffer frame reception, the EDMAC performs the processing shown in Figure 29.8 In the figure, when the RACT bit of receive descriptor 0 (RD0) is 0, the descriptor indicates that data has been successfully received in the buffer. When the RACT bit is 1, the descriptor indicates that data has not yet been received in the buffer. If a frame receive error*1 occurs, the EDMAC stops receiving data of the frame, but it transfers data that has already been stored in the receive FIFO to the receive buffer.

When the receive buffer becomes full during transfer, the EDMAC sets the RACT bit to 0 and the RFP[1:0] bits to 10b or 00b and reads the next descriptor. After all data in the receive FIFO has been transferred, the EDMAC writes back the status to the descriptor.

When the corresponding receive error interrupt is enabled in the EESIPR register, an interrupt request is generated immediately after write-back to the descriptor. When there is a request to receive a new frame, the EDMAC continues reception using the descriptor following the descriptor where the error occurred.

Note 1. For EDMACn, a CRC error, PHY-LSI receive error, frame-too-short error, frame-too-long error, or alignment error is detected. For the PTPEDMAC, a parity error is detected.

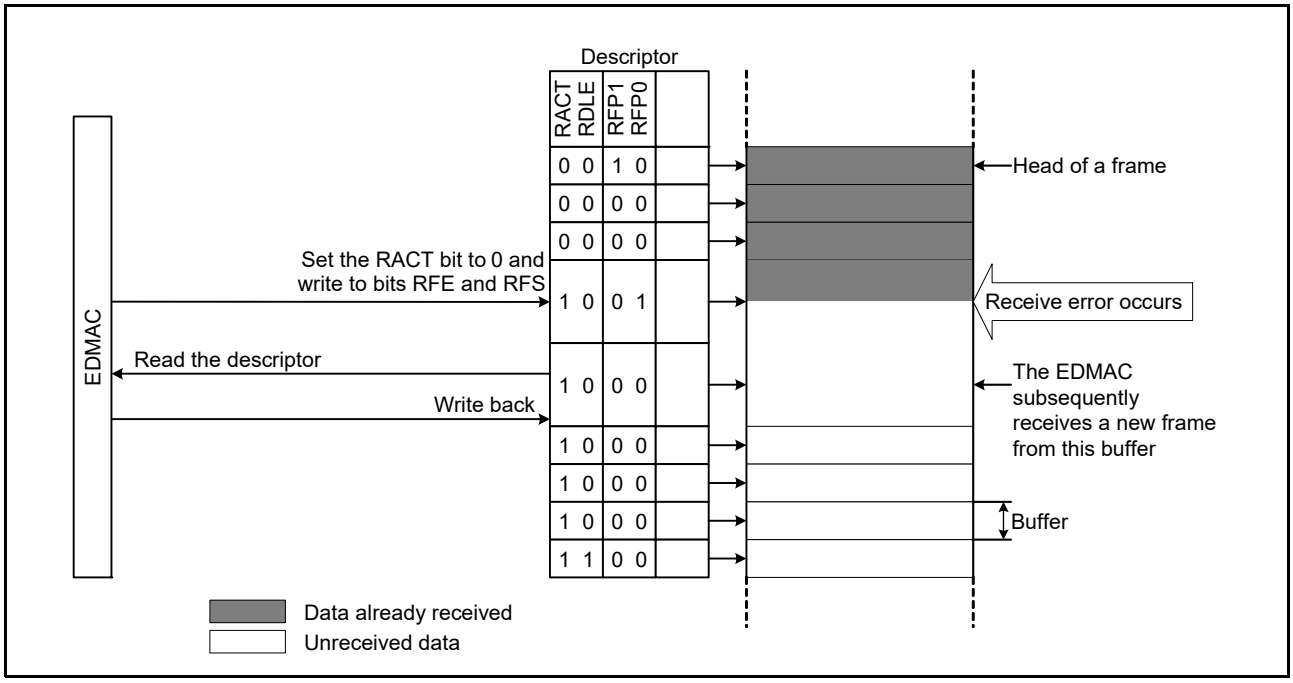


Figure 29.8 EDMAC Operation After a Receive Error Occurs

29.3.5 EDMAC Channel Priority

Priority of the three EDMAC channels (EDMAC0, EDMAC1, PTPEDMAC) is determined in a round-robin fashion. Each time transfer of one channel is completed, the channel becomes the lowest priority. This operation is shown in Figure 29.9. After a reset, the priority is EDMAC0 > EDMAC1 > PTPEDMAC.

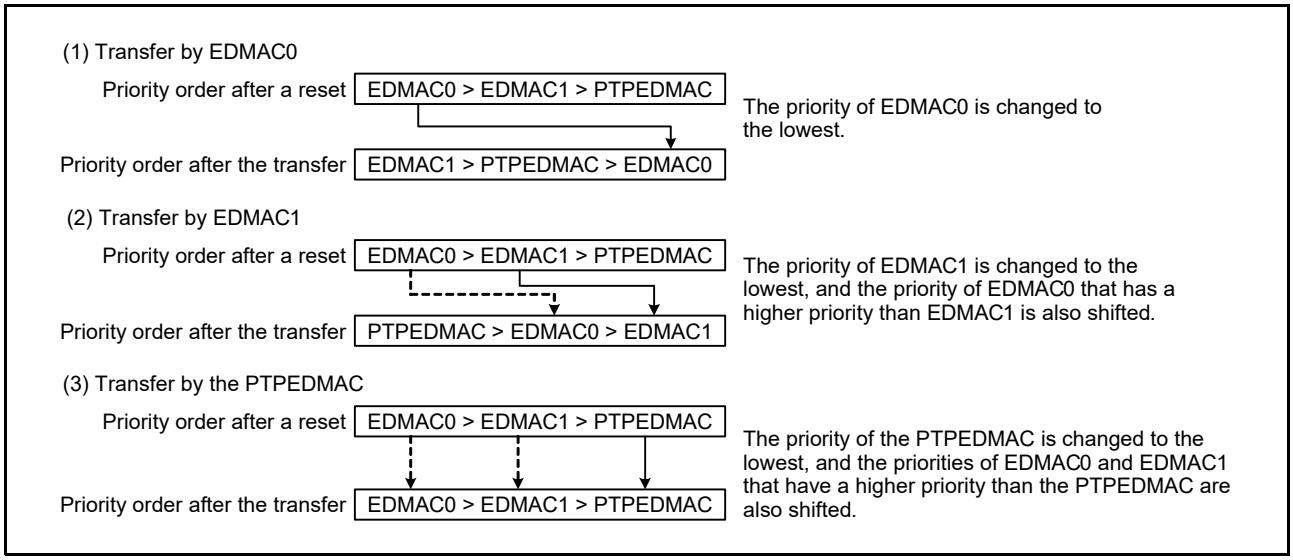


Figure 29.9 Operation in Round-Robin Fashion

Figure 29.10 shows change of the channel priority order when transfer requests are concurrently generated to EDMAC0 and the PTPEDMAC and a transfer request to the EDMAC1 while EDMAC0 is transferring data.

1. Transfer requests are concurrently generated to EDMAC0 and the PTPEDMAC.
2. Since EDMAC0 has a higher priority than the PTPEDMAC, EDMAC0 starts a transfer (PTPEDMAC waits to transfer).
3. A transfer request is generated to EDMAC1 during the transfer by EDMAC0 (EDMAC1 and the PTPEDMAC wait to transfer).
4. After EDMAC0 ends the transfer, the priority of EDMAC0 is changed to the lowest.
5. Since EDMAC1 has higher priority than the PTPEDMAC at this time, EDMAC1 starts a transfer (PTPEDMAC waits to transfer).
6. After EDMAC1 ends the transfer, the priority of EDMAC1 is changed to the lowest.
7. The PTPEDMAC starts a transfer.
8. After the PTPEDMAC ends the transfer, the priority of PTPEDMAC is changed to the lowest.

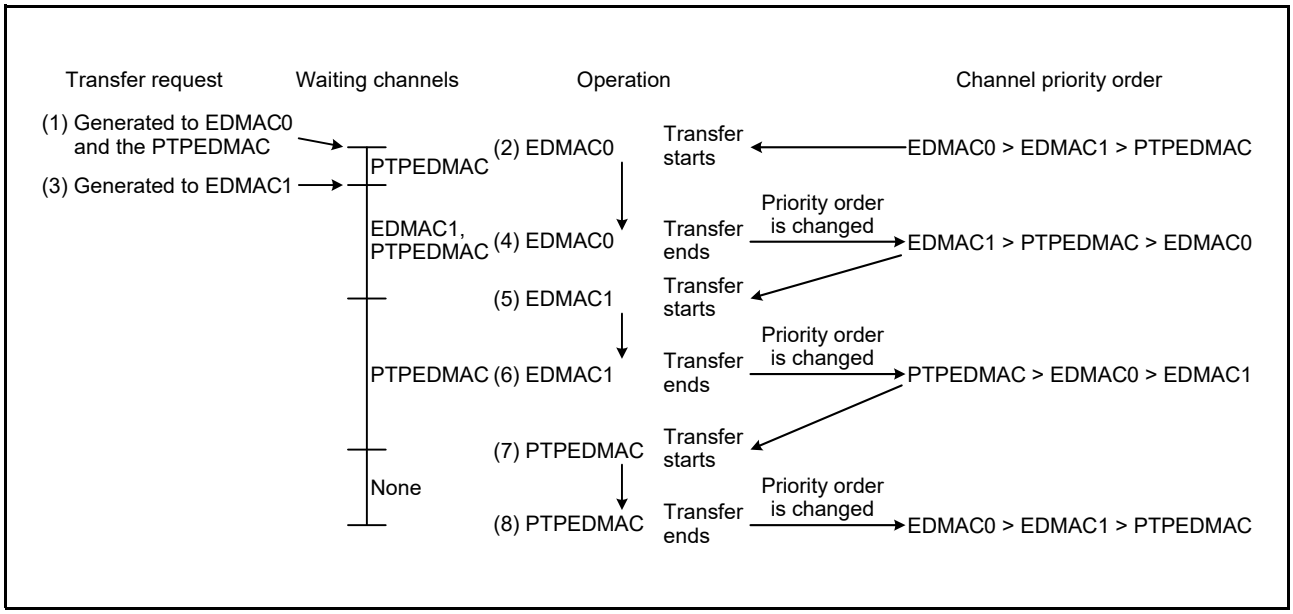


Figure 29.10 Channel Priority in Round-Robin Fashion

29.4 Interrupts

When any of the status flags in the EESR register becomes 1 while the corresponding interrupt request enable bit in the EESIPR register is 1, EDMACn outputs an EINTn interrupt request or the PTPEDMAC outputs a PINT interrupt request to the CPU.

29.5 Usage Notes

29.5.1 Setting the Module-Stop Function

The following bits in Standby Control Register 6 (STBCR6) are used to enable/disable the EDMAC operation.

- MSTP65 bit is used to enable/disable the ETHERC0 and the EDMAC0 operation.
- MSTP64 bit is used to enable/disable the ETHERC1 and the EDMAC1 operation.
- MSTP63 bit is used to enable/disable the EPTPC and the PTPEDMAC operation.

After a reset, the EDMAC is stopped. After exiting the module-stop state, registers can be accessed. Refer to section 52, Power-Down Modes for details.

29.5.2 Stopping the EDMAC during Operations

When stopping EDMAC operation by using a sleep instruction or module-stop function while the EDMAC is running, confirm that bits EDTRR.TR and EDRRR.RR are 0. If the EDMAC is stopped while the EDTRR.TR or EDRRR.RR bit 1, the data for the frame that is being transmitted or received may not be complete, and EDMAC operation after exiting the sleep mode or module-stop state is not guaranteed.

30. A/D Converter

30.1 Overview

This LSI incorporates one unit of a 12-bit successive approximation A/D converter, up to eight analog input channels are selectable.
The A/D conversion accuracy is selectable from 12-bit conversion, and conversion to a digital value can be performed with the relationship between speed and resolution optimized.
The A/D converter has three operating modes: single scan mode in which the analog inputs of up to 8 channels arbitrarily selected are converted for only once in ascending channel order; continuous scan mode in which the analog inputs of up to 8 channels arbitrarily selected are continuously converted in ascending channel order; and group scan mode in which up to 8 channels of the analog inputs are arbitrarily divided into two groups (group A and group B) or three groups (group A, group B, and group C) and converted in ascending channel order in each group.
In group scan mode, either of two groups (group A, B) and three groups (groups A, B, C) must be selected.
The scan start conditions (trigger) for each group (A, B or A, B, C) can be independently selected, and scan for each group (A, B or A, B, C) can be started at different timing.
In group priority control in group scan mode, along with operation as described above, if a request to start scan for a priority group is accepted during scanning for a low priority group, the scanning for the low priority group is discontinued and scan for the priority group is started.
The priority order of group priority control is group A > group B > group C.
In group priority control, if a request to start group B scan is accepted during group C scanning, the group C scanning is interrupted and the group B scan is started. If a request to start group A scan is accepted during group C scanning, the group C scanning is interrupted and group A scan is started. Similarly, if a request to start group A scan is accepted during group B scanning, the group B scanning is interrupted and group A scan is started.
Additionally, scan for the interrupted group can be restarted after the scan for the priority group has completed.
In double trigger mode, one analog input channel arbitrarily selected is converted in single scan mode or group scan mode (group A), and the resulting data of A/D conversion started by the first and second synchronous triggers are stored into different registers (duplication of A/D conversion data).
Self-diagnosis is executed once at the beginning of each scan, and one of the three voltages internally generated in the A/D converter is converted.

The compare function specifies the upper-side reference value for window A and lower-side reference value for window B and outputs an interrupt when the A/D-converted value of the selected channel meets the comparison conditions. Furthermore, the comparator operation to compare the A/D-converted value with the lower-side reference value is also available.

Table 30.1 lists the specifications of the A/D converter and Table 30.2 indicates the functions of the A/D converter.

Table 30.1 Specifications of A/D Converter

Item	Specifications
Input channels	Eight channels
A/D conversion method	Successive approximation method
Resolution	12 bits / 10 bits / 8 bits selectable
Conversion time	1 µs per channel (when A/D conversion clock ADCLK = 33 MHz)
A/D conversion clock	Peripheral module clock PCLK (= P1φ) and A/D conversion clock ADCLK (= P0φ) can be set so that the frequency division ratio should be one of the following. PCLK to ADCLK frequency division ratio = 1:1, 1:2 Each clock is set using the clock pulse generator.
Data registers	• 8 registers for analog input, 1 for A/D-converted data duplication in double trigger mode, 2 for A/D-converted data duplication during extended operation in double trigger mode, and 1 register for self-diagnosis. • The results of A/D conversion are stored in the A/D data registers. • 8-, 10-, and 12-bit accuracy conversion for the results of A/D conversion • The value obtained by adding up A/D-converted results is stored as a value in the number of bit for conversion accuracy + 2 bits*3 in the A/D data registers in A/D-converted value addition mode. • Double trigger mode (selectable in single scan and group scan modes): The first piece of A/D-converted analog-input data on one selected channel is stored in the data register for the channel, and the second piece is stored in the duplication register. • Extended operation in double trigger mode (available for specific triggers): A/D-converted analog-input data on one selected channel is stored in the duplication register that is prepared for each type of trigger.
Operating modes	• Single scan mode: - A/D conversion is performed only once on the analog inputs of up to 8 channels arbitrarily selected. • Continuous scan mode: - A/D conversion is performed repeatedly on the analog inputs of up to 8 channels arbitrarily selected. • Group scan mode: - Two (group A, B) or three (group A, B, C) can be selected for the number of groups to be used. (When the number of groups is two, only the combination of group A and group B can be selected.) - Analog inputs of up to 8 channels arbitrarily selected are divided into group A and group B, or group A, group B, and group C, and A/D conversion of the analog input selected on a group basis is performed only once. - The conditions for scanning start of group A, group B and group C (trigger) can be independently selected, thus allowing A/D conversion of group A, group B, and group C to be started independently. Group scan mode (when group priority is selected) : - If a trigger for a priority group occurs during scanning for a low priority group, the scanning for the low priority group is interrupted and scan for the priority group is started. The priority order is group A (high) > group B > group C (low). - Whether to re-execute (rescan) the scan for the low priority group after the scan for the priority group is completed can be set. Additionally, whether to start rescan from the head of the selected channels or from the channel on which A/D conversion has not finished can be selected.
Conditions for A/D conversion start	• Software trigger • Synchronous trigger Trigger by the multi-function timer pulse unit 3 (MTU3a) and the general-purpose PWM timer (GPT). • Asynchronous trigger A/D conversion can be triggered by the ADTRG# external trigger pin.
Function	• Variable sampling state count (which can be set for each channel) • Self-diagnosis of A/D converter • Selectable A/D-converted value addition mode or average mode • Analog input disconnection detection function (Precharge function/discharge function) • Double trigger mode (duplication of A/D conversion data) • Switching function of 8-, 10-, and 12-bit conversion • Automatic clear function of A/D data registers • Compare Function (Window-A and Window-B)

Item	Specifications
Interrupt source	• In the modes except double trigger mode and group scan mode, A/D scan end interrupt (S12ADI0) request can be generated on completion of single scan. • In double trigger mode, A/D scan end interrupt (S12ADI0) request can be generated on completion of double scan. • In group scan mode: - An A/D scan end interrupt (S12ADI0) request can be generated on completion of group A scan. - An A/D scan end interrupt for group B (S12GBADI0) request can be generated on completion of group B scan. - An A/D scan end interrupt for group C (S12GCADI0) request can be generated on completion of group C scan. • When double trigger mode is selected in group scan mode (including extended operation in double trigger mode), A/D scan end interrupt (S12ADI0) request can be generated on completion of double scan of group A, whereas A/D scan end interrupt especially for group B and group C (S12GBADI0 and S12GCADI0) request can be generated on completion of group B and group C scan, respectively. (pulse output) • When the window A/B comparison conditions are met, S12ADCMPAI0/S12ADCMPBI0 is generated. • The S12ADI0, S12GBADI0 or S12GCADI0 interrupt can activate the DMA controller (DMAC).
Low power consumption function	• Module standby mode, software standby mode, and deep standby mode can be specified.*1,*2

Note 1. See section 52, Power-Down Modes for details.
Note 2. After canceling module standby mode, wait for at least 1 µs before starting A/D conversion.
Note 3. The number of extended bits for addition varies with the A/D conversion accuracy and the addition count.
2-bit extension: 4-time conversion (3-time addition) with the A/D conversion accuracy = 8/10/12 bits.
4-bit extension: 16-time conversion (15-time addition) with the A/D conversion accuracy = 12 bits.

Table 30.2 Functions of A/D Converter

Item				Unit 0 (S12ADB0)
Analog input channel				AN000 to AN007
Conditions for A/D conversion start	Software	Software trigger		Enabled
	Asynchronous trigger	External trigger*1	Trigger input pin	ADTRG#
	Synchronous trigger	Trigger from MTU3a	Compare match with or input capture to MTU0.TGRA	TRGA0N
			Compare match with or input capture to MTU1.TGRA	TRGA1N
			Compare match with or input capture to MTU2.TGRA	TRGA2N
			Compare match with or input capture to MTU3.TGRA	TRGA3N
			Compare match with or input capture to MTU4.TGRA or underflow (trough) of MTU4.TCNT in complementary PWM mode	TRGA4N
			Compare match with or input capture to MTU6.TGRA	TRGA6N
			Compare match with or input capture to MTU7.TGRA or underflow (trough) of MTU7.TCNT in complementary PWM mode	TRGA7N
			Compare match with MTU0.TGRE	TRG0N
			Compare match between MTU4.TADCORA and MTU4.TCNT	TRG4AN
			Compare match between MTU4.TADCORB and MTU4.TCNT	TRG4BN
			Compare match between MTU4.TADCORA and MTU4.TCNT or compare match between MTU4.TADCORB and MTU4.TCNT	TRG4AN or TRG4BN
			Compare match between MTU4.TADCORA and MTU4.TCNT and compare match between MTU4.TADCORB and MTU4.TCNT (when interrupt skipping function 2 is used)	TRG4ABN
			Compare match between MTU7.TADCORA and MTU7.TCNT	TRG7AN
			Compare match between MTU7.TADCORB and MTU7.TCNT	TRG7BN
			Compare match between MTU7.TADCORA and MTU7.TCNT or compare match between MTU7.TADCORB and MTU7.TCNT	TRG7AN or TRG7BN
			Compare match between MTU7.TADCORA and MTU7.TCNT and compare match between MTU7.TADCORB and MTU7.TCNT (when interrupt skipping function 2 is used)	TRG7ABN
Conditions for A/D conversion start	Synchronous trigger	Trigger from GPT	Compare match with GTP0.GTADTRA	ADTRGA0
			Compare match with GPT0.GTADTRB	ADTRGB0
			Compare match with GPT1.GTADTRA	ADTRGA1
			Compare match with GPT1.GTADTRB	ADTRGB1
			Compare match with GPT2.GTADTRA	ADTRGA2
			Compare match with GPT2.GTADTRB	ADTRGB2
			Compare match with GPT3.GTADTRA	ADTRGA3
			Compare match with GPT3.GTADTRB	ADTRGB3
			Compare match with GTP0.GTADTRA or compare match with GTP0.GTADTRB	ADTRGA0 or ADTRGB0
			Compare match with GTP1.GTADTRA or compare match with GTP1.GTADTRB	ADTRGA1 or ADTRGB1
			Compare match with GTP2.GTADTRA or compare match with GTP2.GTADTRB	ADTRGA2 or ADTRGB2
			Compare match with GTP3.GTADTRA or compare match with GTP3.GTADTRB	ADTRGA3 or ADTRGB3
Interrupt				S12ADI0 S12GBADI0 S12GCADI0 S12ADCMPAI0 S12ADCMPBI0

Note 1. To set ADTRG# as the trigger to start A/D conversion, set the general I/O function. See section 51, GPIO for details.

Table 30.3 lists the I/O pins of the A/D converter.

Table 30.3 I/O Pins of A/D Converter

Pin Name	I/O	Name	Function
AN007 to AN000	Input	Analog input pins	Analog input pin 0 to 7
ADTRG#	Input	A/D conversion trigger input	External trigger input pin for starting A/D conversion
AVcc	Input	Analog power supply / Analog reference voltage	A/D converter power supply and Analog reference voltage supply pin
AVss	Input	Analog ground	A/D converter ground pin

30.2 Register Descriptions

Table 30.4 shows the register list of this module.

Table 30.4 List of registers of this A/D Converter

Name	Symbol	Value after reset	Address	Access size
A/D Control Register	ADCSR	0000h	E800_5800h	16
A/D Channel Select Register A0	ADANSA0	0000h	E800_5804h	16
A/D-Converted Value Addition/Average Mode Select Register 0	ADADS0	0000h	E800_5808h	16
A/D-Converted Value Addition/Average Count Select Register	ADADC	00h	E800_580Ch	8
A/D Control Extended Register	ADCER	0000h	E800_580Eh	16
A/D Start Trigger Select Register	ADSTRGR	0000h	E800_5810h	16
A/D Channel Select Register B0	ADANSB0	0000h	E800_5814h	16
A/D Data Duplication Register	ADDBLDR	0000h	E800_5818h	16
A/D Self-Diagnosis Data Register	ADRD	0000h	E800_581Eh	16
A/D Data Register 0	ADDR0	0000h	E800_5820h	16
A/D Data Register 1	ADDR1	0000h	E800_5822h	16
A/D Data Register 2	ADDR2	0000h	E800_5824h	16
A/D Data Register 3	ADDR3	0000h	E800_5826h	16
A/D Data Register 4	ADDR4	0000h	E800_5828h	16
A/D Data Register 5	ADDR5	0000h	E800_582Ah	16
A/D Data Register 6	ADDR6	0000h	E800_582Ch	16
A/D Data Register 7	ADDR7	0000h	E800_582Eh	16
A/D Disconnection Detection Control Register	ADDISCR	00h	E800_587Ah	8
A/D Group Scan Priority Control Register	ADGSPCR	0000h	E800_5880h	16
A/D Data Duplication Register A	ADDBLDRA	0000h	E800_5884h	16
A/D Data Duplication Register B	ADDBLDRB	0000h	E800_5886h	16
A/D Compare Function AB Status Monitor Register	ADWINMON	00h	E800_588Ch	8
A/D Compare Control Register	ADCMPCR	0000h	E800_5890h	16
A/D Compare Function Window-A Channel Selection Register 0	ADCMPANSR0	0000h	E800_5894h	16
A/D Compare Function Window-A Comparison Condition Setting Register 0	ADCMPLR0	0000h	E800_5898h	16
A/D Compare Function Window-A Lower Level Setting Register	ADCMPDR0	0000h	E800_589Ch	16
A/D Compare Function Window-A Upper Level Setting Register	ADCMPDR1	0000h	E800_589Eh	16
A/D Compare Function Window-A Channel Status Register 0	ADCMPSR0	0000h	E800_58A0h	16
A/D Compare Function Window-B Channel Selection Register	ADCMPBNSR	00h	E800_58A6h	8
A/D Compare Function Window-B Lower Level Setting Register	ADWINLLB	0000h	E800_58A8h	16
A/D Compare Function Window-B Upper Level Setting Register	ADWINULB	0000h	E800_58AAh	16
A/D Compare Function Window-B Status Register	ADCMPBSR	00h	E800_58ACh	8
A/D Channel Select Register C0	ADANSC0	0000h	E800_58D4h	16
A/D Group C Trigger Select Register	ADGCTRGR	00h	E800_58D9h	8
A/D Sampling State Register 0	ADSSTR0	0Bh	E800_58E0h	8
A/D Sampling State Register 1	ADSSTR1	0Bh	E800_58E1h	8
A/D Sampling State Register 2	ADSSTR2	0Bh	E800_58E2h	8
A/D Sampling State Register 3	ADSSTR3	0Bh	E800_58E3h	8
A/D Sampling State Register 4	ADSSTR4	0Bh	E800_58E4h	8
A/D Sampling State Register 5	ADSSTR5	0Bh	E800_58E5h	8
A/D Sampling State Register 6	ADSSTR6	0Bh	E800_58E6h	8
A/D Sampling State Register 7	ADSSTR7	0Bh	E800_58E7h	8

30.2.1 A/D Data Register y (ADDRy: y = 0 to 7), A/D Data Duplication Register (ADDBLDR), A/D Data Duplication Register A (ADDBLDRA), A/D Data Duplication Register B (ADDBLDRB)

The ADDRy registers (y = 0 to 7) are 16-bit read-only registers for storing the result of A/D conversion. Register ADDBLDR is a 16-bit read-only register for storing the result of A/D conversion in response to the second trigger in double trigger mode. Registers ADDBLDRA and ADDBLDRB are 16-bit read-only registers for storing the result of A/D conversion during extended operation in double trigger mode.
The formats for data in the ADDRy, ADDBLDR, ADDBLDRA, and ADDBLDRB registers vary according to the following conditions.

- The setting of the A/D data register format select bit (determining whether the data are flush-left or flush-right in the registers)
- The setting of the A/D conversion-accuracy selection bits (8-, 10-, or 12-bit)
- The setting of the A/D-Converted Value Addition/Average Count Select Register (A/D conversion count, A/D converted value average mode is selected or not selected)

The data formats (section 30.2.23, Data Format, (1) to (9)) for each given condition are shown below.

Table 30.5 Correspondence between the Addition Count and the Setting Condition

A/D conversion accuracy	Format specification	Addition/ Average deselected	Conversion count when addition is selected (addition count)					Conversion count when average is selected	
			1 (0) time	2 (1) times	3 (2) times	4 (3) times	16 (15) times	2 times	4 times
8-bit	Right-aligned	(1)	(3)*1	(3)*1	(3) *1	(3)*1		(1)	(1)
	Left-aligned	(2)	(4)*1	(4)*1	(4)*1	(4)*1		(2)	(2)
10-bit	Right-aligned	(3)	(5)*1	(5)*1	(5)*1	(5)*1		(3)	(3)
	Left-aligned	(4)	(6)*1	(6)*1	(6)*1	(6)*1		(4)	(4)
12-bit	Right-aligned	(5)	(7)*1	(7)*1	(7)*1	(7)*1	(9)*2	(5)	(5)
	Left-aligned	(6)	(8)*1	(8)*1	(8)*1	(8)*1	(9)*2	(6)	(6)

Note 1. Only when 1 (0) to 4 (3) times is specified as the conversion count (addition count) in addition mode, the A/D conversion addition result is 2-bit-extended and stored in the A/D data register.
The explanation about this is also described in "When A/D-converted value addition mode is selected".
Note 2. When addition is selected and 16-time conversion (15-time addition) is specified with 12-bit A/D conversion accuracy, the A/D conversion addition result is 4-bit-extended and stored in the A/D data register.

For details about this register format, see section 30.2.23, Data Format, (1) to (9). In that section, see the description as follows:

- Replace bit symbol "DATA[*:0]" with "AD[*:0]".
- "Function" should be n-bit A/D-converted value. (n = 8, 10, 12, 14 to 16)
- The R/W attribute should be R.

When A/D-converted value average mode is selected

A/D-converted value average mode can be selected only when 2 or 4 times is specified in A/D-converted value addition mode. When A/D-converted value average mode is set, the average of A/D conversion results is retained in the A/D data register. Even if A/D-converted value average mode is selected, the value is stored in the A/D data register according to the settings of the A/D data register format select bit in the same way as normal A/D conversion.

When A/D-converted value addition mode is selected

When the bit accuracy is set to 12-bit accuracy, 1, 2, or 4 times can be specified in A/D-converted value addition mode. Additionally, only when the bit accuracy is set to 12-bit accuracy, 16 times can be specified in A/D-converted value addition mode.
In A/D-converted value addition mode, the value obtained by adding up of A/D conversion results is retained in the A/D data register as a bit-extended value of the conversion accuracy specified.
When 12-bit A/D conversion accuracy is selected and 1 to 4 times is specified in addition mode, 2-bit extension is performed. When 12-bit A/D conversion accuracy is selected and 16 times is specified in addition mode, 4-bit extension is performed.
Even if A/D-converted value addition mode is selected, the value is stored in the A/D data register according to the settings of the A/D data register format select bits.

30.2.2 A/D Self-Diagnosis Data Register (ADRD)

ADRD is a 16-bit read-only register that holds the A/D conversion results based on the A/D converter's self-diagnosis. In addition to the AD bit indicating A/D-converted value, the self-diagnosis status bit (DIAGST) is included in. In the ADRD register, the following different formats are used depending on the conditions below.

- The setting of the A/D data register format select bit (determining whether the data are flush-left or flush-right in the registers)
- The setting of the A/D conversion-accuracy selection bits (ADCER.ADPRC[1:0]) (8-, 10-, or 12-bit)

The A/D-converted value addition mode and A/D-converted value average mode cannot be applied to the A/D self-diagnosis function. For details of self-diagnosis, see section 30.2.9, A/D Control Extended Register (ADCER).

The settings for flush-right data

DIAGST[1:0] is allocated to the upper two bits (b15, b14). The A/D conversion result is allocated to the bit position b13 to b0 depending on the setting of the A/D conversion accuracy specify bits.

A/D Conversion Accuracy Specify Bits = 8-bit accuracy

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DIAGST[1:0]		—	—	—	—	—	—	AD[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

A/D Conversion Accuracy Specify Bits = 10-bit accuracy

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DIAGST[1:0]		—	—	—	—	AD[9:0]									
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

A/D Conversion Accuracy Specify Bits = 12-bit accuracy

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DIAGST[1:0]		—	—	AD[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b13 to b0	AD[11**:0]	—	The data width varies depending on the setting of the A/D conversion accuracy specify bits. 8-bit: AD[7:0] is stored in b7 to b0. 10-bit: AD[9:0] is stored in b9 to b0. 12-bit: AD[11:0] is stored in b11 to b0.	R
	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b15, b14	DIAGST[1:0]	Self-Diagnosis Status	b15 b14 0 0: Self-diagnosis has never been executed since power-on. 0 1: Self-diagnosis using the voltage of 0 V has been executed. 1 0: Self-diagnosis using the voltage of reference power supply × 1/2 has been executed. 1 1: Self-diagnosis using the voltage of reference power supply has been executed.	R

The settings for flush-left data

DIAGST[1:0] is allocated to the lower two bits (b1, b0). The A/D conversion result is allocated to the bit position b15 to b2 depending on the setting of the A/D conversion accuracy specify bits.

A/D Conversion Accuracy Specify Bits = 12-bit accuracy

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD[11:0]												—	—	DIAGST[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b1, b0	DIAGST[1:0]	Self-Diagnosis Status	b1 b0 0 0: Self-diagnosis has never been executed since power-on. 0 1: Self-diagnosis using the voltage of 0 V has been executed. 1 0: Self-diagnosis using the voltage of reference power supply × 1/2 has been executed. 1 1: Self-diagnosis using the voltage of reference power supply has been executed.	R
b15 to b2	AD[11** :0]	—	The data width varies depending on the setting of the A/D conversion accuracy specify bits. 8-bit: AD[7:0] is stored in b15 to b8. 10-bit: AD[9:0] is stored in b15 to b6. 12-bit: AD[11:0] is stored in b15 to b4.	R
	—	Reserved	These bits are read as 0. The write value should be 0.	R0

30.2.3 A/D Control Register (ADCSR)

ADCSR sets A/D conversion control, mode/trigger/interrupt.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADST	ADCS[1:0]		ADIE	—	—	TRGE	EXTRG	DBLE	GBADIE	—	DBLANS[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b4 to b0	DBLANS [4:0]	A/D Conversion Data Duplication Channel Select Double Trigger Channel Select	These bits select one analog input channel for which to duplicate A/D conversion data. The setting is only effective while double trigger mode is selected. These bits select one analog input channel for double triggered operation. The setting is only effective while double trigger mode is selected.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R0
b6	GBADIE	Group B Scan End Interrupt Enable	0: Disables S12GBADI0 interrupt generation upon group B scan completion. 1: Enables S12GBADI0 interrupt generation upon group B scan completion.	R/W
b7	DBLE	Double Trigger Mode Select	0: Deselects double trigger mode. 1: Selects double trigger mode.	R/W
b8	EXTRG	Trigger Select*1	0: A/D conversion is started by a synchronous trigger (MTU, GPT). 1: A/D conversion is started by the asynchronous trigger (ADTRG#).	R/W
b9	TRGE	Trigger Start Enable	0: Disables A/D conversion to be started by the synchronous or asynchronous trigger. 1: Enables A/D conversion to be started by the synchronous or asynchronous trigger.	R/W
b11, b10	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b12	ADIE	Scan End Interrupt Enable	0: Disables S12ADI0 interrupt generation upon scan completion. 1: Enables S12ADI0 interrupt generation upon scan completion.	R/W
b14, b13	ADCS [1:0]	Scan Mode Select	00: Single scan mode 01: Group scan mode 10: Continuous scan mode 11: Setting prohibited	R/W
b15	ADST	A/D Conversion Start	0: Stops A/D conversion process 1: Starts A/D conversion process	R/W*2

Note 1. Starting A/D conversion using an external pin (asynchronous trigger)
After a high-level signal is input to the external pin (ADTRG#), write 1 to both the TRGE and EXTRG bits in ADCSR and change the signals of ADTRG# to the low level. Thus the falling edge of ADTRG# are detected and the scan conversion process is started. In this case, the pulse width of the low-level input must be at least 1.5 clock cycles of PCLK.
Note 2. See the following description of the ADST bit.

DBLANS[4:0] Bits (Double Trigger Channel Select)

The DBLANS[4:0] bits select one of the channels for A/D conversion data duplication in double trigger mode. The A/D conversion results of the analog input of the channel selected by the DBLANS[4:0] bits are stored into the A/D data register y when conversion is started by the first trigger, and into the A/D data duplication register when started by the second trigger. Table 30.6 shows selection of the channel for double triggered operation.

The A/D-converted value addition/average functions with double trigger mode become available by selecting the channel selected in the DBLANS[4:0] bits using the ADADS0 register. When double trigger mode is selected, the channels selected in the ADANSA0 register are invalid, and the channel selected in the DBLANS[4:0] bits is subject to A/D conversion.

When double trigger mode is selected in group scan mode, double trigger mode only operates on group A and does not operate on groups B and C.

Although in double trigger mode, multiple analog input channels cannot be selected for group A, they can be selected for groups B and C.

The DBLANS[4:0] bits should be set while the ADST bit is 0. (They should not be set simultaneously when 1 is written to the ADST bit.)

Table 30.6 Selection of the Channel on which to Duplicate A/D Conversion Data

DBLANS[4:0]	Duplication channel
5'd0	AN000
5'd1	AN001
5'd2	AN002
5'd3	AN003
5'd4	AN004
5'd5	AN005
5'd6	AN006
5'd7	AN007

GBADIE Bit (Group B Scan End Interrupt Enable)

The GBADIE bit sets whether to enable or disable group B scan end interrupt (S12GBADI0) in group scan mode.

DBLE Bit (Double Trigger Mode Select)

In double trigger mode, the following operations 1 and 2 are performed on condition that the scan is started by the synchronous trigger (MTU, GPT) selected in ADSTRGR.TRSA[5:0].
If the ADIE bit is set to 1, the scan end interrupt is output not upon completion of the first scan but upon completion of the second scan.
The first piece of A/D conversion data of the analog input selected in the DBLANS[4:0] bits is stored in the A/D data register y, and the second piece is stored in the A/D data duplication register.
When the DBLE bit is set to 1, channels selected in the ADANSA0 register are invalid. Double trigger mode is not selected when the DBLE bit is set to 0. If the DBLE bit is set to 1 again, the preceding operations 1 and 2 are performed with the first scan by the first trigger.
In continuous scan mode, double trigger mode should not be selected.
Software trigger should not be used in double trigger mode.
The DBLE bit should be set when the ADST bit set to 0.
(This bit should not be set simultaneously when 1 is written to the ADST bit.)

EXTRG Bit (Trigger Select)

The EXTRG bit selects the synchronous trigger or the asynchronous trigger as the trigger for starting A/D conversion.

TRGE Bit (Trigger Start Enable)

The TRGE bit sets whether to enable or disable starting of A/D conversion by the synchronous trigger and the asynchronous trigger.
This bit should be set to 1 in group scan mode.

ADIE Bit (Scan End Interrupt Enable)

The ADIE bit sets whether to enable or disable generation of the A/D scan end interrupt (S12ADI0) in group scan mode (except for group B, C scan).
With double trigger mode deselected, the S12ADI0 interrupt is generated after the first scan is completed if the ADIE bit is set to 1.
With double trigger mode selected, the S12ADI0 interrupt is generated after the second scan is completed if the ADIE bit is set to 1 as long as the scan is started by the synchronous trigger (MTU, GPT) selected by the ADSTRGR.TRSA[5:0] bits.

ADCS[1:0] Bits (Scan Mode Select)

The ADCS bit select the scan mode.
In single scan mode, A/D conversion is performed for the analog inputs of a maximum of eight channels selected with the ADANSA0 register in the ascending order of the channel number, and when one cycle of A/D conversion is completed for all the selected channels, the scan conversion is stopped.
In continuous scan mode, while the ADST bit in ADCSR is 1, A/D conversion is performed for the analog inputs of a maximum of eight channels selected with the ADANSA0 register in the ascending order of the channel number, and when one cycle of A/D conversion is completed for all the selected channels, A/D conversion is repeated from the first channel. If the ADST bit in ADCSR is set to 0 during continuous scan, A/D conversion is stopped even if scanning is in progress.
In group scan mode, A/D conversion is performed for the analog inputs (group A) of a maximum of eight channels selected with the ADANSA0 register in the ascending order of the channel number after scanning is started by the synchronous trigger (MTU, GPT) selected by the TRSA[5:0] bits in ADSTRGR, and when one cycle of A/D conversion is completed for all the selected channels, A/D conversion is stopped.
A/D conversion is also performed for the analog inputs (group B/group C) of a maximum of eight channels selected with the ADANSB0/ADANSC0 register in the ascending order of the channel number after scanning is started by the synchronous trigger (MTU, GPT) selected by the TRSB[5:0]/TRSC[5:0] bits in ADSTRGR, and when one cycle of A/D conversion is completed for all the selected channels, A/D conversion is stopped.
In group scan mode, different channels and triggers should be selected for group A, group B and group C.

To use two groups when group scan mode is enabled, group A and group B should be used (ADGCTRGR.GRCE = 1’b0). Likewise, to use three groups, group A, group B, and group C should be used (ADGCTRGR.GRCE = 1’b1).

The ADCS[1:0] bits should be set while the ADST bit is 0 (it should not be set simultaneously when 1 is written to the ADST bit.)

ADST Bit (A/D Conversion Start)

The ADST bit starts or stops A/D conversion process.
Before the ADST bit is set to 1, set the A/D conversion clock, the conversion mode, and conversion target analog input.

[Setting conditions]

The ADST bit is set to 1 if one of the following conditions is satisfied:

- 1 is written by software.
- The synchronous trigger (MTU, GPT) selected by the ADSTRGR.TRSA[5:0] bits is detected with ADCSR.EXTRG and ADCSR.TRGE bits being set to 0 and 1, respectively.
- The synchronous trigger (MTU, GPT) selected by the ADSTRGR.TRSB[5:0] bits is detected with the ADCSR.TRGE bit being set to 1 in group scan mode.
- The asynchronous trigger is detected with the ADCSR.TRGE and ADCSR.EXTRG bits being set to 1 and the ADSTRGR.TRSA[5:0] bits being set to 00000b.
- When group priority control operation mode is enabled (ADCSR.ADCS[1:0] = 01b and ADGSPCR.PGS = 1), the ADGSPCR.GBRP bit is set to 1, and A/D conversion for the lowest-priority group starts.

[Clearing conditions]

The ADST bit is cleared to 0 if one of the following conditions is satisfied:

- 0 is written by software.
- The A/D conversion of all the selected channels is completed in single scan mode.
- Group A scan is completed in group scan mode.
- Group B scan is completed in group scan mode.
- Group C scan is completed in group scan mode.
- When group priority control operation mode is enabled (ADCSR.ADCS[1:0] = 01b and ADGSPCR.PGS = 1), the ADGSPCR.GBRSCN bit is set to 1, and scan for the lowest-priority group started by trigger completes.

Note 1. When group priority control operation mode has been enabled (ADCSR.ADCS[1:0] = 01b and ADGSPCR.PGS = 1), do not set the ADST bit to 1.

Note 2. When group priority control operation mode has been enabled (ADCSR.ADCS[1:0] = 01b and ADGSPCR.PGS = 1), do not set the ADST bit to 0. When forcibly terminating A/D conversion, follow the procedure for clearing the ADST bit.

Note 3. When the single scan succession function is used (ADGSPCR.GBRP = 1) with group priority control operation mode enabled (ADCSR.ADCS[1:0] = 01b and ADGSPCR.PGS = 1), the ADST bit retains 1.

30.2.4 A/D Channel Select Register A0 (ADANSA0)

ADANSA0 selects analog input channels for A/D conversion among AN000 to AN007. In group scan mode, this register selects group A channels.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ANSA0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b7 to b0	ANSA0[7:0]	A/D Conversion Channels Select	0: AN000 to AN007 are not subjected to conversion. 1: AN000 to AN007 are subjected to conversion.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R0

ANSA0[7:0] Bits (A/D Conversion Channel Select)

The ANSA0[7:0] bits select analog input channels for A/D conversion among AN000 to AN007. The channels to be selected and the number of channels can be arbitrarily set. The ANSA0[0] bit corresponds to AN000 and the ANSA0[7] bit corresponds to AN007. When double trigger mode is selected, the channel selected by the ANSA0[7:0] bits is invalid, and the channel selected by the ADCSR.DBLANS[4:0] bits is selected in group A instead. The ANSA0[7:0] bits should be set while the ADCSR.ADST bit is 0.

30.2.5 A/D Channel Select Register B0 (ADANSB0)

ADANSB0 selects analog input channels for A/D conversion among AN000 to AN007 in group B when group scan mode is selected. The ADANSB0 register is not used in any scan mode other than group scan mode.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ANSB0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b7 to b0	ANSB0[7:0]	A/D Conversion Channels Select	0: AN000 to AN007 are not subjected to conversion. 1: AN000 to AN007 are subjected to conversion.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R0

ANSB0[7:0] Bits (A/D Conversion Channel Select)

The ANSB0[7:0] bits select analog input channels for A/D conversion among AN000 to AN07 in group B when group scan mode is selected. The ANSB0[0] bit corresponds to AN000 and the ANSB0[7] bit corresponds to AN007. The ANSB0[7:0] bits should be set while the ADCSR.ADST bit is 0.

30.2.6 A/D Channel Select Register C0 (ADANSC0)

ADANSC0 selects analog input channels for A/D conversion among AN000 to AN007 in group C when group scan mode is selected. The ADANSC0 register is not used in any scan mode other than group scan mode.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ANSC0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b7 to b0	ANSC0[7:0]	A/D Conversion Channels Select	0: AN000 to AN007 are not subjected to conversion. 1: AN000 to AN007 are subjected to conversion.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R0

ANSC0[7:0] Bits (A/D Conversion Channel Select)

The ANSC0[7:0] bits select analog input channels for A/D conversion among AN000 to AN007 in group C when group scan mode is selected. The ANSC0[0] bit corresponds to AN000 and the ANSB0[7] bit corresponds to AN007. The ANSC0[7:0] bits should be set while the ADCSR.ADST bit is 0.

30.2.7 A/D-Converted Value Addition/Average Mode Select Register 0 (ADADS0)

ADADS0 selects the channels AN000 to AN007 on which A/D conversion is performed successively for the specified number of times (The possible addition count depends on the conversion accuracy to be used. See Table 30.5 in section 30.2.1.) and then converted values are added (integrated) or averaged.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ADS0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b7 to b0	ADS0[7:0]	A/D-Converted Value Addition/ Average Channel Select	0: A/D-converted value addition/average mode for AN000 to AN007 is not selected. 1: A/D-converted value addition/average mode for AN000 to AN007 is selected.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R0

ADS0[7:0] Bits (A/D-Converted Value Addition/Average Channel Select)

When the ADS0[n] bit of the number that is the same as that of A/D-converted channel selected by the ANSA0[n] bits (n = 0 to 7) in ADANSA0 register or DBLANS[4:0] bits in ADCSR register, ANSB0[n] bits (n = 0 to 7) in ADANSB0 register, and ANSC0[n] bits (n = 0 to 7) in ADANSC0 register is set to 1, A/D conversion of analog input of the selected channels is performed successively (The possible addition count depends on the conversion accuracy to be used. See Table 30.5 in section 30.2.1.) that is set with the ADC[2:0] bits in ADADC register.
When the ADADC.AVEE bit is 0, the value obtained by addition (integration) is stored in the A/D data register. When the ADADC.AVEE bit is 1, the mean value of the results obtained by addition (integration) is stored in the A/D data register. As for the channel on which the A/D conversion is performed and addition/average mode is not selected, a normal one-time conversion is executed and the conversion result is stored to the A/D data register.
The ADS0[7:0] bits should be set while the ADCSR.ADST bit is 0.

Figure 30.1 shows the scan sequence when ADS0[2] and ADS0[6] are set to 1.
It is assumed that the scan is operated in continuous scan mode (ADCS = 10b in ADCSR), addition mode is selected (ADADC.AVEE = 0), the number of additions is set to 3 (conversion is performed four times) (ADADC.ADC[2:0] = 011b), and AN000 to AN007 is selected (ADANSA0.ANSA0[7:0] = 00FFh). Conversion is started from AN000. Conversion for AN002 is performed four times, and the value obtained by addition (integration) is stored in the A/D data register 2. Then, conversion for AN003 is started. Conversion for AN006 is performed four times successively, and the value obtained by addition (integration) is stored in the A/D data register 6. After conversion of AN007 is completed, the same sequence repeats from AN000.

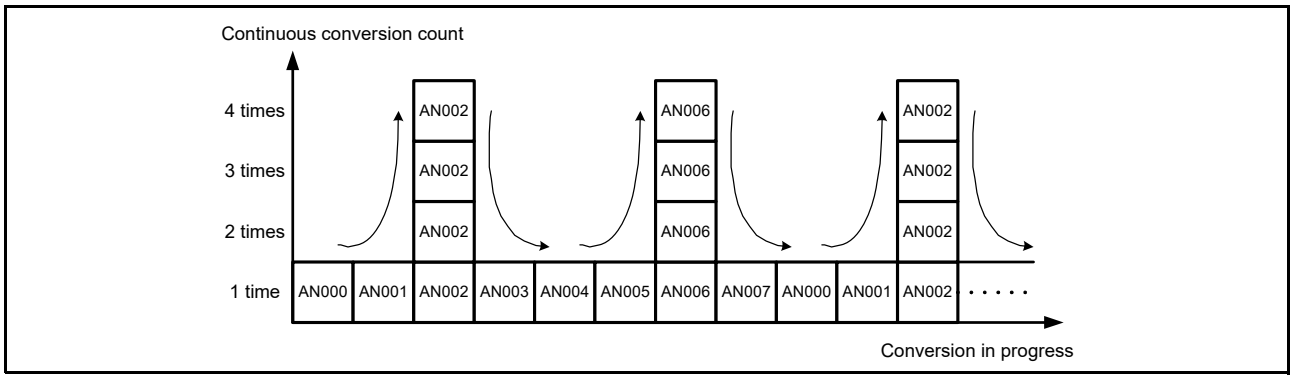


Figure 30.1 Scan Conversion Sequence with ADADC.ADC[2:0] = 011b, ADADC.AVEE = 0, ADS[2] = 1, and ADS[6] = 1

30.2.8 A/D-Converted Value Addition/Average Count Select Register (ADADC)

ADADC sets the addition count and average count for A/D conversion of the channel for which A/D-converted value addition/average mode is selected, and selects either addition or average mode.

	b7	b6	b5	b4	b3	b2	b1	b0
	AVEE	—	—	—	—	ADC[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b2 to b0	ADC[2:0]	Addition Count Select	b2 b1 b0 0 0 0: 1-time conversion (no addition; same as normal conversion) 0 0 1: 2-time conversion (addition once) 0 1 0: 3-time conversion (addition twice)*1 0 1 1: 4-time conversion (addition three times) 1 0 1: 16-time conversion (addition 15 times) Other settings are prohibited.	R/W
b6 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b7	AVEE	Average Mode Enable	0: Addition mode is selected.*1 1: Average mode is selected.*1	R/W

Note 1. The AVEE bit is only effective for 2-time and 4-time conversion. When average mode is selected by setting the ADADC.AVEE bit to 1, do not set the addition count to three times. (ADADC.ADC[2:0] = 010b)

ADC[2:0] Bits (Addition Count Select)

The ADC[1:0] bits set the addition count common to the channels for which A/D conversion and A/D-converted value addition/average mode is selected, including the channels selected in double trigger mode (by ADCSR.DBLANS[4:0] bits).

When average mode is selected by setting the ADADC.AVEE bit to 1, do not set the addition count to three times (ADADC.ADC[2:0] = 010b).

Additionally, as described in Table 30.5 in section 30.2.1, the combination of a condition in which the addition count is 16 (ADADC.ADC[2:0] = 101b) and a condition in which the A/D conversion accuracy is 8 or 10 bits (ADCER.ADPRC[1:0] = 10b or 01b) is a prohibited setting.

The ADC[2:0] bits should be set while the ADCSR.ADST bit is 0.

AVEE Bit (Average Mode Enable)

The AVEE bit selects addition or average mode for A/D conversion of the channel for which A/D conversion and A/D-converted value addition/average mode is selected, including the channels selected in double trigger mode (by ADCSR.DBLANS[4:0] bits).

When average mode is selected by setting the ADADC.AVEE bit to 1, do not set the addition count to three times (ADADC.ADC[2:0] = 010b). The average of 3-time conversion cannot be calculated.

The AVEE bits should be set while the ADCSR.ADST bit is 0.

30.2.9 A/D Control Extended Register (ADCER)

The ADCER register sets the A/D data register, A/D conversion accuracy, and self-diagnosis conversion.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADR FMT	—	—	—	DIAGM	DIAGLD	DIAGVAL[1:0]		—	—	ACE	—	—	ADPRC[1:0]		—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R0
b2, b1	ADPRC[1:0]	A/D Conversion Accuracy Specify	b2 b1 0 0: A/D conversion is performed with 12-bit accuracy. 0 1: A/D conversion is performed with 10-bit accuracy. 1 0: A/D conversion is performed with 8-bit accuracy. 1 1: Setting is prohibited.	R/W
b4, b3	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b5	ACE	A/D Data Register Automatic Clearing Enable	0: Disables automatic clearing. 1: Enables automatic clearing.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b9, b8	DIAGVAL[1:0]	Self-Diagnosis Conversion Voltage Select	b9 b8 0 0: Setting prohibited when self-diagnosis is enabled 0 1: Uses the voltage of 0 V for self-diagnosis. 1 0: Uses the voltage of reference power supply × 1/2 for self-diagnosis. 1 1: Uses the voltage of reference power supply for self-diagnosis.	R/W
b10	DIAGLD	Self-Diagnosis Mode Select	0: Rotation mode for self-diagnosis voltage 1: Fixed mode for self-diagnosis voltage	R/W
b11	DIAGM	Self-Diagnosis Enable	0: Disables self-diagnosis of A/D converter. 1: Enables self-diagnosis of A/D converter.	R/W
b14 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b15	ADRFMT	A/D Data Register Format Select	0: Right-alignment is selected for the A/D data register format. 1: Left-alignment is selected for the A/D data register format.	R/W

ADPRC[1:0] Bits (A/D Conversion Accuracy Specify)

These bits select the A/D conversion accuracy among 8-, 10-, or 12-bit accuracy. When the A/D conversion accuracy is changed, A/D conversion time is also changed.

See section 30.3.6, Analog Input Sampling and Scan Conversion Time for details.

The ADPCR[1:0] bits should be set while the ADCSR.ADST bit is 0.

ACE Bit (Automatic Clearing Enable)

The ACE bit selects enabling or disabling of automatic clearing (all “0”) of the A/D data register (ADDRy, ADRD, ADDBLDR, ADDBLDRA, or ADDBLDRB) when reading any of these registers by the CPU or DMACA. Automatic clearing of the A/D data register enables a failure which has not been updated in the A/D data register to be detected.

DIAGVAL[1:0] Bits (Conversion Voltage Select Self-Diagnosis)

For details, refer to the descriptions of the ADCER.DIAGLD bit.

Self-diagnosis should not be executed by setting the ADCER.DIAGLD bit to 1 when the ADCER.DIAGVAL[1:0] bits are set to 00b.

DIAGLD Bit (Self-Diagnosis Mode Select)

The DIAGLD bit selects whether the three voltage values are rotated or the fixed voltage is used in self-diagnosis. Setting this bit (ADCER.DIAGLD) to 0 allows conversion of the voltages in rotation mode where 0, the reference power supply × 1/2, and the reference power supply are converted in this order. The fixed voltage specified by the ADCER.DIAGVAL[1:0] bits is converted when self-diagnosis voltage fixed mode is selected. In self-diagnosis voltage rotation mode, the self-diagnosis voltage value does not return to 0 when scan conversion is completed. When scan conversion is restarted, therefore, rotation starts at the voltage value following the previous value. If fixed mode is switched to rotation mode, rotation starts at the fixed voltage value.
The DIAGLD bit should be set while the ADCSR.ADST bit is 0.

DIAGM Bit (Self-Diagnosis Enable)

The DIAGM bit selects execution of self-diagnosis.
Self-diagnosis is used to detect a failure of the A/D converter. It is used to convert the voltage value selected from the internally generated voltage values 0, the reference power supply × 1/2, and the reference power supply. When conversion is completed, information on the converted voltage and the conversion result is stored into the self-diagnosis data register (ADRD). ADRD can then be read out by software to determine whether the conversion result falls within the normal range (normal) or not (abnormal). Self-diagnosis is executed once at the beginning of each scan, and one of the three voltages is converted. To use the double trigger function (ADCSR.DBLE = 1), self-diagnosis should be deselected (set the DIAGM bit to 0).
When self-diagnosis is selected in group scan mode, self-diagnosis is separately executed in group A, B and C.
The DIAGM bit should be set while the ADCSR.ADST bit is 0.

ADRFMT Bit (A/D Data Register Format Select)

The ADRFMT bit specifies right-alignment or left-alignment for the data to be stored in ADDRy, ADRD, ADDBLDR, ADDBLDRA, ADDBLDRB, ADCMPDR0/1, ADWINLLB, or ADWINULB.
The ADRFMT bit should be set while the ADCSR.ADST bit is 0.
see section 30.2.23, Data Format for details.

30.2.10 A/D Start Trigger Select Register (ADSTRGR)

ADSTRGR selects the A/D conversion start trigger.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	TRSA[5:0]						—	—	TRSB[5:0]					
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b5 to b0	TRSB[5:0]	A/D Conversion Start Trigger Select for Group B	Select the A/D conversion start trigger for group B in group scan mode.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b13 to b8	TRSA[5:0]	A/D Conversion Start Trigger Select	Select the A/D conversion start trigger in single scan mode and continuous mode. In group scan mode, the A/D conversion start trigger for group A is selected.	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R0

TRSB[5:0] Bits (A/D Conversion Start Trigger Select for Group B)

The TRSB[5:0] bits select the trigger to start scanning of the analog input selected in group B. The TRSB[5:0] bits require to be set only in group scan mode and are not used in any other scan mode. For the scan conversion start trigger for group B, setting a software trigger or an asynchronous trigger is prohibited. Therefore, the TRSB[5:0] bits should be set to the value other than 000000b and the ADCSR.TRGE bit should be set to 1 in group scan mode.
When group is given priority in group scan mode, setting the ADGSPCR.GBRP bit to 1 allows group B to continuously operate in single scan mode. When setting the ADGSPCR.GBRP bit to 1, set the TRSB[5:0] bits to 3Fh.
Note that the issuance period of trigger for A/D conversion must be more than or equal to the actual scan conversion time (tSCAN). If the issuance period is less than tSCAN, A/D conversion by the trigger may have no effect.
Because an A/D conversion start trigger requires synchronization processing, a delay of the period for synchronization processing occurs. See section 30.3.6, Analog Input Sampling and Scan Conversion Time for details.
Table 30.7 lists the A/D conversion startup sources selected by the TRSB[5:0] bits.

TRSA[5:0] Bits (A/D Conversion Start Trigger Select)

The TRSA[5:0] bits select the trigger to start A/D conversion in single scan mode and continuous scan mode. In group scan mode, the trigger to start scanning of the analog input selected in group A is selected. When scanning is executed in group scan mode or double trigger mode, a software trigger and an asynchronous trigger cannot be used at the same time.

- When using the A/D conversion startup source of a synchronous trigger (MTU, GPT), set the TRGE bit in ADCSR to 1 and set the EXTRG bit in ADCSR to 0.
- When using the asynchronous trigger (ADTRG#), set the TRGE bit in ADCSR to 1 and set the EXTRG bit in ADCSR to 1.
- Software trigger (ADCSR.ADST) is enabled regardless of the settings of the ADCSR.TRGE bit, the ADCSR.EXTRG bit, and the TRSA[5:0] bits.

Note that the issuance period of trigger for A/D conversion must be more than or equal to the actual scan conversion time (tSCAN). If the issuance period is less than tSCAN, A/D conversion by a trigger may have no effect.
Because an A/D conversion start trigger requires synchronization processing, a delay of the period for synchronization processing occurs.
See section 30.3.6, Analog Input Sampling and Scan Conversion Time for details.
Table 30.8 lists the selection of A/D activation sources selected by the TRSA[5:0] bits.

Table 30.7 Selection of A/D Activation Sources by the TRSB[5:0] Bits (for Group B only)

Module	Source	Remarks	TRSB[5]	TRSB[4]	TRSB[3]	TRSB[2]	TRSB[1]	TRSB[0]
Trigger source de-selection state			1	1	1	1	1	1
MTU	TRGA0N	Compare match with or input capture to MTU0.TGRA	0	0	0	0	0	1
	TRGA1N	Compare match with or input capture to MTU1.TGRA	0	0	0	0	1	0
	TRGA2N	Compare match with or input capture to MTU2.TGRA	0	0	0	0	1	1
	TRGA3N	Compare match with or input capture to MTU3.TGRA	0	0	0	1	0	0
	TRGA4N	Compare match with or input capture to MTU4.TGRA, or an underflow of MTU4.TCNT (in the trough) in complementary PWM mode	0	0	0	1	0	1
	TRGA6N	Compare match with or input capture to MTU6.TGRA	0	0	0	1	1	0
	TRGA7N	Compare match with or input capture to MTU7.TGRA, or an underflow of MTU7.TCNT (in the trough) in complementary PWM mode	0	0	0	1	1	1
	TRG0N	Compare match with MTU0.TGRE	0	0	1	0	0	0
	TRG4AN	Compare match between MTU4.TADCORA and MTU4.TCNT	0	0	1	0	0	1
	TRG4BN	Compare match between MTU4.TADCORB and MTU4.TCNT	0	0	1	0	1	0
	TRG4AN or TRG4BN	Compare match between MTU4.TADCORA and MTU4.TCNT, or between MTU4.TADCORB and MTU4.TCNT	0	0	1	0	1	1
	TRG4ABN	Compare match between MTU4.TADCORA and MTU4.TCNT, and between MTU4.TADCORB and MTU4.TCNT (when interrupt skipping function 2 is in use)	0	0	1	1	0	0
	TRG7AN	Compare match between MTU7.TADCORA and MTU7.TCNT	0	0	1	1	0	1
	TRG7BN	Compare match between MTU7.TADCORB and MTU7.TCNT	0	0	1	1	1	0
	TRG7AN or TRG7BN	Compare match between MTU7.TADCORA and MTU7.TCNT, or between MTU7.TADCORB and MTU7.TCNT	0	0	1	1	1	1
	TRG7ABN	Compare match between MTU7.TADCORA and MTU7.TCNT, and between MTU7.TADCORB and MTU7.TCNT (when interrupt skipping function 2 is in use)	0	1	0	0	0	0

Module	Source	Remarks	TRSB[5]	TRSB[4]	TRSB[3]	TRSB[2]	TRSB[1]	TRSB[0]
GPT	ADTRGA0	Compare match with GPT0.GTADTRA	0	1	0	0	0	1
	ADTRGB0	Compare match with GPT0.GTADTRB	0	1	0	0	1	0
	ADTRGA1	Compare match with GPT1.GTADTRA	0	1	0	0	1	1
	ADTRGB1	Compare match with GPT1.GTADTRB	0	1	0	1	0	0
	ADTRGA2	Compare match with GPT2.GTADTRA	0	1	0	1	0	1
	ADTRGB2	Compare match with GPT2.GTADTRB	0	1	0	1	1	0
	ADTRGA3	Compare match with GPT3.GTADTRA	0	1	0	1	1	1
	ADTRGB3	Compare match with GPT3.GTADTRB	0	1	1	0	0	0
	ADTRGA0 or ADTRGB0	Compare match with GPT0.GTADTRA or with GPT0.GTADTRB	0	1	1	0	0	1
	ADTRGA1 or ADTRGB1	Compare match with GPT1.GTADTRA or with GPT1.GTADTRB	0	1	1	0	1	0
	ADTRGA2 or ADTRGB2	Compare match with GPT2.GTADTRA or with GPT2.GTADTRB	0	1	1	0	1	1
	ADTRGA3 or ADTRGB3	Compare match with GPT3.GTADTRA or with GPT3.GTADTRB	0	1	1	1	0	0

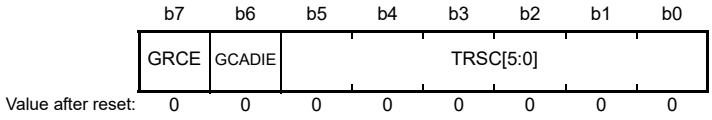
Table 30.8 Selection of A/D Activation Sources by the TRSA[5:0] Bits

Module	Source	Remarks	TRSA[5]	TRSA[4]	TRSA[3]	TRSA[2]	TRSA[1]	TRSA[0]
Trigger source de-selection state			1	1	1	1	1	1
External	ADTRG#	Input pin for the trigger	0	0	0	0	0	0
MTU	TRGA0N	Compare match with or input capture to MTU0.TGRA	0	0	0	0	0	1
	TRGA1N	Compare match with or input capture to MTU1.TGRA	0	0	0	0	1	0
	TRGA2N	Compare match with or input capture to MTU2.TGRA	0	0	0	0	1	1
	TRGA3N	Compare match with or input capture to MTU3.TGRA	0	0	0	1	0	0
	TRGA4N	Compare match with or input capture to MTU4.TGRA or, in complementary PWM mode, an underflow of MTU4.TCNT (in the trough)	0	0	0	1	0	1
	TRGA6N	Compare match with or input capture to MTU6.TGRA	0	0	0	1	1	0
	TRGA7N	Compare match with or input capture to MTU7.TGRA or, in complementary PWM mode, an underflow of MTU7.TCNT (in the trough)	0	0	0	1	1	1
	TRG0N	Compare match with MTU0.TGRE	0	0	1	0	0	0
	TRG4AN	Compare match between MTU4.TADCORA and MTU4.TCNT	0	0	1	0	0	1
	TRG4BN	Compare match between MTU4.TADCORB and MTU4.TCNT	0	0	1	0	1	0
	TRG4AN or TRG4BN	Compare match between MTU4.TADCORA and MTU4.TCNT, or between MTU4.TADCORB and MTU4.TCNT	0	0	1	0	1	1

Module	Source	Remarks	TRSA[5]	TRSA[4]	TRSA[3]	TRSA[2]	TRSA[1]	TRSA[0]
MTU	TRG4ABN	Compare match between MTU4.TADCORA and MTU4.TCNT, and between MTU4.TADCORB and MTU4.TCNT (when interrupt skipping function 2 is in use)	0	0	1	1	0	0
	TRG7AN	Compare match between MTU7.TADCORA and MTU7.TCNT	0	0	1	1	0	1
	TRG7BN	Compare match between MTU7.TADCORB and MTU7.TCNT	0	0	1	1	1	0
	TRG7AN or TRG7BN	Compare match between MTU7.TADCORA and MTU7.TCNT, or between MTU7.TADCORB and MTU7.TCNT	0	0	1	1	1	1
	TRG7ABN	Compare match between MTU7.TADCORA and MTU7.TCNT, and between MTU7.TADCORB and MTU7.TCNT (when interrupt skipping function 2 is in use)	0	1	0	0	0	0
GPT	ADTRGA0	Compare match with GPT0.GTADTRA	0	1	0	0	0	1
	ADTRGB0	Compare match with GPT0.GTADTRB	0	1	0	0	1	0
	ADTRGA1	Compare match with GPT1.GTADTRA	0	1	0	0	1	1
	ADTRGB1	Compare match with GPT1.GTADTRB	0	1	0	1	0	0
	ADTRGA2	Compare match with GPT2.GTADTRA	0	1	0	1	0	1
	ADTRGB2	Compare match with GPT2.GTADTRB	0	1	0	1	1	0
	ADTRGA3	Compare match with GPT3.GTADTRA	0	1	0	1	1	1
	ADTRGB3	Compare match with GPT3.GTADTRB	0	1	1	0	0	0
	ADTRGA0 or ADTRGB0	Compare match with GPT0.GTADTRA or with GPT0.GTADTRB	0	1	1	0	0	1
	ADTRGA1 or ADTRGB1	Compare match with GPT1.GTADTRA or with GPT1.GTADTRB	0	1	1	0	1	0
	ADTRGA2 or ADTRGB2	Compare match with GPT2.GTADTRA or with GPT2.GTADTRB	0	1	1	0	1	1
	ADTRGA3 or ADTRGB3	Compare match with GPT3.GTADTRA or with GPT3.GTADTRB	0	1	1	1	0	0

30.2.11 A/D Group C Trigger Select Register (ADGCTRGR)

The ADGCTRGR register enables the operation of group C and selects the A/D conversion start trigger. For details about the setting of group priority control, see section 30.3.4.3, Group Priority Control.



Bit	Symbol	Bit Name	Function	R/W
b5 to b0	TRSC[5:0]	A/D Conversion Start Trigger Select for Group C	Select the A/D conversion start trigger for group C in group scan mode.	R/W
b6	GCADIE	Group C Scan End Interrupt Enable	0: Disables S12GCADI0 interrupt generation upon group C scan completion. 1: Enables S12GCADI0 interrupt generation upon group C scan completion.	R/W
b7	GRCE	Group C Dedicated A/D Conversion Operation Enable	Sets whether to enable A/D conversion operation of group C. 0: Do not use group C. 1: Use group C.	R/W

TRSC[5:0] Bits (A/D Conversion Start Trigger Select for Group C)

The TRSC[5:0] bits select the trigger to start scanning of the analog input selected in group C. The TRSC[5:0] bits require to be set only in group scan mode and are not used in any other scan mode. For the scan conversion start trigger for group C, setting a software trigger or an asynchronous trigger is prohibited. Therefore, the TRSC[5:0] bits should be set to the value other than 000000b and the ADCSR.TRGE bit and ADCSR.GRCE bit should be set to 1 in group scan mode.

When group is given priority in group scan mode, setting the ADGSPCR.GBRP bit to 1 allows group C to continuously operate in single scan mode. When setting the ADGSPCR.GBRP bit to 1, set the TRSC[5:0] bits to 3Fh.

Note that the issuance period of trigger for A/D conversion must be more than or equal to the actual scan conversion time (tSCAN). If the issuance period is less than tSCAN, A/D conversion by the trigger may have no effect.

Because an A/D conversion start trigger requires synchronization processing, a delay of the period for synchronization processing occurs. See section 30.3.6, Analog Input Sampling and Scan Conversion Time for details.

Table 30.9 lists the A/D conversion startup sources selected by the TRSB[5:0] bits.

GCADIE Bit (Group C Scan End Interrupt Enable)

The GCADIE bit enables or disables generation of the group C scan end interrupt (S12GCADI0) in group scan mode.

GRCE Bit (Group C Dedicated A/D Conversion Operation Enable)

The GRCE bit should be set to 1 to use group C in group scan mode.

When the GRCE bit is set to 0, input of a trigger for group C is invalid. When ADGSPCR.GBRP is set to 1 under group priority control with group C (ADGSPCR.PGS bit = 1), single scan operation is performed continuously on group C. (When the GRCE bit is set to 1, single scan operation is not performed continuously on group B.)

The ADCSR.ADST bit must be 0 when the GRCE bit is set.

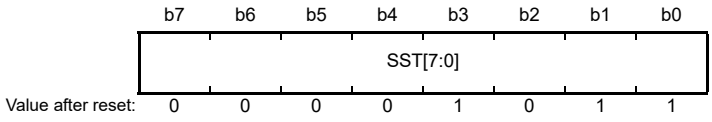
Table 30.9 Selection of A/D Activation Sources by the TRSC[5:0] Bits (for Group C only)

Module	Source	Remarks	TRSC[5]	TRSC[4]	TRSC[3]	TRSC[2]	TRSC[1]	TRSC[0]
Trigger source de-selection state			1	1	1	1	1	1
MTU	TRGA0N	Compare match with or input capture to MTU0.TGRA	0	0	0	0	0	1
	TRGA1N	Compare match with or input capture to MTU1.TGRA	0	0	0	0	1	0
	TRGA2N	Compare match with or input capture to MTU2.TGRA	0	0	0	0	1	1
	TRGA3N	Compare match with or input capture to MTU3.TGRA	0	0	0	1	0	0
	TRGA4N	Compare match with or input capture to MTU4.TGRA, or an underflow of MTU4.TCNT (in the trough) in complementary PWM mode	0	0	0	1	0	1
	TRGA6N	Compare match with or input capture to MTU6.TGRA	0	0	0	1	1	0
	TRGA7N	Compare match with or input capture to MTU7.TGRA, or an underflow of MTU7.TCNT (in the trough) in complementary PWM mode	0	0	0	1	1	1
	TRG0N	Compare match with MTU0.TGRE	0	0	1	0	0	0
	TRG4AN	Compare match between MTU4.TADCORA and MTU4.TCNT	0	0	1	0	0	1
	TRG4BN	Compare match between MTU4.TADCORB and MTU4.TCNT	0	0	1	0	1	0
	TRG4AN or TRG4BN	Compare match between MTU4.TADCORA and MTU4.TCNT, or between MTU4.TADCORB and MTU4.TCNT	0	0	1	0	1	1
	TRG4ABN	Compare match between MTU4.TADCORA and MTU4.TCNT, and between MTU4.TADCORB and MTU4.TCNT (when interrupt skipping function 2 is in use)	0	0	1	1	0	0
	TRG7AN	Compare match between MTU7.TADCORA and MTU7.TCNT	0	0	1	1	0	1
	TRG7BN	Compare match between MTU7.TADCORB and MTU7.TCNT	0	0	1	1	1	0
	TRG7AN or TRG7BN	Compare match between MTU7.TADCORA and MTU7.TCNT, or between MTU7.TADCORB and MTU7.TCNT	0	0	1	1	1	1
	TRG7ABN	Compare match between MTU7.TADCORA and MTU7.TCNT, and between MTU7.TADCORB and MTU7.TCNT (when interrupt skipping function 2 is in use)	0	1	0	0	0	0

Module	Source	Remarks	TRSC[5]	TRSC[4]	TRSC[3]	TRSC[2]	TRSC[1]	TRSC[0]
GPT	ADTRGA0	Compare match with GPT0.GTADTRA	0	1	0	0	0	1
	ADTRGB0	Compare match with GPT0.GTADTRB	0	1	0	0	1	0
	ADTRGA1	Compare match with GPT1.GTADTRA	0	1	0	0	1	1
	ADTRGB1	Compare match with GPT1.GTADTRB	0	1	0	1	0	0
	ADTRGA2	Compare match with GPT2.GTADTRA	0	1	0	1	0	1
	ADTRGB2	Compare match with GPT2.GTADTRB	0	1	0	1	1	0
	ADTRGA3	Compare match with GPT3.GTADTRA	0	1	0	1	1	1
	ADTRGB3	Compare match with GPT3.GTADTRB	0	1	1	0	0	0
	ADTRGA0 or ADTRGB0	Compare match with GPT0.GTADTRA or with GPT0.GTADTRB	0	1	1	0	0	1
	ADTRGA1 or ADTRGB1	Compare match with GPT1.GTADTRA or with GPT1.GTADTRB	0	1	1	0	1	0
	ADTRGA2 or ADTRGB2	Compare match with GPT2.GTADTRA or with GPT2.GTADTRB	0	1	1	0	1	1
	ADTRGA3 or ADTRGB3	Compare match with GPT3.GTADTRA or with GPT3.GTADTRB	0	1	1	1	0	0

30.2.12 A/D Sampling State Register n (ADSSTRn) (n = 0 to 7)

The ADSSTRn register sets the sampling time for analog input.



Bit	Symbol	Bit Name	Function	R/W
b7 to b0	SST[7:0]	Sampling Time Setting	These bits set the sampling time in the range from 5 to 255 states.	R/W

If one state is one ADCLK (A/D conversion clock) cycle and the ADCLK clock is 33 MHz, one state is 30.3 ns. The initial value is 11 states. If the impedance of analog input signal source is too high to secure sufficient sampling time or if the ADCLK clock is slow, the sampling time can be adjusted.
These bits should be set while the ADCSR.ADST bit is 0.

The lower limit of the sampling time setting depends on the frequency ratio:
If the frequency ratio of PCLK to ADCLK = 1:1 or 1:2, the sampling time must be set to 5 states or longer.

Table 30.10 shows the relationship between the A/D sampling state register and the relevant channels.
For details, refer to section 30.3.6, Analog Input Sampling and Scan Conversion Time.

Table 30.10 Relationship between A/D Sampling State Register and Relevant Channels

Bit Name	Corresponding Channel
ADSSTR0.SST[7:0] bits	AN000, Self-diagnosis
ADSSTR1.SST[7:0] bits	AN001
ADSSTR2.SST[7:0] bits	AN002
ADSSTR3.SST[7:0] bits	AN003
ADSSTR4.SST[7:0] bits	AN004
ADSSTR5.SST[7:0] bits	AN005
ADSSTR6.SST[7:0] bits	AN006
ADSSTR7.SST[7:0] bits	AN007

30.2.13 A/D Disconnection Detection Control Register (ADDISCR)

ADDISCR sets the disconnection detection assist function.

Bit	Symbol	Bit Name	Function	R/W
b4 to b0	ADNDIS[4:0]	Disconnection Detection Assist Setting	Disconnection detection assist function is set. b4 ADNDIS[4]: Setting precharge or discharge 0: discharge 1: precharge b3 to b0 ADNDIS[3:0]: Setting the period of precharge or discharge.	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R0

ADNDIS[4:0] Bits (Disconnection Detection Assist Setting)

These bits selects either precharge or discharge and the period of precharge/discharge for the A/D disconnection detection assist function. Setting the ADNDIS[4] = 1 allows to select precharge and setting the ADNDIS[4] = 0 allows to select discharge. The period of precharge/discharge can be set with the ADNDIS[3:0] bits. When the ADNDIS[3:0] = 0000b, the disconnection detection assist function is not effective. Setting of the ADNDIS[3:0] bits to 0001b is prohibited. Except for the case of ADNDIS[3:0] = 0000b or 0001b, the specified value indicates the number of states for the period of precharge/discharge.

ADNDIS[4:0] bits should be set while the ADCSR.ADST bit is 0.

30.2.14 A/D Group Scan Priority Control Register (ADGSPCR)

The ADGSPCR register sets priority control to interrupt scanning for a low priority group in group scan mode and execute scan for a priority group.

For details about the setting of group priority control, see section 30.3.4.3, Group Priority Control.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	GBRP	LGRRS	—	—	—	—	—	—	—	—	—	—	—	—	GBRSCN	PGS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	PGS	Group Priority Control Setting*1	0: Operation is without group A priority control 1: Operation is with group A priority control	R/W
b1	GBRSCN	Low Priority Group Restart Setting	(Enabled only when PGS = 1. Reserved when PGS = 0.) 0: Scanning for group is not restarted after having been discontinued due to group priority control. 1: Scanning for group is restarted after having been discontinued due to group priority control.	R/W
b8 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b13 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b14	LGRRS	Restart Channel Select	(Enabled only when PGS = 1 and GBRSCN = 1. Reserved when PGS = 0 or GBRSCN = 0.) 0: Rescan is performed from the first channel of the scan. 1: Rescan is performed from the channel on which A/D conversion has not completed.	R/W
b15	GBRP	Single Scan Continuous Start*2	(Enabled only when PGS = 1. Reserved when PGS = 0.) 0: Single scan is not continuously activated. 1: Single scan for the lowest priority group is continuously activated.	R/W

Note 1. When the PGS bit is to be set to 1, the ADCSR.ADCS[1:0] bits must be set to 01b (group scan mode). If the bits are set to any other values, proper operation cannot be guaranteed.

Note 2. When the GBRP bit is set to 1, single scan is performed continuously on the lowest priority group regardless of the setting of the GBRSCN bit.

PGS Bit (Group A Priority Control Setting)

This bit controls priority operation in group scan mode. Set the PGS bit to 1 to enable group priority control. When the PGS bit is to be set to 1, the ADCSR.ADCS[1:0] bits must be set to 01b (group scan mode). If the bits are set to any other values, proper operation cannot be guaranteed.

In group priority control, if a request to start scan for a priority group is accepted during scanning for a low priority group, the scanning for the low priority group is interrupted and scan for the priority group is started.

The priority order is group A > group B > group C. If a request to start group B scan is accepted during group C scanning, the group C scanning is interrupted and group B scan is started.

If a request to start group A scan is accepted during group C scanning, the group C scanning is interrupted and group A scan is started. Likewise, if a request to start group A scan is accepted during group B scanning, the group B scanning is interrupted and group A scan is started.

When the PGS bit has been set to 0, A/D conversion must be cleared by software according to section 30.5.2, Notes on Stopping A/D Conversion.

When the PGS bit has been set to 1, make register settings according to section 30.3.4.3, Group Priority Control.

GBRSCN Bit (Low priority group Restart Setting)

This bit controls the restarting of scan operation when group priority control is enabled. The setting of the GBRSCN bit has an effect when the PGS bit is set to 1.

If a scan operation on a low priority group has been stopped by a priority group trigger input with the GBRSCN bit set to 1, the scan operation is restarted on completion of the A/D conversion on a priority group. Also, if a low priority group

trigger is input during A/D conversion on a priority group, the scan operation on a low priority group is restarted on completion of the A/D conversion on a priority group. If the GBRSCN bit has been set to 0, triggers that are input during A/D conversion are ignored.
GBRSCN bit should be set while the ADCSR.ADST bit is 0.

LGRRS Bit (Restart Channel Select)

This bit sets the rescan start channel when group priority control is enabled. The setting of the LGRRS bit is valid when the PGS and GBRSCN bits are 1.
When the LGRRS bit is 0, scan to be performed after scan for the priority group is re-executed from the first channel for the low priority group whose scan was interrupted due to group priority control.
When the LGRRS bit is 1, scan to be performed after scan for the priority group is re-executed*1 from the channel on which A/D conversion has not completed for the low priority group whose scan was interrupted due to group priority control.
The LGRRS bits should be set while the ADCSR.ADST bit is 0.

Note: If A/D conversion for an addition setting channel has not completed for the specified number of times at the time of interruption, A/D conversion for the addition setting channel is performed again for the specified number of times at the time of re-execution.

GBRP Bit (Single Scan Continuous Start Setting)

Set the GBRP bit to perform single scan operation continuously on the lowest priority group under group priority control. This bit is set when a single scan operation is to be performed continuously on group B.

The setting of the GBRP bit is valid when the PGS bit is 1.
The lowest priority group is group C when groups A, B, C are used, and group B when only groups A and B are used.
Setting the GBRP bit to 1 starts a single scan for the lowest priority group. On completion of a scan, a single scan for the lowest priority group is automatically restarted. If scan has been interrupted due to group priority control, a single scan for the lowest priority group is automatically restarted on completion of the A/D conversion on the priority group.
Disable trigger input for the lowest priority group before setting the GBRP bit to 1. Setting the GBRP bit to 1 performs a rescan only for the lowest priority group even if the GBRSCN bit is 0.
GBRP bit should be set while the ADCSR.ADST bit is 0.

30.2.15 A/D Compare Control Register (ADCMPCR)

ADCMPCR is used to set compare window A/B operation.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CMPAIE	WCMPE	CMPBIE	—	CMPAE	—	CMPBE	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b8 to b0	—	Reserved	These bits are read as 0. The write value should be 0	R0
b9	CMPBE	Compare Window B Operation Enable	0: Compare Window B disabled 1: Compare Window B enabled	R/W
b10	—	Reserved	This bit is read as 0. The write value should be 0	R0
b11	CMPAE	Compare Window A Operation Enable	0: Compare Window A disabled 1: Compare Window A enabled	R/W
b12	—	Reserved	This bit is read as 0. The write value should be 0	R0
b13	CMPBIE	Compare B Interrupt Enable	0: Generation of an S12ADCMPBI0 interrupt in response to matches with a condition for comparison (Window B) is disabled. 1: Generation of an S12ADCMPBI0 interrupt in response to matches with a condition for comparison (Window B) is enabled.	R/W
b14	WCMPE	Window Function Setting	0: Window function disabled Window A and B operate as a comparator to compare the single value on the lower side with the A/D conversion result. 1: Window function enabled Window A and B operate as a window comparator to compare the two values on the upper and lower sides with the A/D conversion result.	R/W
b15	CMPAIE	Compare A Interrupt Enable	0: Generation of an S12ADCMPAI0 interrupt in response to matches with a condition for comparison (Window A) is disabled. 1: Generation of an S12ADCMPAI0 interrupt in response to matches with a condition for comparison (Window A) is enabled.	R/W

CMPBE Bit (Compare Window B Operation Enable)

This bit selects enabling or disabling of Window B operation. This bit should be set while the ADCSR.ADST bit is 0. Set this bit to 0 when setting the following registers. This bit sets to “0” when

- A/D Channel Select Register A0/B0 (ADANSA0, ADANSB0)
- The CMPCHB[5:0] bits in the A/D Compare Function Window-B Channel Selection Register (ADCMPBNSR) (ADCMPBNSR.CMPCHB[5:0])

CMPAE Bit (Compare Window A Operation Enable)

Enabling or disabling of Window A operation. This bit should be set while the ADCSR.ADST bit is 0.
Set this bit to 0 when setting the following registers.

- A/D Channel Select RegisterA0/B0 (ADANSA0, ADANSB0)
- Window-A Channel Selection Register 0 (ADCMPANSR0)

CMPBIE Bit (Compare B Interrupt Enable)

This bit selects enables or disables generation of a compare interrupt (S12ADCMPBI0) in response to matches with a condition for comparison (Window B).

WCMPE Bit (Window Function Setting)

This bit selects enabling or disabling of Window function. This bit should be set while the ADCSR.ADST bit is 0.

CMPAIE Bit (Compare A Interrupt Enable)

This bit selects enables or disables generation of a compare interrupt (S12ADCMPAI0) in response to matches with a condition for comparison (Window A).

30.2.16 A/D Compare Function Window-A Channel Selection Register 0 (ADCMPANSR0)

ADCMPANSR0 selects analog input channels for comparison from among AN000 to AN007 in the compare window A condition.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CMPCHA0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b7 to b0	CMPCHA0[7:0]	Compare Window A Channel Select	0: The corresponding channel from among AN000 to AN007 is not a target for the compare Window A. 1: The corresponding channel from among AN000 to AN007 is a target for the compare Window A.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0	R0

CMPCHA0[7:0] Bits (Compare Channel Select)

Setting the CMPCHA0[n] bit which has the same number as the A/D channel selected by the ADANSA.ANSA[n] (n = 0 to 7), ADANSB.ANSB[n] (n = 0 to 7) or ADANSC.ANSC[n] (n = 0 to 7) bits to 1 enables comparison with that channel. Set the CMPCHA0[7:0] bits while ADCSR.ADST bit is 0.

30.2.17 A/D Compare Function Window-A Comparison Condition Setting Register 0 (ADCMPLR0)

The ADCMPLR0 register sets the condition for use in comparing the values of the ADCMPDR0 and ADCMPDR1 registers with results of A/D conversion.
Set the ADCMPLR0 register while ADCSR.ADST is 0.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CMPLCHA0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b7 to b0	CMPLCHA0 [7:0]	Compare window A Compare Level Select	Set the condition for comparison with the selected channels from among AN000 to AN007 under the compare window A condition. Compare conditions are shown in Table 30.11. When Window function is disabled (ADCMPCR.WCMPE = 0): 0: ADCMPDR0 register value > A/D-converted value 1: ADCMPDR0 register value < A/D-converted value When Window function is enabled (ADCMPCR.WCMPE = 1): 0: AD-converted value < ADCMPDR0 register value or A/D-converted value > ADCMPDR1 register value 1: ADCMPDR0 register value < A/D-converted value < ADCMPDR1 register value	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0	R0

CMPLCHA0[7:0] Bits (Compare Condition Select)

The CMPLCHA0[7:0] bits set the condition for use in comparison with the selected channel from among AN000 to AN007 under the compare window A condition. A condition can be set for individual comparison of each analog input. The CMPLCHA0[0] bit is used for AN000, and the CMPLCHA0[7] bit is used for AN007.
When the result of comparison matches the set condition, ADCMPSR0.CMPSTCHA0n is set to 1 and a compare interrupt (S12ADCMPAI0) is generated.

Table 30.11 Compare function description of compare function window A

Compare condition when Window function is disabled

CMPLCHA0[*] = 1'b0		CMPLCHA0[*] = 1'b1	
ADCMPDR0 register value ≤ A/D-converted value	Not matched	ADCMPDR0 register value < A/D-converted value	Matched
ADCMPDR0 register value > A/D-converted value	Matched	ADCMPDR0 register value ≥ A/D-converted value	Not matched

Compare condition when Window function is enabled

CMPLCHA0[*] = 1'b0	
ADCMPDR1 register value < A/D-converted value	Matched
ADCMPDR0 register value ≤ A/D-converted value ≤ ADCMPDR1 register value	Not matched
A/D-converted value < ADCMPDR0 register value	Matched

CMPLCHA0[*] = 1'b1	
ADCMPDR1 register value ≤ A/D-converted value	Not matched
ADCMPDR0 register value < A/D-converted value < ADCMPDR1 register value	Matched
A/D-converted value ≤ ADCMPDR0 register value	Not matched

30.2.18 A/D Compare Function Window-A Lower Level Setting Register (ADCMPDR0), A/D Compare Function Window-A Upper Level Setting Register (ADCMPDR1), A/D Compare Function Window-B Lower Level Setting Register (ADWINLLB), A/D Compare Function Window-B Upper Level Setting Register (ADWINULB)

The ADCMPDR0 register is a readable/writable register to set the reference data when the compare window A function is used. ADCMPDR0 sets the reference data for comparison with the selected channels.

ADCMPDR0 sets the lower-side reference value (CMPLLA[15:0]) for window A, and ADCMPDR1 sets the upper-side reference value (CMPULA[15:0]) for window A.

The ADWINULB and ADWINLLB registers are readable/writable registers to set the reference data when the compare window B function is used.

ADWINLLB sets the lower-side reference value (CMPLLB[15:0]) for window B, and ADWINULB sets the upper-side reference value (CMPULB[15:0]) for window B.

- The ADCMPDRy, ADWINULB, and ADWINLLB registers are writable even during A/D conversion. The reference value can be dynamically*1 changed by rewriting register values during A/D conversion.
- Specify the settings so that upper-side reference value ≥ lower-side reference value (CMPULA ≥ CMPLLA, CMPULB ≥ CMPLLB).
- The ADCMPDR1 and ADWINULB registers are not used when Window function is disabled.

Note 1. Each of the upper-side and lower-side reference values are changed when the corresponding register is written. For example, a comparison is made between the upper-side reference value (after rewriting) and the lower-side reference value (before rewriting) as shown below in the time between rewriting of the upper-side reference value and rewriting of the lower-side reference value.

If the comparison during the rewriting of these two reference values is problematic, rewrite these reference values when the ADCSR.ADST bit and the associated window operation enable bit (ADCMPCR.CMPAE or ADCMPCR.CMPBE) are 0.

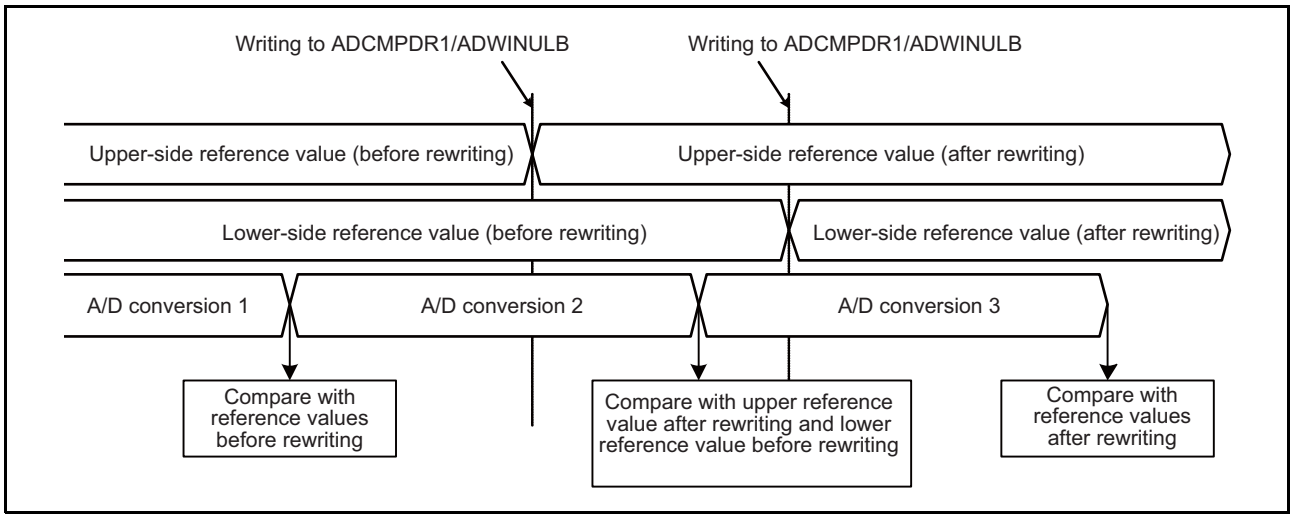


Figure 30.2 Reflection Timing of Reference Values

Note the following settings so that the ADCMPDRy, ADWINULB, and ADWINLLB registers are set according to the A/D data register format shown in section 30.2.1, A/D Data Register y (ADDRy: y = 0 to 7), A/D Data Duplication Register (ADDBLDR), A/D Data Duplication Register A (ADDBLDRA), A/D Data Duplication Register B (ADDBLDRB)*2.

For the correspondence between each condition and data format, see section 30.2.1, Table 30.5.

- The value of the A/D data register format select bit (flush-right or flush-left)
- The value of A/D conversion accuracy specify bits (12 bits, 10 bits, 8 bits)
- The value of A/D-converted value addition/average mode select register (A/D conversion value average mode is selected or not selected)

Note 2. If a compare value is set in a format different from the A/D data register format, the correct comparison result cannot be obtained.

For details about this register format, see section 30.2.23, (1) to (9). In that section, see the description as follows:

- For ADCMPDR0, replace bit symbol "DATA[*:0]" with "CMPLLA[*:0]".
  "Function" should be n-bit compare level (lower-side reference value for window A) (n = 8, 10, 12, 14, 16).
  The R/W attribute of CMPLLA[*:0] should be R/W.
- For ADCMPDR1, replace bit symbol "DATA[*:0]" with "CMPULA[*:0]".
  "Function" should be n-bit compare level (upper-side reference value for window A) (n = 8, 10, 12, 14, 16).
  The R/W attribute of CMPULA[*:0] should be R/W.
- For ADWINLLB, replace bit symbol "DATA[*:0]" with "CMPLLB[*:0]".
  "Function" should be n-bit compare level (lower-side reference value for window B) (n = 8, 10, 12, 14, 16).
  The R/W attribute of CMPLLB[*:0] should be R/W.
- For ADWINULB, replace bit symbol "DATA[*:0]" with "CMPULB[*:0]".
  "Function" should be n-bit compare level (upper-side reference value for window B) (n = 8, 10, 12, 14, 16).
  The R/W attribute of CMPULB[*:0] should be R/W.

30.2.19 A/D Compare Function Window-A Channel Status Register 0 (ADCMPSR0)

The ADCMPSR0 register sets the compare window A function.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CMPSTCHA0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b7 to b0	CMPSTCHA0 [7:0]	Compare Window-A Flag	Indicates compare results of CH(AN000 to AN007) when window A is active (ADCMPCR.CMPAE = 1b). 0: Condition for comparison was not met. 1: Condition for comparison was met.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R0

CMPSTCHA0[7:0] Bits (Compare Window-A Flag)

These bits are status flags that indicate the comparison result of channels (AN000 to AN007) that are subject to the window A comparison condition.

When the result of comparison on completion of A/D conversion matches the condition set in ADCMPLR0.CMPLCHAn, the corresponding flags are set to 1. When the ADCMPCR.CMPAIE bit is 1, a compare interrupt (S12ADCMPAI0) request is generated when the setting of the flag becomes 1. The CMPSTCHA0[0] bit is used for AN000, and the CMPSTCHA0[7] bit is used for AN007. The value 1 cannot be written to the CMPSTCHA0n bit.

[Setting “1” condition]

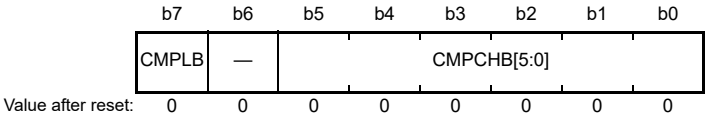
When ADCMPCR.CMPAE = 1b, The condition set in ADCMPLR0.CMPLCHAn is met.

[Clearing “0” condition]

0 is written after reading 1.

30.2.20 A/D Compare Function Window-B Channel Selection Register (ADCMPBNSR)

The ADCMPBNSR register sets the compare window B function.



Bit	Symbol	Bit Name	Function	R/W
b5 to b0	CMPCHB [5:0]	Compare Window B Channel Select	Selects a channel to be compared with the compare window B conditions. 6'b000000: AN000 6'b000001: AN001 6'b000010: AN002 to 6'b111111: Not selected Note: Setting any value from 6'b001000 to 6'b111110 is prohibited.	R/W
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R0
b7	CMPLB	Compare Window B Compare Condition Setting	Sets comparison conditions for channels for window B. Table 30.12 shows the compare conditions. When Window function is disabled (ADCMPCR.WCMPE bit = 0): 0: CMPLLB register value > A/D-converted value 1: CMPLLB register value < A/D-converted value When Window function is enabled (ADCMPCR.WCMPE bit = 1): 0: A/D-converted value < CMPLLB register value or CMPULB register value < A/D-converted value 1: CMPLLB register value < A/D-converted value < CMPULB register value	R/W

CMPCHB[5:0] Bits (Compare Window B Channel Select)

These bits select a channel to be compared with the compare window B conditions from AN000 to AN007.
Specifying the number (hexadecimal) of the A/D conversion channel selected in the ADANSA0.ANSA0[n] (n = 0 to 7) bits and ADANSB0.ANSB0[n] (n = 0 to 7) bits enables the compare window B function.
Set the CMPCHB[5:0] bits while the ADCSR.ADST bit is 0.

CMPLB Bit (Compare Window B Compare Condition Setting)

This bit sets comparison conditions for channels for window B. When the result of comparison of analog input matches the set condition, ADCMPBSR0.CMPSTB is set to 1 and a compare interrupt (S12ADCMPBI0) is generated.

Table 30.12 Explanation of Compare Conditions for Compare Function Window B

Compare condition when Window function is disabled

CMPLB = 1'b0

ADWINLLB register value ≤ A/D-converted value	Not matched
ADWINLLB register value	Matched

CMPLB = 1'b1

ADWINLLB register value < A/D-converted value	Matched
ADWINLLB register value ≥ A/D-converted value	Not matched

Compare condition when Window function is enabled

CMPLB = 1'b0

A/D-converted value > ADWINULB register value	Matched
ADWINLLB register value ≤ A/D-converted value ≤ ADWINULB register value	Not matched
A/D-converted value < ADWINLLB register value	Matched

CMPLB = 1'b1

A/D-converted value ≥ ADWINULB register value	Not matched
ADWINLLB register value < A/D-converted value < ADWINULB register value	Matched
A/D-converted value ≤ ADWINLLB register value	Not matched

30.2.21 A/D Compare Function Window-B Status Register (ADCMPBSR)

The ADCMPBSR register stores the result of comparison by the compare window B function.

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	CMPSTB
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	CMPSTB	Compare Window-B Flag	Indicates compare results of CH (AN000 to AN007) when window B is active (ADCMPCR.CMPBE = 1b) 0: Condition for comparison was not met. 1: Condition for comparison was met.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R0

CMPSTB Bits (Compare Window-B Flag)

These bits are status flags that indicate the comparison result of channels (AN000 to AN007) that are subject to the window B comparison condition. When the result of comparison on completion of A/D conversion matches the condition set in ADCMPBNSR.CMPLB, the corresponding flags are set to 1. When the ADCMPCR.CMPBIE bit is 1, a compare interrupt (S12ADCMPBI0) request is generated when the setting of the flag becomes 1. The value 1 cannot be written to the CMPSTB bit.

[Setting “1” condition]

When ADCMPCR.CMPBE = 1b, The condition set in ADCMPBNSR.CMPLB is met.

[Clearing “0” condition]

0 is written after reading 1.

30.2.22 A/D Compare Function AB Status Monitor Register (ADWINMON)

The ADWINMON register can be used to monitor comparison results and combination results.

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	MON CMPB	MON CMPA	—	—	—	MON COMB
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b5	MONCMPB	Comparison Result Monitor B	0: Window B comparison conditions are not met. 1: Window B comparison conditions are met.	R
b4	MONCMPA	Comparison Result Monitor A	0: Window A comparison conditions are not met. 1: Window A comparison conditions are met.	R
b3 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b0	MONCOMB	Combination Result Monitor	Indicates the combination result. This bit is valid while both window A and B are active. 0: Window A/B composite conditions are not met. 1: Window A/B composite conditions are met.	R

MONCMPB Bit (Comparison Result Monitor B)

This bit is a read-only bit which is read as 1 when the A/D-converted value of the target channel for window B meets the condition set in the ADCMPBNSR.CMPLB bit, and is read as 0 when it does not meet the condition.
[Setting “1” condition]
Under the condition that ADCMPCR.CMPBE = 1b, the condition set in ADCMPBNSR.CMPLB is met.
[Setting “0” condition]
Under the condition that ADCMPCR.CMPBE = 1b, the condition set in ADCMPBNSR.CMPLB is not met.
When ADCMPCR.CMPBE = 0b (Automatic clearing is performed when the value of ADCMPCR.CMPBE is changed from 1b to 0b.)

MONCMPA Bit (Comparison Result Monitor A)

This bit is a read-only bit which is read as 1 when the A/D-converted value of the target channel for window A meets the condition set in ADCMPLR0 and ADCMPLER, and is read as 0 when it does not meet the condition.
[Setting “1” condition]
Under the condition that ADCMPCR.CMPAE = 1b, the condition set in ADCMPLR0.CMPLCHAn is met.
[Setting “0” condition]
Under the condition that ADCMPCR.CMPAE = 1b, the condition set in ADCMPLR0.CMPLCHAn is not met.
When ADCMPCR.CMPAE = 0b (Automatic clearing is performed when the value of ADCMPCR.CMPAE is changed from 1b to 0b.)

MONCOMB Bit (Combination Result Monitor)

This read-only bit indicates a result of combining comparison condition result A and comparison result condition B with the combination condition set in the ADCMPCR.CMPAB[1:0] bits.
[Setting “1” condition]
Under the condition that ADCMPCR.CMPAE = 1 and ADCMPCR.CMPBE = 1, the combination condition set in the ADCMPCR.CMPAB[1:0] bits is met.
[Setting “0” condition]
The combination condition set in the ADCMPCR.CMPAB[1:0] bits is not met.
When ADCMPCR.CMPAE = 0 or ADCMPCR.CMPBE = 0

30.2.23 Data Format

The data formats in this module are shown below.

Data Format (1)

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	DATA[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b7 to b0	DATA[7:0]	A/D-converted value/compare level	8-bit A/D-converted value	R
			8-bit compare level	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R0

Data Format (2)

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DATA[7:0]								—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

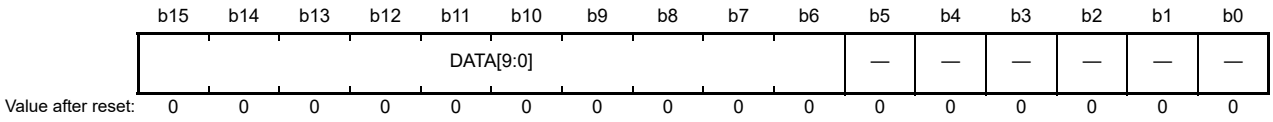
Bit	Symbol	Bit Name	Function	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b15 to b8	DATA[7:0]	A/D-converted value/compare level	8-bit A/D-converted value	R
			8-bit compare level	R/W

Data Format (3)

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	DATA[9:0]									
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

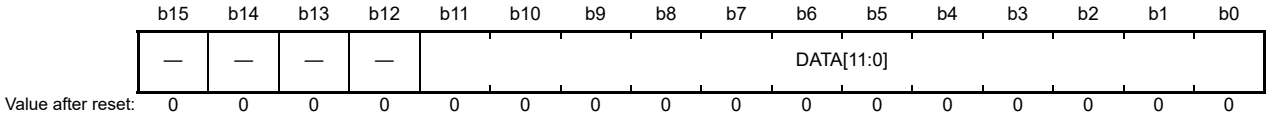
Bit	Symbol	Bit Name	Function	R/W
b9 to b0	DATA[9:0]	A/D-converted value/compare level	10-bit A/D-converted value	R
			10-bit compare level	R/W
b15 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R0

Data Format (4)



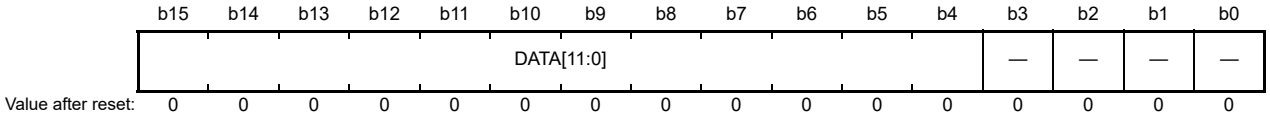
Bit	Symbol	Bit Name	Function	R/W
b5 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b15 to b6	DATA[9:0]	A/D-converted value/compare level	10-bit A/D-converted value	R
			10-bit compare level	R/W

Data Format (5)



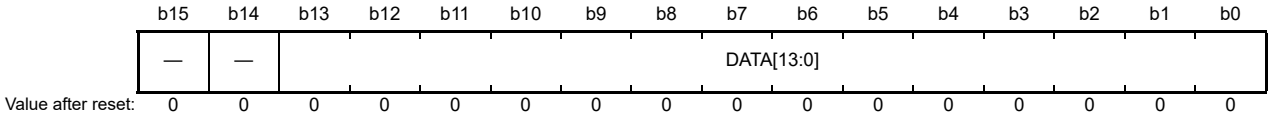
Bit	Symbol	Bit Name	Function	R/W
b11 to b0	DATA[11:0]	A/D-converted value/compare level	12-bit A/D-converted value	R
			12-bit compare level	R/W
b15 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R0

Data Format (6)



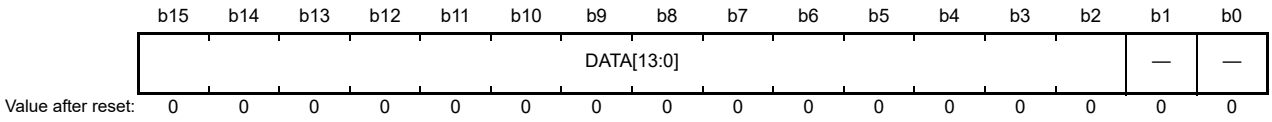
Bit	Symbol	Bit Name	Function	R/W
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b15 to b4	DATA[11:0]	A/D-converted value/compare level	12-bit A/D-converted value	R
			12-bit compare level	R/W

Data Format (7)



Bit	Symbol	Bit Name	Function	R/W
b13 to b0	DATA[13:0]	A/D-converted value/compare level	14-bit A/D-converted value	R
			14-bit compare level	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R0

Data Format (8)



Bit	Symbol	Bit Name	Function	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R0
b15 to b2	DATA[13:0]	A/D-converted value/compare level	14-bit A/D-converted value	R
			14-bit compare level	R/W

Data Format (9)

Bit	Symbol	Bit Name	Function	R/W
b15 to b0	DATA[15:0]	A/D-converted value/compare level	16-bit A/D-converted value	R
			16-bit compare level	R/W

30.3 Operation

30.3.1 Scanning Operation

In scanning, A/D conversion is performed sequentially on the analog inputs of the specified channels.
A scan conversion is performed in three operating modes: single scan mode, continuous scan mode, and group scan mode. In single scan mode, one or more specified channels are scanned once. In continuous scan mode, one or more specified channels are scanned repeatedly until the ADST bit in ADCSR register is cleared to 0 from 1 by software.
In group scan mode, the selected channels of group A, the selected channels of group B, and the selected channels of group B are scanned once after starting to be scanned according to the respective synchronous triggers.
In single scan mode and continuous scan mode, A/D conversion is performed for ANn channels selected by the ADANSA0 register, starting from the channel with the smallest number n. In group scan mode, A/D conversion is performed for ANn channels of group A selected by the ADANSA0 register first, and then performed for ANn channels of group B selected by the ADANSB0 register, and then performed for ANn channels of group C selected by the ADANSC0 register, respectively, starting from the channel with the smallest number n.
When self-diagnosis is selected, it is executed once at the beginning of each scan and one of the three voltages internally generated in the A/D converter is converted.
Double trigger mode is to be used with single scan mode or group scan mode. With double trigger mode being enabled (set ADCSR.DBLE to 1) , A/D conversion data of a channel selected by the DBLANS[4:0] bits in ADCSR register is duplicated only if the conversion is started by any of the synchronous triggers selected by the TRSA[5:0] bits in ADSTRGR register.
In group scan mode, the double trigger function can be used only for group A.

Extended double trigger mode means that double trigger mode is enabled, and the following synchronous triggers (two types of synchronous trigger source enabled) are selected as the A/D conversion start trigger in ADSTRGR.TRSA[5:0].

- TRG4AN or TRG4BN (Sets ADSTRGR.TRSA[5:0] to “001011”.)
- TRG7AN or TRG7BN (Sets ADSTRGR.TRSA[5:0] to “001111” .)
- ADTRGA0 or ADTRGB0 (Sets ADSTRGR.TRSA[5:0] to “011001”.)
- ADTRGA1 or ADTRGB1 (Sets ADSTRGR.TRSA[5:0] to “011010”.)
- ADTRGA2 or ADTRGB2 (Sets ADSTRGR.TRSA[5:0] to “011011”.)
- ADTRGA3 or ADTRGB3 (Sets ADSTRGR.TRSA[5:0] to “011100”.)

In extended double trigger mode, in addition to normal double trigger mode operation, A/D conversion data triggered by TRGnAN (n = 4, 7) and ADTRGAm (m = 0 to 3) is stored in A/D data duplication register A (ADDBLDRA), and A/D conversion data triggered by TRGnBN (n = 4, 7) and ADTRGBm (m = 0 to 3) is stored in A/D data duplication register B (ADDBLDRB).

If two types of trigger sources have occurred simultaneously, the A/D conversion data is stored in A/D data duplication register B (ADDBLDRB) without being sorted by the trigger sources. Note that if a new trigger is input during A/D conversion caused by another trigger, the new trigger is ignored.

30.3.2 Single Scan Mode

30.3.2.1 Basic Operation

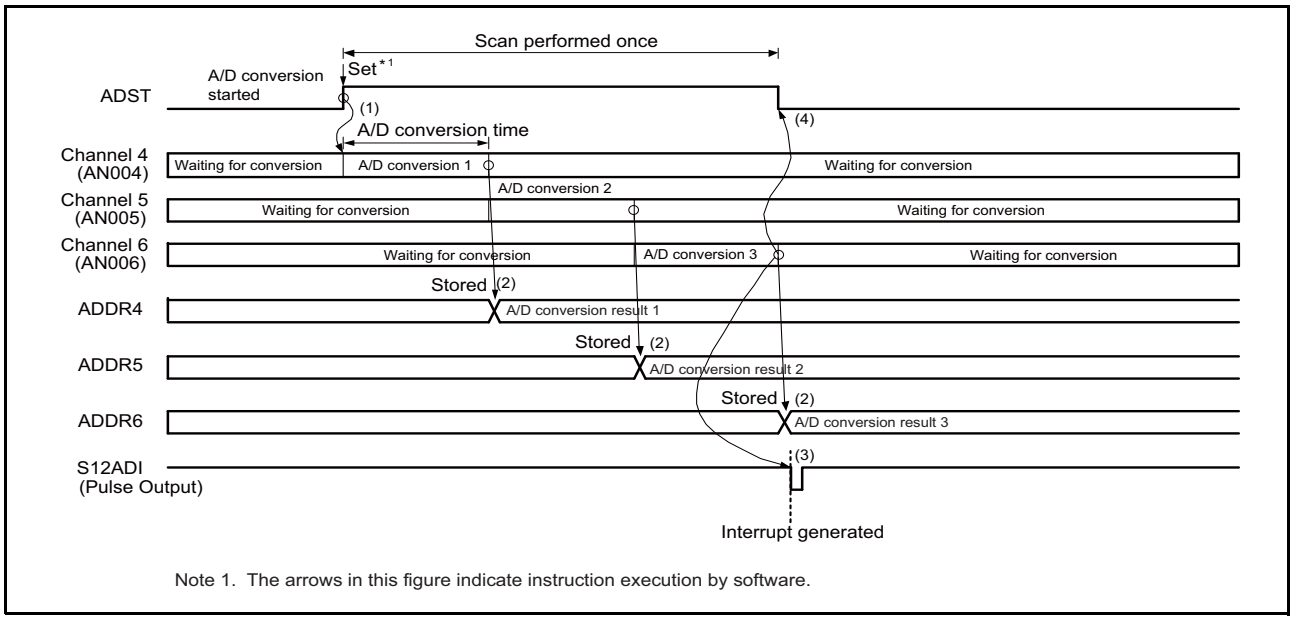
In basic operation of single scan mode, A/D conversion is performed once on the analog input of the specified channels as below.

(1) When the ADST bit in ADCSR register is set to 1 (A/D conversion start) by a software trigger, input of a synchronous trigger signal (MTU, GPT), or input of an asynchronous trigger. A/D conversion is performed for ANn channels selected by the ADANSA0 register, starting from the channel with the smallest number n.

(2) Each time A/D conversion of a single channel is completed, the result of A/D conversion is stored in the corresponding A/D data register (ADDRy).

(3) When A/D conversion of all the selected channels is completed, an S12ADI0 interrupt request is generated if the ADIE bit in ADCSR is 1 (S12ADI0 interrupt upon scanning completion is enabled).

(4) The ADST bit in ADCSR remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion of all the selected channels is completed. Then the A/D converter enters a waiting state.



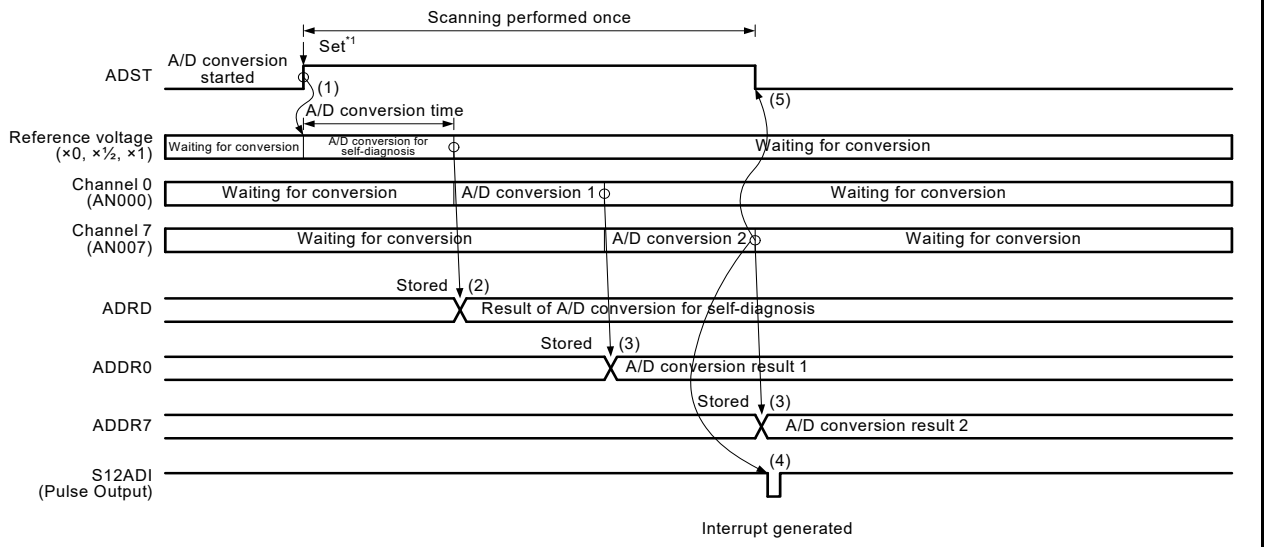
Note 1. The arrows in this figure indicate instruction execution by software.

Figure 30.3 Example of Operation in Single Scan Mode (Basic Operation: AN004 to AN006 Selected)

30.3.2.2 Channel Selection and Self-Diagnosis

When channels and self-diagnosis are selected, A/D conversion is first performed for the reference voltage AVcc supplied to the A/D converter, and then A/D conversion is performed once on the analog input of the selected channels as below.

(1) A/D conversion for self-diagnosis is first started when the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (MTU, GPT), or an asynchronous trigger input.
(2) When A/D conversion for self-diagnosis is completed, the A/D conversion result is stored into the A/D self-diagnosis data register (ADRD). A/D conversion is then performed for ANn pins selected by the ADANSA0 register, starting from the channel with the smallest number n.
(3) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(4) When A/D conversion of all the selected channels is completed, an S12ADI0 interrupt request is generated if the ADIE bit in ADCSR is 1 (S12ADI0 interrupt upon scanning completion enabled).
(5) The ADST bit remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion of all the selected channels is completed. Then the A/D converter enters a waiting state.



Note 1. The arrows in this figure indicate instruction execution by software.

Figure 30.4 Example of Operation in Single Scan Mode
(Basic Operation: AN000 and AN007 Selected + Self-Diagnosis)

30.3.2.3 A/D Conversion in Double Trigger Mode

When double trigger mode is selected in single scan mode, A/D conversion is performed for two rounds of single scan operation started by a synchronous trigger (MTU, GPT) as a sequence as shown below. Self-diagnosis should be deselected.
Duplication of A/D conversion data is enabled by setting the channel numbers to be duplicated to the DBLANS[4:0] bits in ADCSR and setting the DBLE bit in ADCSR to 1. When the DBLE bit in ADCSR is set to 1, channel selection using the ADANSA0 register is invalid. In double trigger mode, a synchronous trigger (MTU, GPT) should be selected using the TRSA[5:0] bits in ADSTRGR; the EXTRG bit and TRGE bit in ADCSR should be set to 0 and 1, respectively. Software trigger should not be used.

(1) When the ADST bit in ADCSR is set to 1 (A/D conversion start) by a synchronous trigger input (MTU, GPT), A/D conversion is started on the single channel selected by the DBLANS[4:0] bits in ADCSR.
(2) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(3) The ADST bit in ADCSR is automatically cleared to 0 and the A/D converter enters a waiting state. Here, an S12ADI0 interrupt request is not generated irrespective of the ADIE (S12ADI0 interrupt upon scanning completion enabled) bit setting in ADCSR.
(4) When the ADST bit in ADCSR is set to 1 (A/D conversion start) by the second synchronous trigger input, A/D conversion is started on the single channel selected by the DBLANS[4:0] bits in ADCSR.
(5) When A/D conversion is completed, the A/D conversion result is stored into the A/D data duplication register (ADDBLDR), which is exclusively used in double trigger mode.
(6) If the ADIE bit in ADCSR is 1 (S12ADI0 interrupt upon scanning completion enabled), an S12ADI0 interrupt is generated.
(7) The ADST bit remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion is completed. Then the A/D converter enters a waiting state.

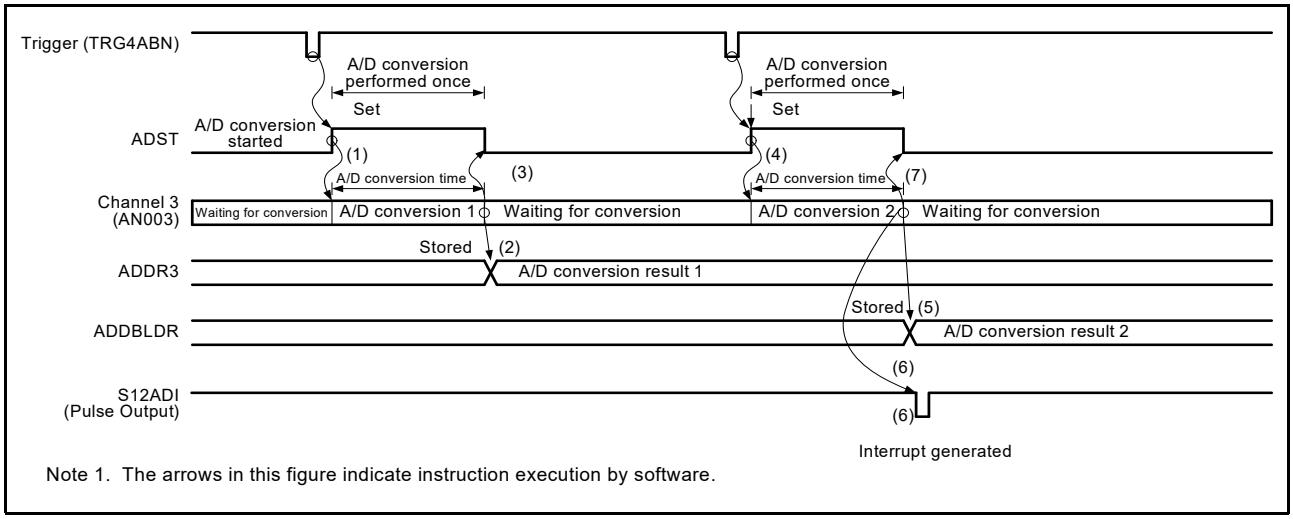


Figure 30.5 Example of Operation in Single Scan Mode
(Double Trigger Mode Selected; AN003 Duplicated; TRG4ABN is selected as the trigger)

30.3.2.4 Extended Operations When Double Trigger Mode is Selected

When double trigger mode is selected in single scan mode, two rounds of single scan operation are performed as shown below.

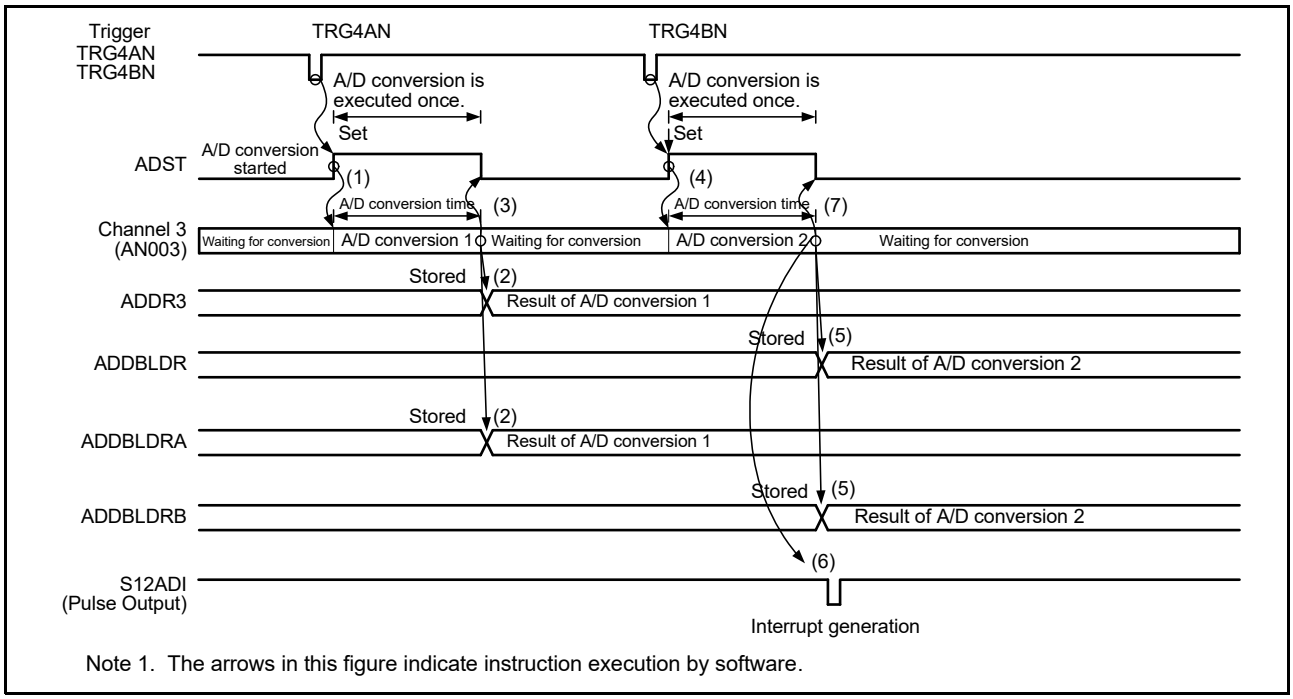
In double trigger extended mode, should be selected TRG4AN or TRG4BN, TRG7AN or TRG7BN, ADTRGA0 or ADTRGB0, ADTRGA1 or ADTRGB1, ADTRGA2 or ADTRGB2, and ADTRGA3 or ADTRGB3 using the TRSA[5:0] bits in ADSTRGR register; the EXTRG bit and TRGE bit in ADCSR register should be set to 0 and 1, respectively.

Software trigger should not be used.

Self-diagnosis should be deselected.

Duplication of A/D conversion data is enabled by setting the channel numbers to be duplicated to the DBLANS[4:0] bits in ADCSR register and setting the DBLE bit in ADCSR register to 1. When the DBLE bit in ADCSR register is set to 1, channel selection using the ADANSA0 register is invalid.

(1) A/D conversion for the single channel selected by the ADCSR.DBLANS[4:0] bits starts when the TRG4AN input sets the ADCSR.ADST bit to 1 (starting A/D conversion).
(2) The result is stored in the corresponding A/D data register (ADDRy) and in A/D data-duplication register A (ADDBLDRA) on completion of the A/D conversion for the channel.
(3) The ADST bit is automatically cleared and the A/D converter enters the waiting state. An S12ADI0 interrupt is not generated at this time, regardless of the setting of the ADCSR.ADIE bit (i.e. whether or not this is set to enable S12ADI0 interrupts in response to scan completion).
(4) When the TRG4BN input sets the ADCSR.ADST bit to 1 (starting A/D conversion), conversion for the single channel selected by the ADCSR.DBLANS[4:0] bits starts.
(5) The result is stored in the A/D data-duplication register (ADDBLDR) and in A/D data-duplication register B (ADDBLDRB) on completion of A/D conversion.
(6) An S12ADI0 interrupt is generated if the setting of the ADCSR.ADIE bit is 1 (S12ADI0 interrupt upon scanning completion enabled).
(7) The ADST bit retains the value 1 (starting A/D conversion) during A/D conversion and is automatically cleared on completion of conversion, after which the A/D converter enters the waiting state.



Note 1. The arrows in this figure indicate instruction execution by software.

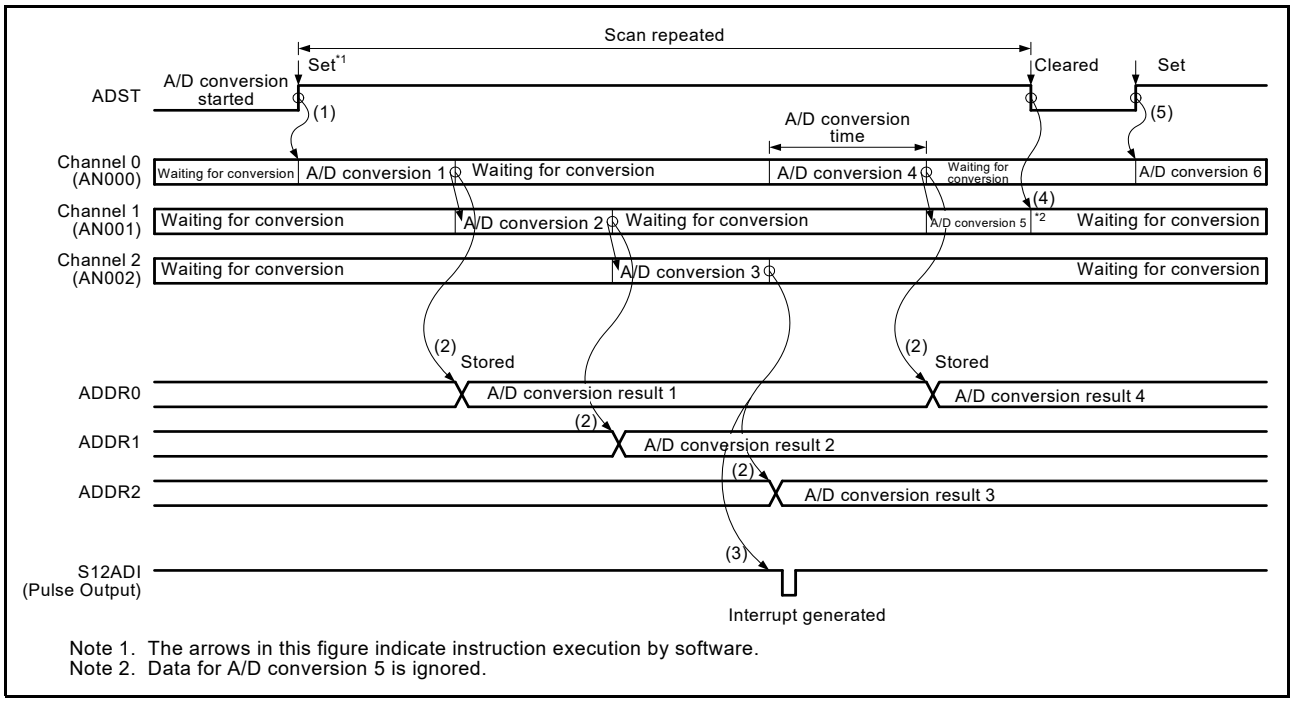
Figure 30.6 Example of Operation in Single Scan Mode
(Duplication Selected for AN003, TRG4AN or TRG4BN Selected)

30.3.3 Continuous Scan Mode

30.3.3.1 Basic Operation

In basic operation of continuous scan mode, A/D conversion is performed repeatedly on the analog input of the channels selected as below.

(1) When the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (MTU, GPT), or an asynchronous trigger input, A/D conversion is performed for ANn pins selected by the ADANSA0 register, starting from the channel with the smallest number n.
(2) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(3) When A/D conversion of all the selected channels is completed, an S12ADI0 interrupt is generated if the ADCSR.ADIE bit is 1 (S12ADI0 interrupt upon scanning completion enabled). The A/D converter sequentially starts A/D conversion for ANn pins selected by the ADANSA0 register, starting from the channel with the smallest number n.
(4) The ADST bit in ADCSR is not automatically cleared and steps 2 and 3 are repeated as long as the bit remains 1 (A/D conversion start). When the ADCSR.ADST bit is set to 0 (A/D conversion stop), A/D conversion stops and the A/D converter enters a waiting state.
(5) When the ADST bit is later set to 1 (A/D conversion start), A/D conversion is started again for ANn pins selected by the ADANSA0 register, starting from the channel with the smallest number n.



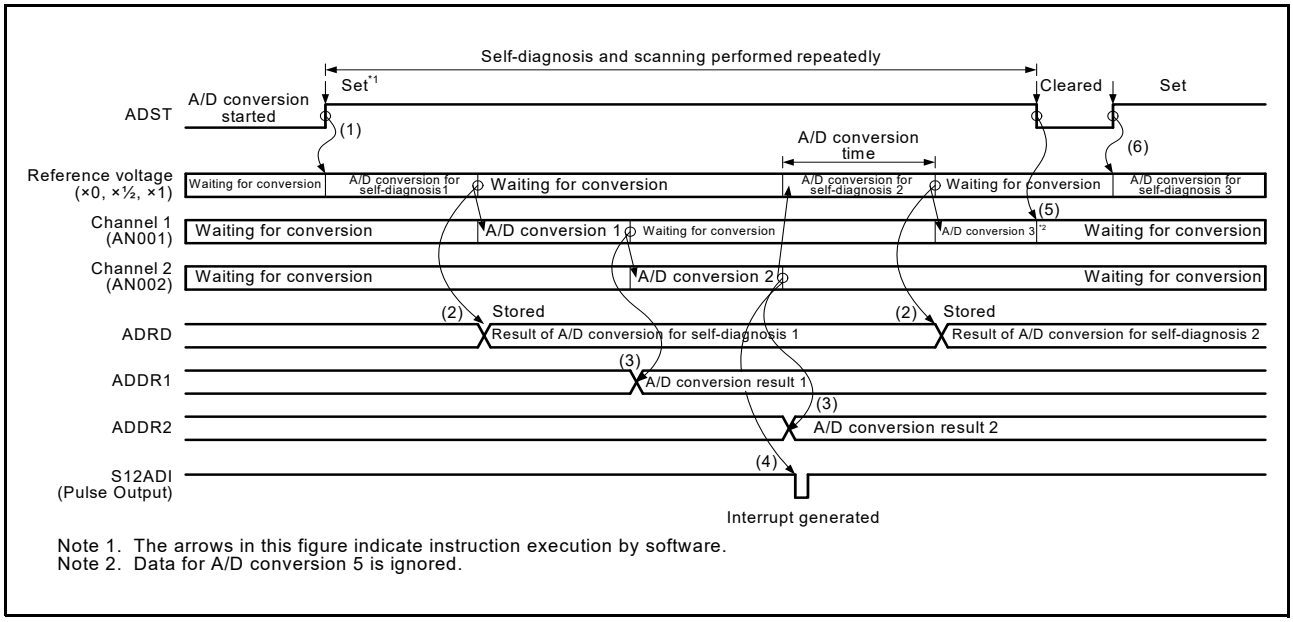
Note 1. The arrows in this figure indicate instruction execution by software.
Note 2. Data for A/D conversion 5 is ignored.

Figure 30.7 Example of Operation in Continuous Scan Mode (Basic Operation: AN000 to AN002 Selected)

30.3.3.2 Channel Selection and Self-Diagnosis

When channels and self-diagnosis are selected at the same time, A/D conversion is first performed for the reference voltage AVcc supplied to the A/D converter, and then A/D conversion is performed on the analog input of the selected channels, which sequence is repeated as below.

(1) A/D conversion due to self-diagnosis is first started when the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (MTU, GPT), or an asynchronous trigger input.
(2) When A/D conversion due to self-diagnosis is completed, the A/D conversion result is stored into the A/D self-diagnosis data register (ADRD). A/D conversion is then performed for ANn pins selected by the ADANSA0 register, starting from the channel with the smallest number n.
(3) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(4) When A/D conversion of all the selected channels is completed, an S12ADI0 interrupt is generated if the ADCSR.ADIE bit is 1 (S12ADI0 interrupt upon scanning completion is enabled). At the same time, the A/D converter starts A/D conversion due to self-diagnosis and then starts A/D conversion on ANn pins selected by the ADANSA0 register, starting from the channel with the smallest number n.
(5) The ADST bit in ADCSR is not automatically cleared and steps 2 to 4 are repeated as long as the bit remains 1. When the ADST bit in ADCSR is set to 0 (A/D conversion stop), A/D conversion stops and the A/D converter enters a waiting state.
(6) When the ADST bit is later set to 1 (A/D conversion start), conversion is started again from the A/D conversion due to self-diagnosis.



Note 1. The arrows in this figure indicate instruction execution by software.
Note 2. Data for A/D conversion 5 is ignored.

Figure 30.8 Example of Operation in Continuous Scan Mode
(Basic Operation; AN001 and AN002 Selected + Self-Diagnosis)

30.3.4 Group Scan Mode

30.3.4.1 Basic Operation

Either two (group A, B) or three (group A, B, C) can be selected for the number of groups to be used in group scan mode. In basic operation of group scan mode, A/D conversion is performed once on the analog inputs of all the specified channels in group A and group B or group A, group B, and group C after scanning is started by a synchronous trigger as below. Scan operation of each group is similar to the scan operation in single scan mode.
The synchronous triggers of group A, B, and C can be selected using the TRSA[5:0], TRSB[5:0], and TRSC[5:0] bits in ADSTRGR, respectively. The different triggers should be set for group A, group B, and group C to prevent simultaneous scan operation of group A, group B, and group C. Software trigger should not be used.
The ADANSA0 register is used to select the channels subject to scan operation for group A. The ADANSB0 register is used to select the channels subject to scan operation for group B. The ADANSC0 register is used to select the channels subject to scan operation for group C.
When self-diagnosis is selected in group scan mode, self-diagnosis is separately executed for group A and group B or group A, group B, and group C.
The following describes operation in group scan mode using a trigger of the MTU. Specifically, the TRG4AN, TRG4BN, and TRG4ABN triggers of the MTU are assumed to be used to start conversion of group A, group B, and group C, respectively.

(1) Scanning of group A is started by the TRG4AN trigger of the MTU.
(2) When group A scanning is completed, an S12ADI0 interrupt is generated if the ADIE bit in ADCSR is 1 (S12ADI0 interrupt upon scanning completion is enabled).
(3) Scanning of group B is started by the TRG4BN trigger of the MTU.
(4) When group B scanning is completed, a S12GBADI0 interrupt is generated if the GBADIE bit in ADCSR is 1 (S12GBADI0 interrupt upon scanning completion is enabled).
(5) Scanning of group C is started by the TRG4ABN trigger of the MTU.
(6) When group C scanning is completed, a S12GCADI0 interrupt is generated if the GBADIE bit in ADCSR is 1 (S12GCADI0 interrupt upon scanning completion is enabled).

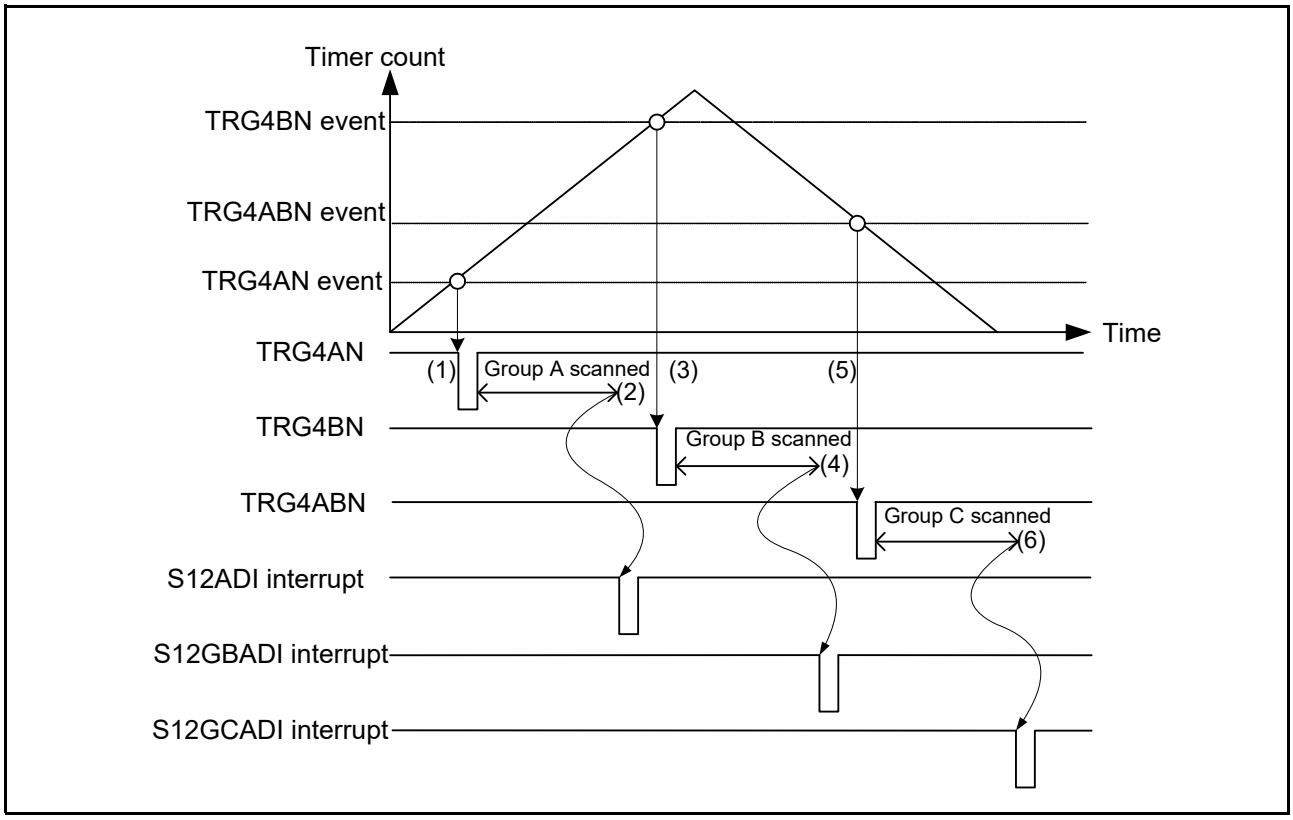


Figure 30.9 Example of Operation in Group Scan Mode (Basic Operation: Triggers from MTU Used)

30.3.4.2 A/D Conversion in Double Trigger Mode

When double trigger mode is selected in group scan mode, A/D conversion is performed for two rounds of single scan operation started by a synchronous trigger (MTU, GPT) as a sequence for group A. For group B and group C, single scan operation started by a synchronous trigger (MTU, GPT) is performed once.
In group scan mode, the synchronous triggers of group A, B, and C can be selected using the TRSA[5:0], TRSB[5:0], and TRSC[5:0] bits in ADSTRGR, respectively. The different triggers should be selected for group A, group B, and group C to prevent simultaneous A/D conversion of group A, group B and group C. Software trigger, or asynchronous trigger (ADTRG#) should not be used.
When TRG4AN or TRG4BN, TRG7AN or TRG7BN, ADTRGA0 or ADTRGB0, ADTRGA1 or ADTRGB1, ADTRGA2 or ADTRGB2, or ADTRGA3 or ADTRGB3 is selected as the synchronous trigger for group A (the ADSTRGR.TRSA[5:0] bits), operation proceeds in extended double trigger mode.
The DBLANS[4:0] bits in the ADCSR register are used to select the channels subject to scan operation for group A. The ADANSB0 register is used to select the channels subject to scan operation for group B. The ADANSC0 register is used to select the channels subject to scan operation for group C.
When double trigger mode is selected in group scan mode, self-diagnosis cannot be selected.
Duplication of A/D conversion data is enabled by setting the channel numbers to be duplicated to the DBLANS[4:0] bits in ADCSR and setting the DBLE bit in ADCSR to 1.
The following describes operation in group scan mode with double trigger mode using a trigger of the MTU. Specifically, the TRG4ABN, TRGA0N, and TRGA1N triggers of the MTU are assumed to be used to start conversion of group A, group B, and group C, respectively.

(1) Scanning of group C is started by the TRGA1N trigger of the MTU.
(2) When group C scanning is completed, a S12GCADI0 interrupt is generated if the GCADIE bit in ADGCTRGR is 1 (S12GCADI0 interrupt upon scanning completion is enabled).
(3) Scanning of group B is started by the TRGA0N trigger of the MTU.
(4) When group B scanning is completed, a S12GBADI0 interrupt is generated if the GBADIE bit in ADCSR is 1 (S12GBADI0 interrupt upon scanning completion is enabled).
(5) The first scanning of group A is started by the first TRG4ABN trigger of the MTU.
(6) When the first scanning of group A is completed, the conversion result is stored into the corresponding A/D data register (ADDRy); an S12ADI0 interrupt is not generated irrespective of the ADIE bit setting in ADCSR.
(7) The second scanning of group A is started by the second TRG4ABN trigger of the MTU.
(8) When the second scanning of group A is completed, the conversion result is stored into the ADDBLDR register. An S12ADI0 interrupt is generated if the ADIE bit in ADCSR is 1 (S12ADI0 interrupt is enabled).
(9) The second scanning of group B is started by the second TRGA0N trigger of the MTU.
(10) When the second scanning of group B is completed, an S12GBADI0 interrupt is generated if the GBADIE bit in ADCSR is 1 (S12GBADI0 interrupt is enabled).
(11) The second scanning of group C is started by the second TRGA1N trigger of the MTU.
(12) When the second scanning of group C is completed, an S12GCADI0 interrupt is generated if the GCADIE bit in ADGCTRGR is 1 (S12GCADI0 interrupt is enabled).

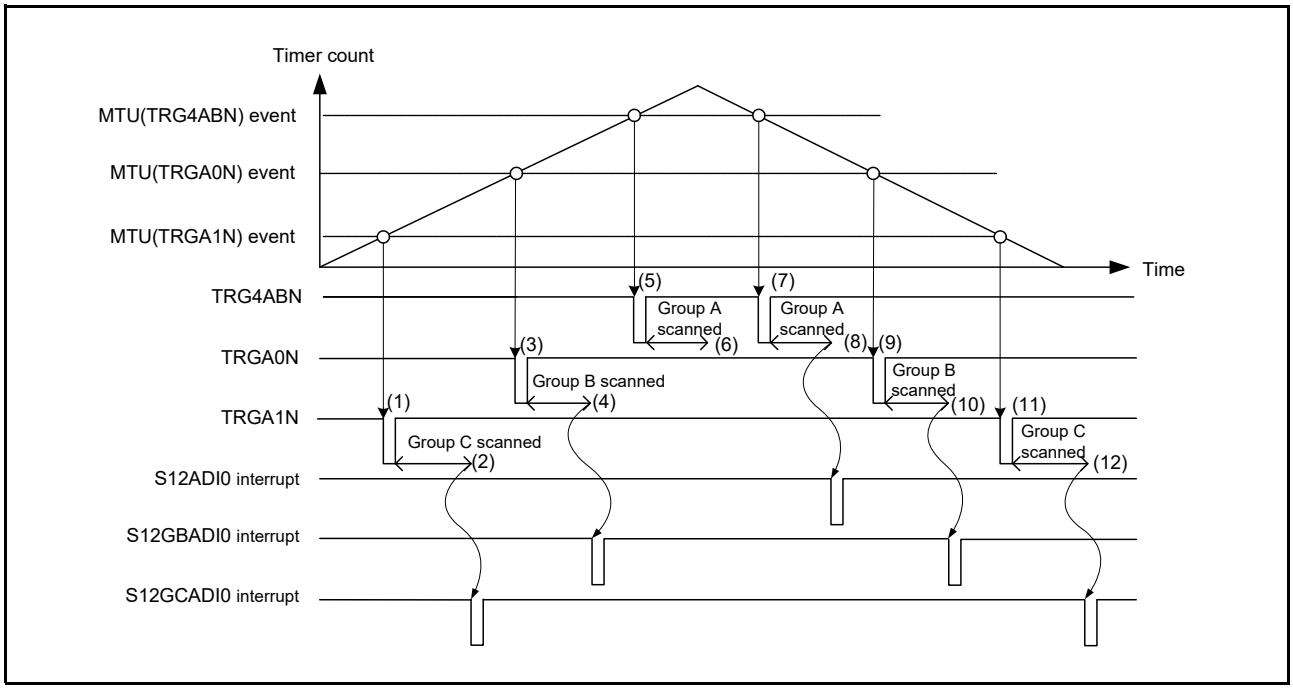


Figure 30.10 Example of Operation in Group Scan Mode with Double Trigger Mode
(Basic Operation: Triggers from MTU Used)

30.3.4.3 Group Priority Control

Setting the ADGSPCR.PGS bit to 1 in group scan mode makes operation proceed under group priority control.
The order of group priority is group A > group B > group C. The number of groups to be used in group scan mode can be selected from either two (group A, B) or three (group A, B, C) by setting ADGCTRGR.GRCE.
When setting the ADGSPCR.PGS bit to 1, follow the procedure described in Figure 30.11. If the procedure is not followed, A/D conversion operation and stored data are not guaranteed.
In operation in basic group scan mode, input of the trigger for the other group during scan operation in group A, group B, or group B is ignored, and scan operations in group A, group B, or group C are the same in single scan mode.
Under group priority control, if a priority group trigger is input during scan operation for low priority group, scan operation for low priority group is discontinued and scan operation for priority group proceeds.
If the setting of the ADGSPCR.GBRSCN bit is 0, the low priority group enters the waiting state on completion of the scan operation for priority group. Additionally, trigger input for the low priority group that occurs during scanning is ignored.
If the setting of the ADGSPCR.GBRSCN bit is 1, the converter automatically restarts scanning for low priority group after completion of the scan operation for priority group.
Additionally, trigger input for the low priority group that occurs during scanning for the priority group is valid, and on completion of the scan for the priority group, scanning for the low priority group is automatically executed.
If the setting of the ADGSPCR.GBRSCN bit is 1 and ADGSPCR.LGRRS bit is 0, the converter restarts scanning for low priority group from the head of the group.
Additionally, when ADGSPCR.LGRRS is set to 1, scan for the low priority group is re-executed from the interrupted channel Note that when the self-diagnosis function is used, scan is re-executed from the interrupted channel on completion of self-diagnosis.

Table 30.13 shows the relation between the setting of the ADGSPCR.GBRSCN bit and operation in response to trigger input during scanning.
When the ADGSPCR.GBRP bit is set to 1, single scan is continuously performed as scanning operation for the lowest priority group.
When setting a trigger in group scan mode, select a synchronous trigger for group A using the ADSTRGR.TRSA[5:0] bits, select a synchronous trigger for group B different from that of group A using the ADSTRGR.TRSB[5:0] bits, and select a synchronous trigger for group C different from those of group A and group B using the ADGCTRGR.TRSC[5:0] bits.
When two groups are enabled (the ADGCTRGR.GRCE bit is set to 0) in group scan mode and the ADGSPCR.GBRP bit is set to 1, set the ADSTRGR.TRSB[5:0] bits to 3Fh.
Additionally, when three groups are enabled (the ADGCTRGR.GRCE bit is set to 1) in group scan mode and the ADGSPCR.GBRP bit is set to 1, set the ADGCTRGR.TRSC[5:0] bits to 3Fh.
Furthermore, as targets for scan operation, select channels for group A, group B, and group C using the ADANSA0, ADANSB0, and ADANSC0 registers, respectively.

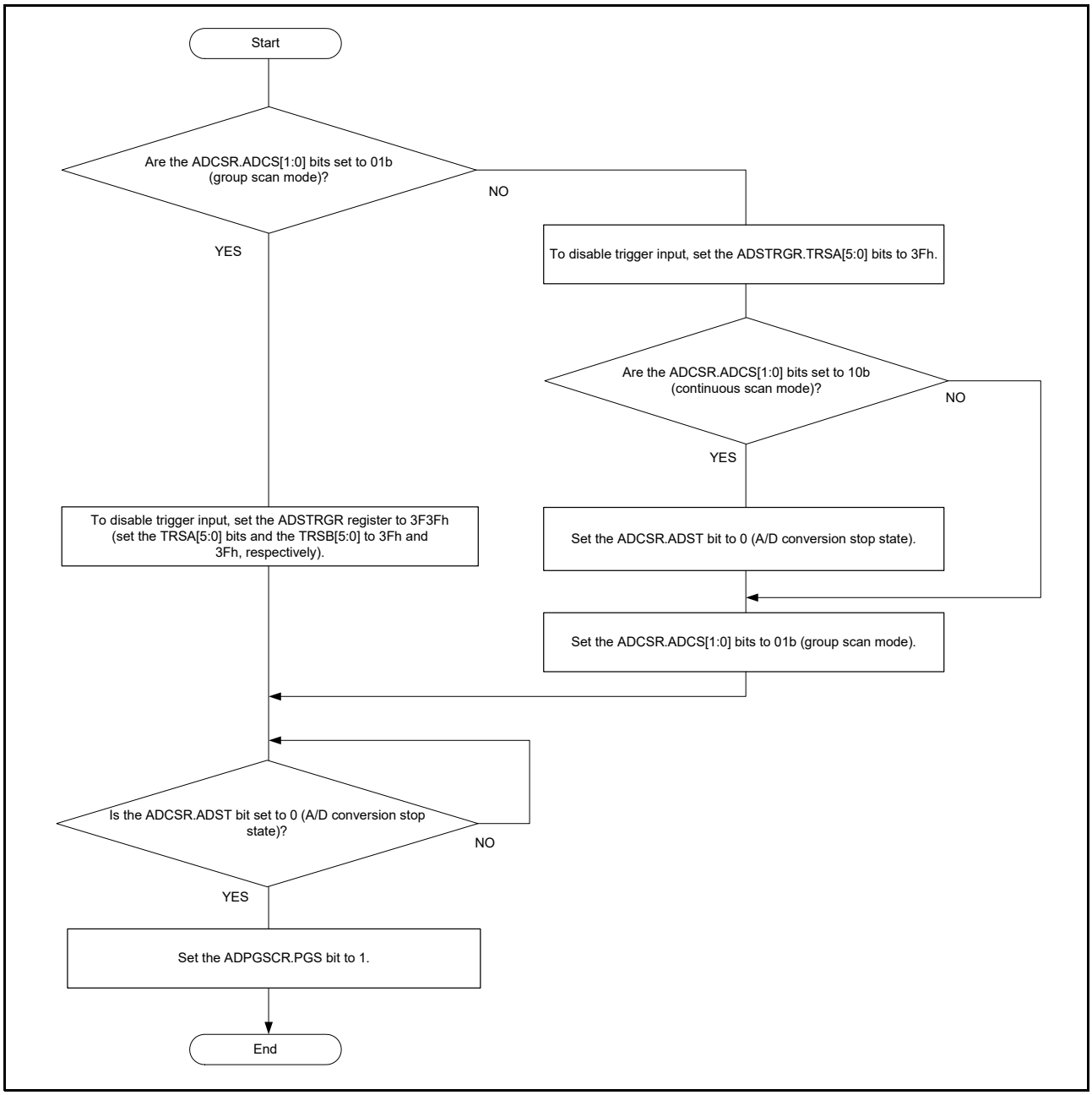


Figure 30.11 Flow of Setting the ADPGSCR.PGS Bit

Table 30.13 Control of Scan Operations According to the Settings of the ADGSPCR.GBRSCN Bit

A/D Conversion Operation	Trigger Input	ADGSPCR.GBRSCN = 0	ADGSPCR.GBRSCN = 1
When A/D conversion for group A is in progress	Input of trigger for group A	Trigger input is ineffective.	Trigger input is ineffective.
	Input of trigger for group B	Trigger input is ineffective.	A/D conversion is performed on group B after A/D conversion on group A is completed.
	Input of trigger for group C	Trigger input is ineffective.	A/D conversion is performed on group C after A/D conversion on group A is completed.
When A/D conversion for group B is in progress	Input of trigger for group A	Conversion for group B that is in progress is discontinued and conversion for group A starts.	• Conversion in progress for group B is discontinued and conversion for group A starts. • Conversion for group B starts after conversion for group A is completed.
	Input of trigger for group B	Trigger input is ineffective.	• Trigger input is ineffective.
	Input of trigger for group C	Trigger input is ineffective.	• Conversion for group C starts after conversion for group B is completed.
When A/D conversion for group C is in progress	Input of trigger for group A	Conversion for group C that is in progress is discontinued and conversion for group A starts.	• Conversion in progress for group C is discontinued and conversion for group A starts. • Conversion for group C starts after conversion for group A is completed.
	Input of trigger for group B	Conversion for group C that is in progress is discontinued and conversion for group B starts.	• Conversion in progress for group C is discontinued and conversion for group B starts. • Conversion for group C starts after conversion for group B is completed.
	Input of trigger for group C	Trigger input is ineffective.	Trigger input is ineffective.

When using group priority control operation mode, refer to the following table, and select the desired operation mode and set the register.

Table 30.14 Group Priority Control Settings and Operation Mode for Two Groups (ADGSPCR.PGS = 1, ADGCTRGR.GRCE = 0)

ADGSPCR GBRSCN	LGRRS	GBRP	Operation Category
0	—	0	Group priority control for two groups (A, B) When a trigger for group A is input, the group B scanning is ended (no re-execution).
1	0	0	Group priority control for two groups (A, B) After the group B scanning is interrupted, the group B scan is restarted from the head of the channels specified in ADANSB0 on completion of the group A scan.
1	1	0	Group priority control for two groups (A, B) After the group B scanning is interrupted, the group B scan is restarted from the interrupted channel*1 of the channels specified in ADANSB0 on completion of the group A scan.
—	0	1	Group priority control for two groups (A, B) Single scan is continuously performed on group B without any start trigger input. After the group B scanning is interrupted, single scan is restarted from the head of the channels specified in ADANSB0 on completion of the group A scan.
1	1	1	Group priority control for two groups (A, B) Single scan is continuously performed on group B without any start trigger input. After the group B scanning is interrupted, single scan is restarted from the interrupted channel*1 of the channels specified in ADANSB0 on completion of the group A scan.

Note: When the self-diagnosis function is enabled (ADCER.DIAGM = 1), A/D conversion for the interrupted channel starts on completion of self-diagnosis.

Table 30.15 Group Priority Control Settings and Operation Mode for Three Groups (ADGSPCR.PGS = 1, ADGCTRGR.GRCE = 1)

ADGSPCR			
GBRSCN	LGRRS	GBRP	Operation Category
0	—	0	Group priority control for three groups (A, B, C) When a trigger for group A is input, the group B scanning is ended (no re-execution). When a trigger for group A or group B is input, the group C scanning is ended (no re-execution).
0	—	1	Group priority control for three groups (A, B, C) When a trigger for group A is input, the group B scanning is ended (no re-execution). Single scan is continuously performed on group C without any start trigger input. After the group C scanning is interrupted, the scan is restarted from the head of the channels specified in ADANSC0 on completion of the group A/B scan.
1	0	0	Group priority control for three groups (A, B, C) After the group B scanning is interrupted, the scan is restarted from the head of the channels specified in ADANSB0 on completion of the group A scan. After the group C scanning is interrupted, the scan is restarted from the head of the channels specified in ADANSC0 on completion of the group A/B scan.
1	1	0	Group priority control for three groups (A, B, C) After the group B scanning is interrupted, the scan is restarted from the interrupted channel* of the channels specified in ADANSB0 on completion of the group A scan. After the group C scanning is interrupted, the scan is restarted from the interrupted channel* of the channels specified in ADANSC0 on completion of the group A/B scan.
1	0	1	Group priority control for three groups (A, B, C) After the group B scanning is interrupted, the scan is restarted from the head of the channels specified in ADANSB0 on completion of the group A scan. Single scan is continuously performed on group C without any start trigger input. After the group C scanning is interrupted, single scan is restarted from the head of the channels specified in ADANSC0 on completion of the group A/B scan.
1	1	1	Group priority control for three groups (A, B, C) After A/D conversion on group B is interrupted, the A/D conversion is restarted from the interrupted channel* of the channels specified in ADANSB0 on completion of the group A scan. Single scan is continuously performed on group C without any start trigger input. After the group C scanning is interrupted, single scan is restarted from the interrupted channel* of the channels specified in ADANSC0 on completion of the group A/B scan.

Note: When the self-diagnosis function is enabled (ADCER.DIAGM = 1), A/D conversion for the interrupted channel starts on completion of self-diagnosis.

(1) Group priority control for two groups (when ADGSPCR.PGS = 1, ADGCTRGR.GRCE = 0)

The following examples 1 to 3 describes the operations in group scan mode under group scan mode group priority operation (ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, and ADGSPCR.LGRRS = 0) when channel 0 is selected for group A and channels 1 to 3 are selected for group B.

Example of operation (1) "Trigger input for group A during group B scanning" with rescan enabled

(1) When input of a trigger for group B sets the ADCSR.ADST bit to 1 (starting A/D conversion), conversion for the ANn channel pins selected in the ADANSB0 register starts in order from the channel with the lowest number.
(2) On completion of A/D conversion for a channel, the result is stored in the corresponding A/D data register (ADDRy).
(3) When a trigger for group A is input during group B scanning, the group B scanning is interrupted while the ADCSR.ADST bit holds 1, and group A scanning for the ANn channels selected in the ADANSA0 register starts in order from the channel with the smallest number n. If A/D conversion is not completed at the time of interruption, the A/D conversion result is not stored in the A/D data register y (ADDRy).
(4) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(5) An S12ADI0 interrupt is generated if the setting of the ADCSR.ADIE bit is 1 (S12ADI0 interrupt upon scanning completion is enabled).
(6) If the ADGSPCR.GBRSCN bit is set to 1 (rescan is performed on the group interrupted due to group priority control), conversion for the ANn channels of group B selected in the ADANSB0 register starts in order from the channel with the lowest number n while the ADCSR.ADST bit holds 1.
(7) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(8) An S12GBADI0 interrupt is generated if the setting of the ADCSR.GBADIE bit is 1 (S12GBADI0 interrupt upon group B scanning completion enabled).
(9) The ADCSR.ADST bit is automatically cleared on completion of all scanning, after which the A/D converter enters the waiting state.

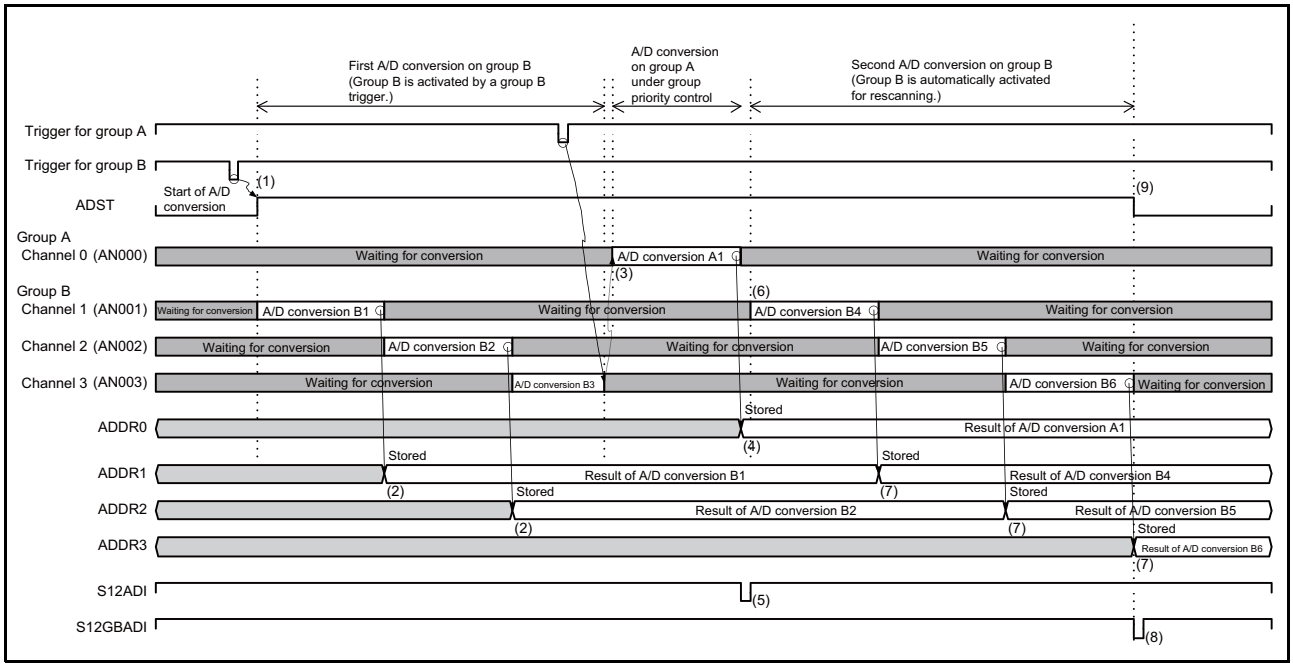


Figure 30.12 Example of Operations under Group Priority Control (1)
"Trigger Input for Group A during Group B Scanning" with Rescan Enabled
(when ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, and ADGSPCR.LGRRS = 0)

Example of operation (2) "Trigger input for group A during group B rescanning" with rescan enabled

Figure 30.13 shows an example when a group A trigger is input again during rescanning operation on group B. Even during rescanning, if a trigger for group A is input, group A scan starts and on completion of the group A scan, group B scan starts.

Operation related to the ADST bit, storage of the A/D conversion result in the A/D data register y (ADDRy), and interrupt request is the same as that of Example of operation (1).

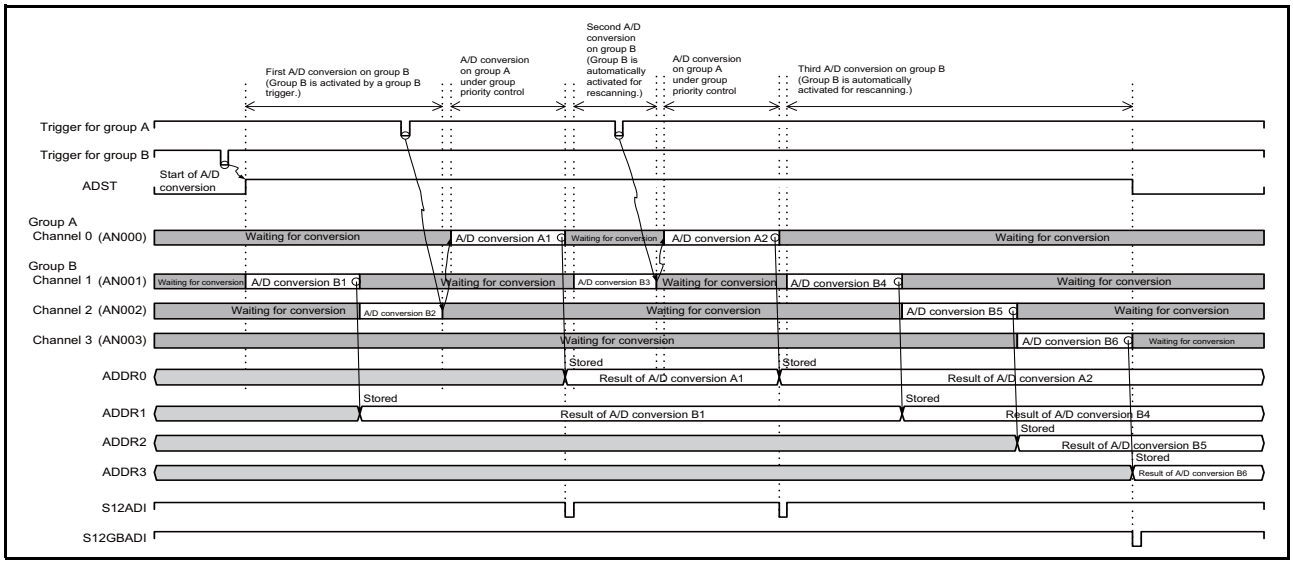


Figure 30.13 Example of Operations under Group Priority Control (2) "Trigger Input for Group A during Group B Rescanning" with Rescan Enabled(when ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 0)

Example of operation (3) "Trigger input for group B during group A scanning" with rescan enabled

This section describes operation when the ADGPSCR.GBRSCN bit is set to 1 (rescan is performed on the group interrupted due to group priority control), and a trigger for group B is input during group A scanning.
When the ADGPSCR.GBRSCN bit is set to 0, any trigger for group B that is input during group A scanning is invalid.

(1) When a group A trigger input sets the ADCSR.ADST bit to 1 (starting A/D conversion), conversion for the ANn channel pins of group A selected in the ADANSA0 register starts in order from the channel with the lowest number n.
(2) When a trigger for group B is input during group A scanning, group B enters the scan-ready state.
(3) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(4) After that, an S12ADI0 interrupt is generated if the setting of the ADCSR.ADIE bit is 1 (S12ADI0 interrupt upon scanning completion is enabled).
(5) On completion of the group A scan, scan for the ANn channels of group B selected in the ADANSB0 register is performed in order from the channel with the smallest number n while the ADCSR.ADST bit holds 1.
(If a trigger for group A is input during group B scanning, as in Example of operation (1), group A scan starts and on completion of the group A scan, group B scan starts.)
(6) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(7) After that, an S12GBADI0 interrupt is generated if the setting of the ADCSR.GBADIE bit is 1 (S12GBADI0 interrupt upon group B scanning completion enabled).
(8) The ADCSR.ADST bit is automatically cleared on completion of all scanning, after which the A/D converter enters the waiting state.

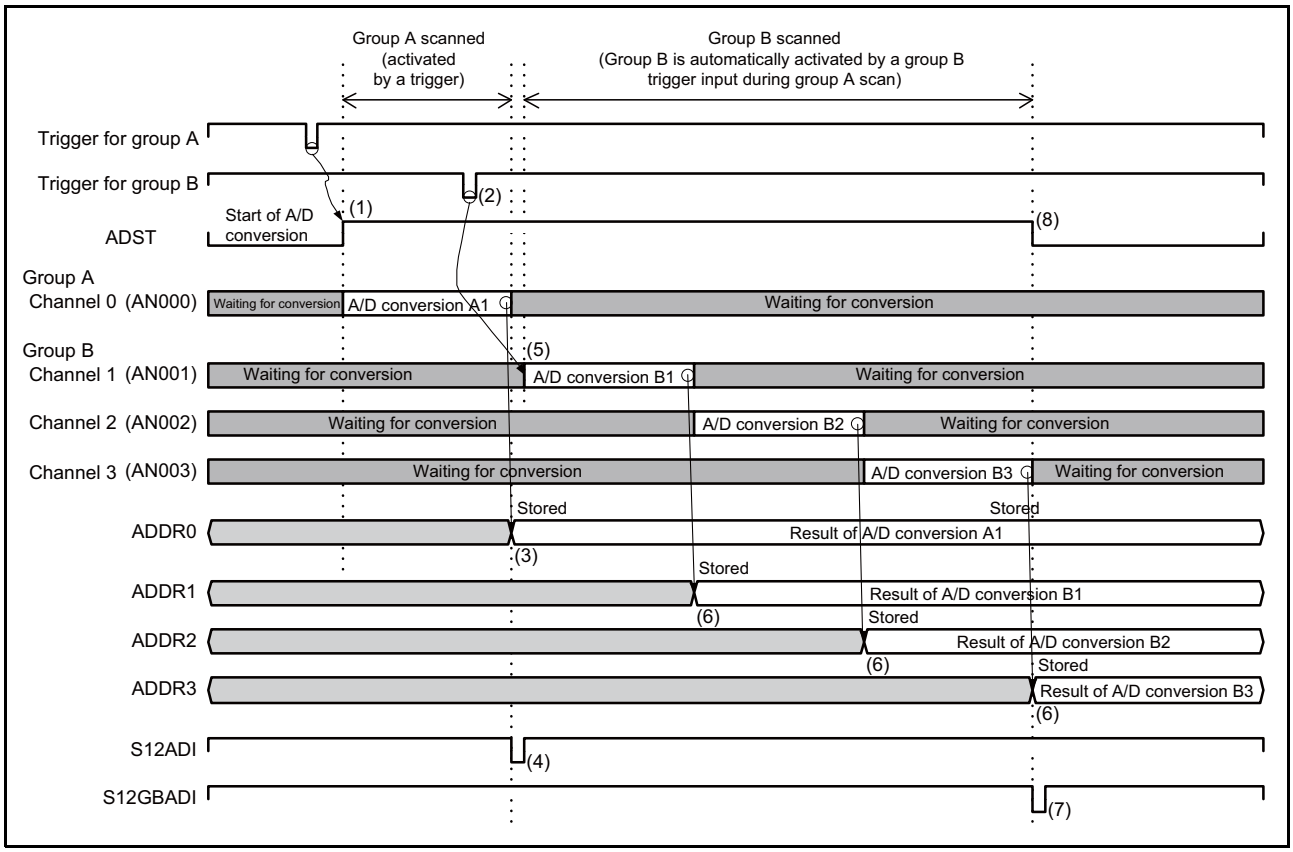


Figure 30.14 Example of Operations under Group Priority Control (3) "Trigger Input for Group B during Group A Scanning" with Rescan Enabled (when ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 0)

Example of operation (4) describes group priority control in group scan mode (ADGSPCR.GBRSCN = 0, ADGSPCR.GBRP = 0, and ADGSPCR.LGRRS = 0) when channel 0 is selected for group A and channels 1 to 3 are selected for group B.

Example of operation (4) "Trigger input for group A during group B scanning" with rescan disabled

(1) When input of a trigger for group B sets the ADCSR.ADST bit to 1 (starting A/D conversion), scan for the ANn channels selected in the ADANSB0 register starts in order from the channel with the smallest number n.
(2) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(3) When a trigger for group A is input during group B scanning, the group B scanning is interrupted while the ADCSR.ADST bit holds 1, and group A scan for the ANn channels selected in the ADANSA0 register starts in order from the channel with the smallest number n. If A/D conversion is not completed at the time of interruption, the A/D conversion result is not stored in the A/D data register y (ADDRy).
(4) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(5) On completion of the group A scan, an S12ADI0 interrupt request is generated if the ADCSR.ADIE bit is set to 1 (S12ADI0 interrupt upon scanning completion is enabled).
(6) The ADST bit is automatically cleared on completion of the group A scan and the A/D converter enters the waiting state. Group B scan is not performed until the next input of a trigger for group B.

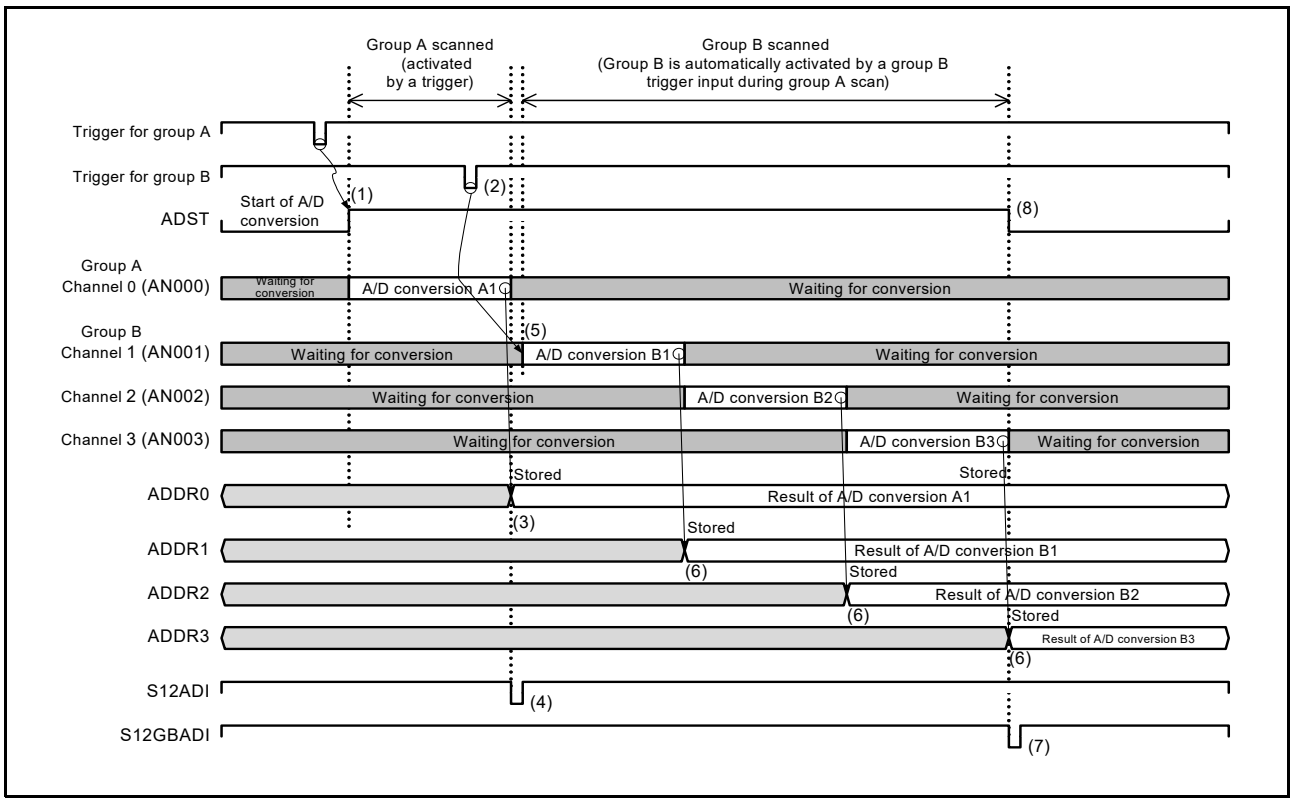


Figure 30.15 Example of Operations under Group Priority Control (4) "Trigger Input for Group A during Group B Scanning" with Rescan Disabled (when ADGSPCR.GBRSCN = 0, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 0)

Example of operation (5) describes group priority control in group scan mode (ADGSPCR.GBRP = 1 and ADGSPCR.LGRRS = 0) when channel 0 is selected for group A and channels 1 and 2 are selected for group B. When the ADGCTRGR.GRCE bit is set to 1, single scan is performed continuously on group C and group B scan starts in response to trigger input.

Example of operation (5) "Continuous Operation of Single Scan for Group B"

(1) When ADGSPCR.GBRP is set to 1, the ADCSR.ADST bit is set to 1 (starting A/D conversion), and scan for the ANn channels selected in the ADANSB0 register starts in order from the channel with the smallest number n.
(2) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(3) When a trigger for group A is input during group B scanning, the group B scanning is interrupted while the ADCSR.ADST bit holds 1, and group A scan for the ANn channels selected in the ADANSA0 register starts in order from the channel with the smallest number n. If A/D conversion is not completed at the time of interruption, the A/D conversion result is not stored in the A/D data register y (ADDRy).
(4) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(5) On completion of the group A scan, an S12ADI0 interrupt request is generated if the ADCSR.ADIE bit is set to 1 (S12ADI0 interrupt upon scanning completion is enabled).
(6) If the ADGSPCR.GBRP bit is set to 1 (performing single scan continuously), group B scan for the ANn channels selected in the ADANSB0 register restarts in order from the channel with the smallest number n while the ADCSR.ADST bit holds 1.
(7) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(8) An S12GBADI0 interrupt request is generated if the ADCSR.GBADIE bit is set to 1 (S12GBADI0 interrupt upon group B scanning completion is enabled).
(9) If the ADGSPCR.GBRP bit is set to 1 (performing single scan continuously), group B scan for the An channels selected in the ADANSB0 register restarts in order from the channel with the smallest number n while the ADCSR.ADST bit holds 1.

Disable group B trigger input to perform a single scan operation continuously on group B.
Steps 6 to 9 are repeated as long as the ADGSPCR.GBRP bit is set to 1. Clearing of the ADCSR.ADST bit is prohibited while the ADGSPCR.GBRP bit is set to 1. Follow the procedure for clearing operation by software through the ADCSR.ADST bit, shown in section 30.5.2, Notes on Stopping A/D Conversion, to forcibly stop scanning when ADGSPCR.GBRP = 1.

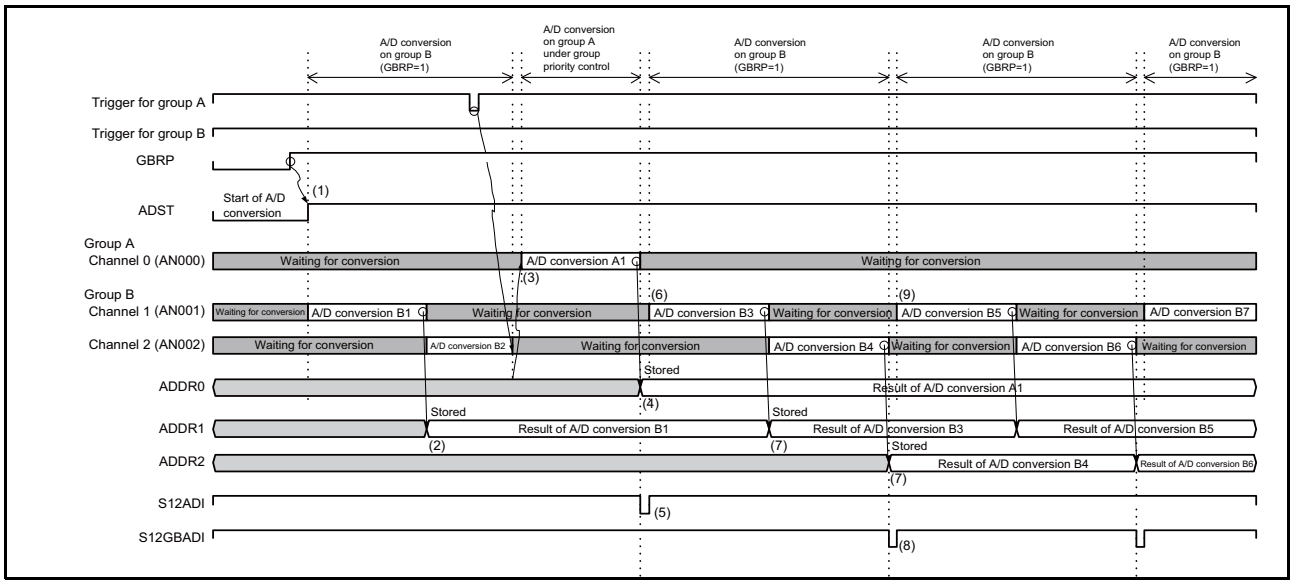


Figure 30.16 Example of Operations under Group Priority Control (5) "Continuous Operation of Single Scan for Group B" (when ADGSPCR.GBRP = 1, ADGSPCR.LGRRS = 1'b0, ADGCTRGR.GRCE = 0)

(2) Group priority control for three groups (when ADGSPCR.PGS = 1, ADGCTRGR.GRCE = 1)
Examples of operation (1) to (3) describe group priority control in group scan mode (ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, and ADGSPCR.LGRRS = 1) when channel 0 is selected for group A, channels 1, 2 are selected for group B, and channels 3, 4 are selected for group C. The priority group means group A and group B for group C, and group A for group B.

Example of operation (1) "Trigger input for priority group during scanning for low priority group" with rescan enabled

(1) When input of a trigger for group C sets the ADCSR.ADST bit to 1 (starting A/D conversion), scan for the ANn channels selected in the ADANSC0 register starts in order from the channel with the smallest number n.
(2) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(3) When a trigger for group B is input during group C scanning, the group C scanning is interrupted while the ADCSR.ADST bit holds 1, and group B scan for the ANn channels selected in the ADANSB0 register starts in order from the channel with the smallest number n. If A/D conversion is not completed at the time of interruption, the A/D conversion result is not stored in the A/D data register y (ADDRy).
(4) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(5) When a trigger for group A is input during group B scanning, the group B scanning is interrupted while the ADCSR.ADST bit holds 1, and group A scan for the ANn channels selected in the ADANSA0 register starts in order from the channel with the smallest number n. If A/D conversion is not completed at the time of interruption, the A/D conversion result is not stored in the A/D data register y (ADDRy).
(6) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(7) An S12ADI0 interrupt request is generated if the ADCSR.ADIE bit is set to 1 (S12ADI0 interrupt upon scanning completion is enabled).
(8) If the ADGSPCR.GBRSCN bit is set to 1 (rescan is performed on the group interrupted due to group priority control), group B scan for the ANn channels selected in the ADANSB0 register restarts in order from the channel with the smallest number n while the ADCSR.ADST bit holds 1. At this time, if the ADGSPCR.LGRRS bit is set to 1, group B scan starts from the channel on which A/D conversion was interrupted.
(9) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(10) An S12GBADI0 interrupt request is generated if the ADCSR.GBADIE bit is set to 1 (S12GBADI0 interrupt upon group B scanning completion is enabled).
(11) If the ADGSPCR.GBRSCN bit is set to 1, group C scan for the ANn channels selected in the ADANSC0 register starts in order from the channel with the smallest number n while the ADCSR.ADST bit holds 1. At this time, if the ADGSPCR.LGRRS bit is set to 1, group C scan starts from the channel on which A/D conversion was interrupted.
(12) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(13) An S12GCADI0 interrupt request is generated if the ADGCTRGR.GCADIE bit is set to 1 (S12GCADI0 interrupt upon group C scanning completion is enabled).
(14) The ADCSR.ADST bit is automatically cleared on completion of all the scans and the A/D converter enters the waiting state.

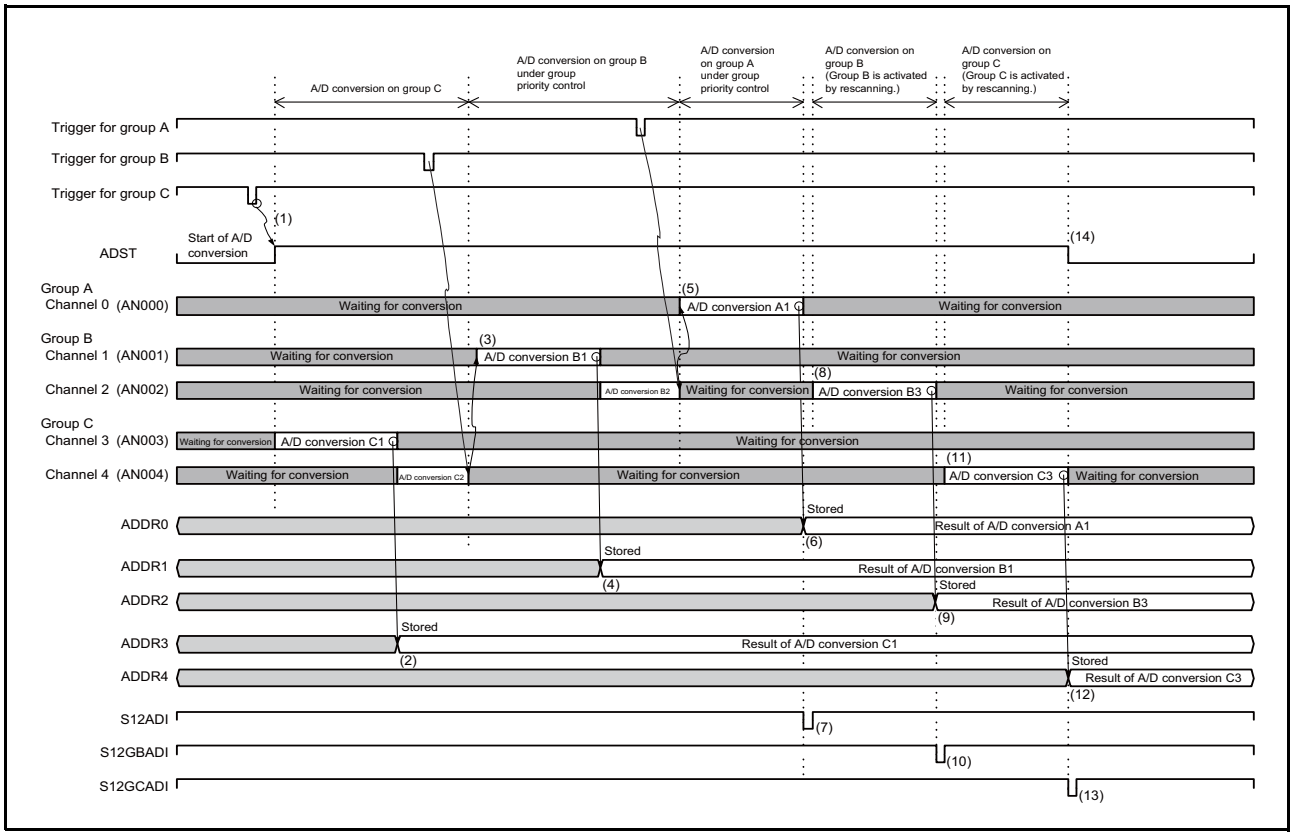


Figure 30.17 Example of Operations under Group Priority Control (1) "Trigger Input for Priority Group during Scanning for Low Priority Group" with Rescan Enabled (when ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 1)

Example of operation (2) "Trigger input for priority group during rescanning for low priority group" with rescan enabled

Figure 30.18 describes the case in which a trigger for group A is input during group B rescanning.
Even during rescanning for a low priority group, if a trigger for a priority group (priority group to group C includes group A and group B, and that to group B includes group A) is input, the priority group scan starts and on completion of the priority group scan, scan for the interrupted low priority group starts.
Operation related to the ADST bit, storage of the A/D conversion result in the A/D data register y (ADDRy), and interrupt request is the same as that of Example of operation (1).

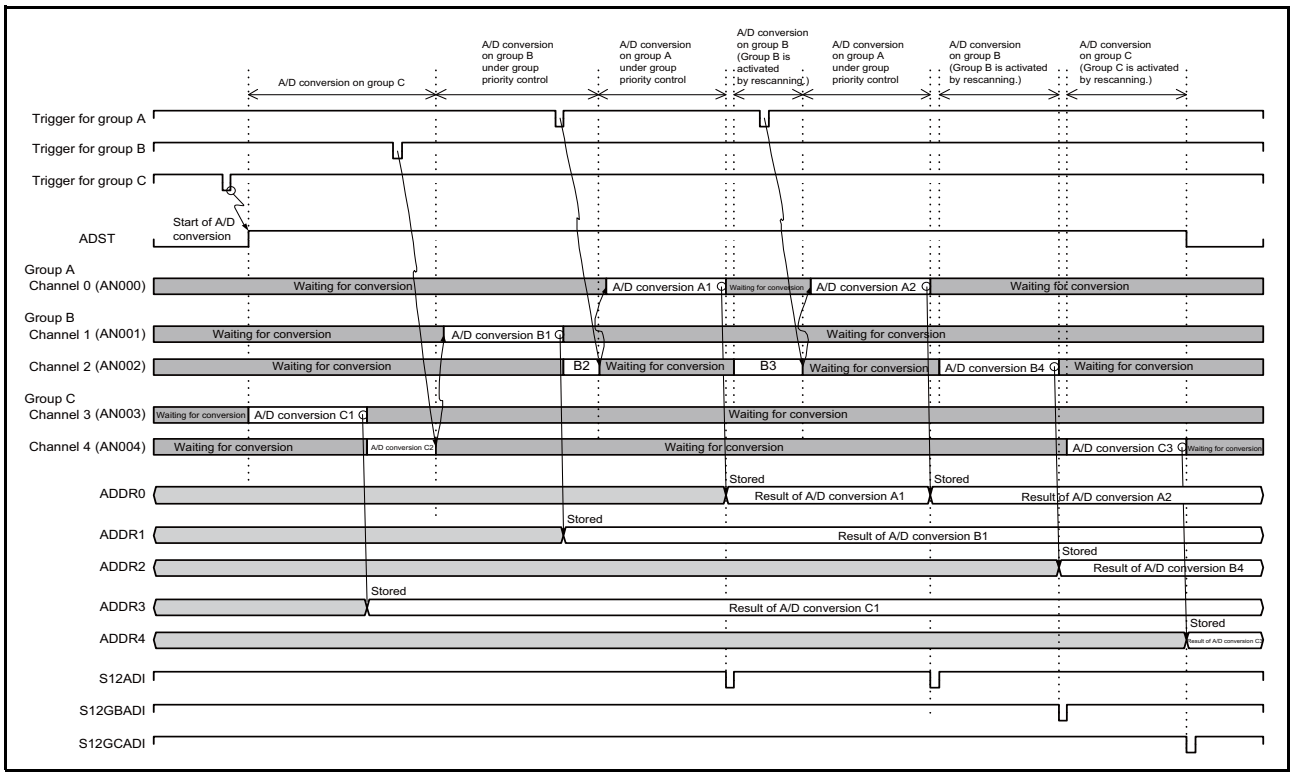


Figure 30.18 Example of Operations under Group Priority Control (2) "Trigger Input for Priority Group during Rescanning for Low Priority Group" with Rescan Enabled (when ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 1)

Example of operation (3) "Trigger input for low priority group during scanning for priority group" with rescan enabled

This section describes operation when the ADGPSCR.GBRSCN bit is set to 1 (rescan is performed on the group interrupted due to group priority control), and a trigger for a low priority group is input during scanning for a priority group. When the ADGPSCR.GBRSCN bit is set to 0, any trigger for a low priority group that is input during scanning for a priority group is invalid.

(1) When input of a trigger for group A sets the ADCSR.ADST bit to 1 (starting A/D conversion), scan for the ANn channels of group A selected in the ADANSA0 register starts in order from the channel with the smallest number n.
(2) When a trigger for group B is input during group A scanning, group B enters the scan-ready state.
(3) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(4) On completion of the group A scan, an S12ADI0 interrupt request is generated if the ADCSR.ADIE bit is set to 1 (S12ADI0 interrupt upon scanning completion is enabled).
(5) On completion of the group A scan, scan for the ANn channels of group B selected in the ADANSB0 register is performed in order from the channel with the smallest number n while the ADCSR.ADST bit holds 1. At this time, if the ADGSPCR.LGRRS bit is set to 1, group B scan starts from the channel on which A/D conversion was interrupted.
(If a trigger for group A is input during group B scanning, as in Example of operation (1), group A scan starts and on completion of the group A scan, group B scan starts.)
(6) When a trigger for group C is input during group B scanning, group C enters the scan-ready state.
(7) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(8) On completion of the group B scan, an S12GBADI0 interrupt request is generated if the ADCSR.GBADIE bit is set to 1 (S12GBADI0 interrupt upon scanning completion is enabled).
(9) On completion of the group B scan, scan for the ANn channels of group C selected in the ADANSC0 register is performed in order from the channel with the smallest number n while the ADCSR.ADST bit holds 1. At this time, if the ADGSPCR.LGRRS bit is set to 1, group C scan starts from the channel on which A/D conversion was interrupted.
(If a trigger for group A or group B is input during group C scanning, as in Example of operation (1), group A or group B scan starts and on completion of the scan, group C scan starts.)
(10) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(11) On completion of the group C scan, an S12GCADI0 interrupt request is generated if the ADGCTRGR.GCADIE bit is set to 1 (S12GCADI0 interrupt upon scanning completion is enabled).
(12) The ADCSR.ADST bit is automatically cleared on completion of all the scans and the A/D converter enters the waiting state.

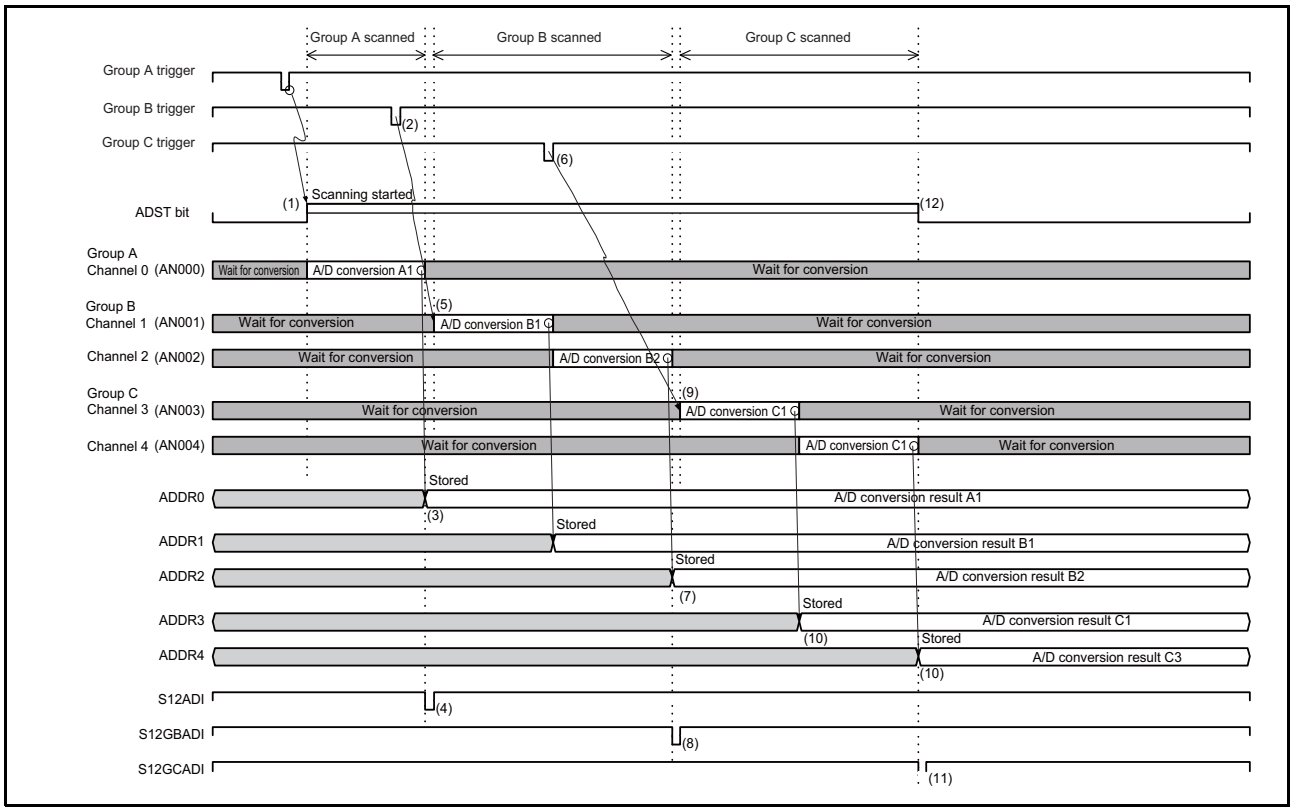


Figure 30.19 Example of Operations under Group Priority Control (3) "Trigger Input for Low Priority Group during Scanning for Priority Group" with Rescan Enabled (when ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 1'b1, ADGCTRGR.GRCE = 1)

Examples of operation (4) describes group priority control in group scan mode (ADGSPCR.GBRSCN = 0, ADGSPCR.GBRP = 0, and ADGSPCR.LGRRS = 0) when channel 0 is selected for group A, channels 1, 2 are selected for group B, and channels 3, 4 are selected for group C.

Example of operation (4) "Trigger input for priority group during scanning for low priority group" with rescan disabled

(1) When input of a trigger for group C sets the ADCSR.ADST bit to 1 (starting A/D conversion), scan for the ANn channels selected in the ADANSC0 register starts in order from the channel with the smallest number n.
(2) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(3) When a trigger for group B is input during group C scanning, the group C scanning is interrupted while the ADCSR.ADST bit holds 1, and group B scan for the ANn channels selected in the ADANSB0 register starts in order from the channel with the smallest number n. If A/D conversion is not completed at the time of interruption, the A/D conversion result is not stored in the A/D data register y (ADDRy).
(4) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(5) When a trigger for group A is input during group B scanning, the group B scanning is interrupted while the ADCSR.ADST bit holds 1, and group A scan for the ANn channels selected in the ADANSA0 register starts in order from the channel with the smallest number n. If A/D conversion is not completed at the time of interruption, the A/D conversion result is not stored in the A/D data register y (ADDRy).
(6) On completion of the group A scan, an S12ADI0 interrupt request is generated if the ADCSR.ADIE bit is set to 1 (S12ADI0 interrupt upon scanning completion is enabled).
(7) The ADST bit is automatically cleared on completion of the group A scan and the A/D converter enters the waiting state. Scan for group B and group C is not performed until the next input of a trigger for each group.

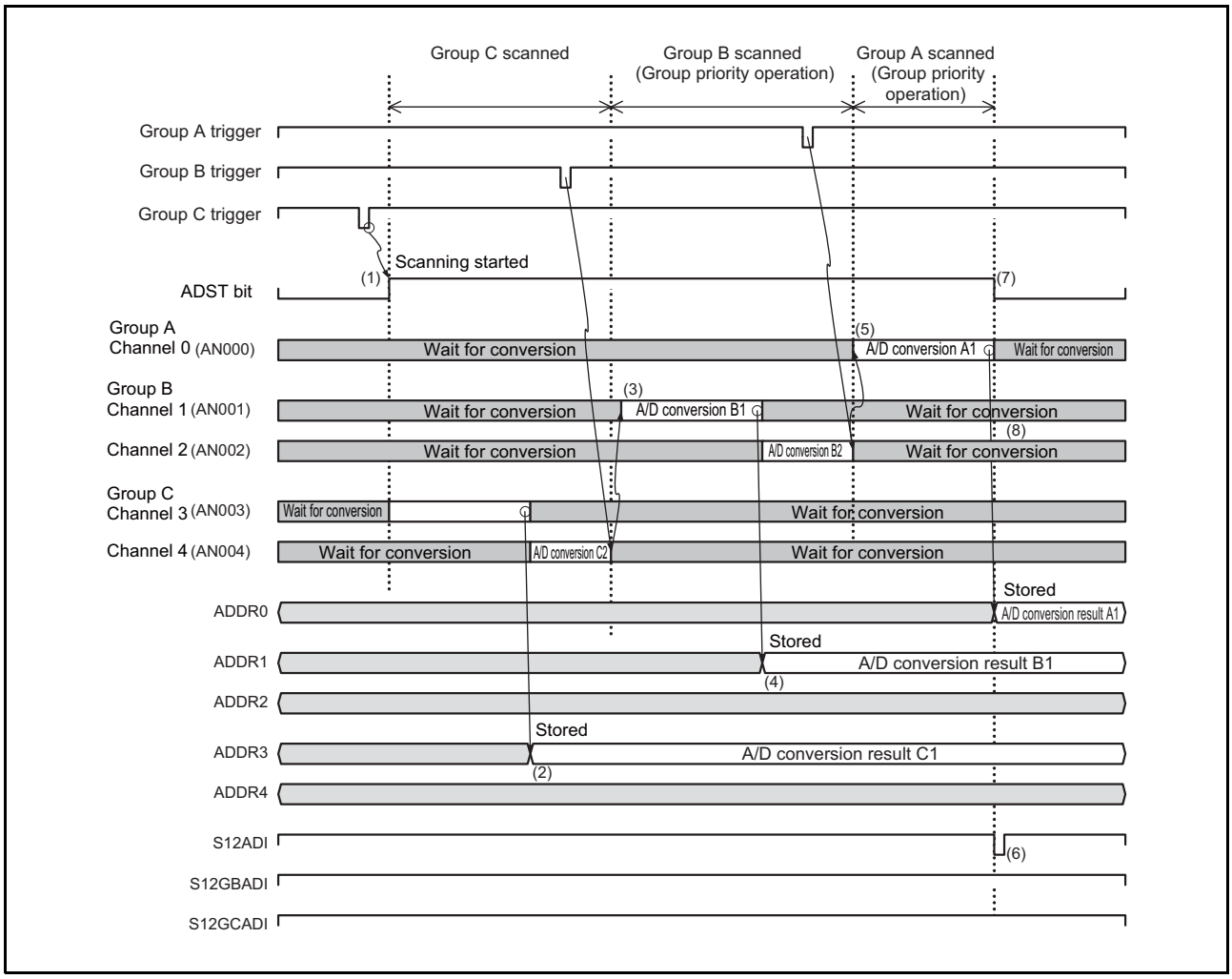


Figure 30.20 Example of Operations under Group Priority Control (4) "Trigger Input for Priority Group during Scanning for Low Priority Group" with Rescan Disabled (when ADGSPCR.GBRSCN = 0, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 1)

Examples of operation (5) describes group priority control in group scan mode (ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 1, and ADGSPCR.LGRRS = 1) when channel 0 is selected for group A, channel 1 is selected for group B, and channels 2, 3 are selected for group C.
When the ADGCTRGR.GRCE bit is set to 0, single scan is performed continuously on group B and input of a trigger for group C is invalid.

Example of operation (5) "Continuous Operation of Single Scan for Group C"

(1) When ADGSPCR.GBRP is set to 1, the ADCSR.ADST bit is set to 1 (starting A/D conversion), and scan for the ANn channels selected in the ADANSC0 register starts in order from the channel with the smallest number n.
(2) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(3) When a trigger for group B is input during group C scanning, the group C scanning is interrupted while the ADCSR.ADST bit holds 1, and group B scan for the ANn channels selected in the ADANSB0 register starts in order from the channel with the smallest number n. If A/D conversion is not completed at the time of interruption, the A/D conversion result is not stored in the A/D data register y (ADDRy).
(4) When a trigger for group A is input during group B scanning, the group B scanning is interrupted while the ADCSR.ADST bit holds 1, and group A scan for the ANn channels selected in the ADANSA0 register starts in order from the channel with the smallest number n. If A/D conversion is not completed at the time of interruption, the A/D conversion result is not stored in the A/D data register y (ADDRy).
(5) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(6) An S12ADI0 interrupt request is generated if the ADCSR.ADIE bit is set to 1 (S12ADI0 interrupt upon scanning completion is enabled).
(7) If the ADGSPCR.GBRSCN bit is set to 1 (rescan is performed on the group interrupted due to group priority control), group B scan for the ANn channels selected in the ADANSB0 register restarts in order from the channel with the smallest number n while the ADCSR.ADST bit holds 1. At this time, if the ADGSPCR.LGRRS bit is set to 1, group B scan starts from the channel on which A/D conversion was interrupted.
(8) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(9) An S12GBADI0 interrupt request is generated if the ADCSR.GBADIE bit is set to 1 (S12GBADI0 interrupt upon group B scanning completion is enabled).
(10) If the ADGSPCR.GBRSCN bit is set to 1 (rescan is performed on the group interrupted due to group priority control), group C scan for the ANn channels selected in the ADANSC0 register restarts in order from the channel with the smallest number n while the ADCSR.ADST bit holds 1. At this time, if the ADGSPCR.LGRRS bit is set to 1, group C scan starts from the channel on which A/D conversion was interrupted.
(11) On completion of A/D conversion on a single channel, the A/D conversion result is stored in the corresponding A/D data register y (ADDRy).
(12) An S12GCADI0 interrupt request is generated if the ADGSPCR.GBCDIE bit is set to 1 (S12GCADI0 interrupt upon group C scanning completion is enabled).
(13) If the ADGSPCR.GBRP bit is set to 1 (performing single scan continuously), group C scan for the An channels selected in the ADANSC0 register restarts in order from the channel with the smallest number n while the ADCSR.ADST bit holds 1.

Disable group B trigger input to perform a single scan operation continuously on group C.
Steps 13, 11, 12, and then 13 are repeated as long as the ADGSPCR.GBRP bit is set to 1.
Clearing of the ADCSR.ADST bit is prohibited while the ADGSPCR.GBRP bit is set to 1.
Follow the procedure for clearing operation by software through the ADCSR.ADST bit, shown in section 30.5.2, Notes on Stopping A/D Conversion, to forcibly stop scanning when ADGSPCR.GBRP = 1.

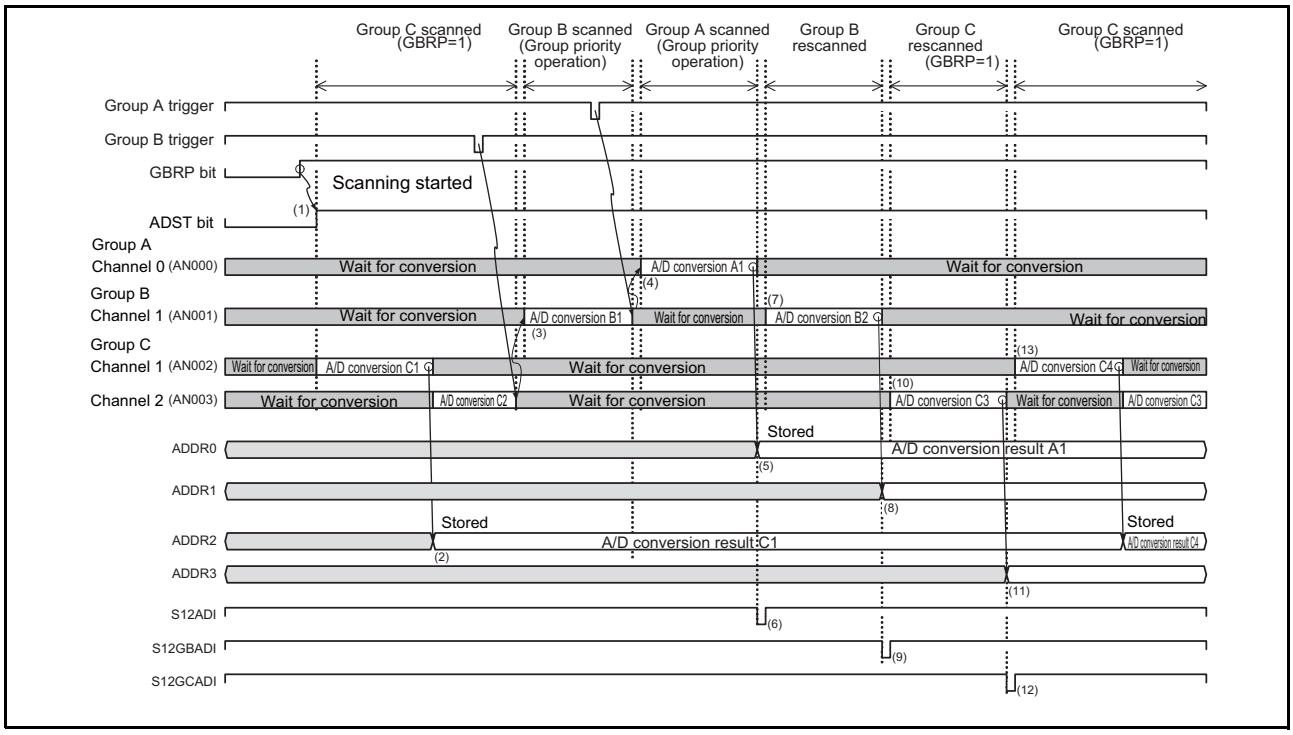


Figure 30.21 Example of Operations under Group Priority Control "Continuous Operation of Single Scan for Group C" (when ADGSPCR.GBRP = 1, ADGSPCR.LGRRS = 1)

30.3.5 Comparison (Window A, Window B)

30.3.5.1 Compare Function Window A/B

Comparison is of a reference value set in a register with the result of A/D conversion on selected channels.
A reference value can be set for each window (A/B).
Self-diagnosis function and double-triggered mode are not available while comparison is in use.
The main difference between window A and window B is that the number of channels which can be selected in window B is one and the interrupt output signal is different.
Operation using comparison in combination with continuous scan mode is described below.

(1) A/D conversion is started in the order of the selected channels when ADCSR.ADST is set to 1 (to start A/D conversion) by software, or in response to a synchronous trigger (MTU, GPT) or asynchronous trigger.
(2) When A/D conversion is completed, the result is stored in the A/D Data Register y (ADDRy).
When ADCMPCR.CMPAE = 1'b1, if the channel is set to a target for window A in the ADCMPANSRy and ADCMPANSER registers, the result is compared with the ADCMPDR0/1 register settings. If ADCMPCR.CMPAE = 1'b1, the register is selected for Window-A by the settings of the ADCMPANSRy and ADCMPANSER registers, its value is then compared with that of the ADCMPDR0 or ADCMPDR1 registers.
When ADCMPCR.CMPBE = 1'b1, if the channel is set to a target for window B in the ADCMPBNSR register, the result is compared with the ADWINULB/ADWINLLB register settings. If ADCMPCR.CMPBE = 1'b1, the register is selected for Window-B by the settings of the ADCMPBNSR register, its value is then compared with that of the ADWINULB or ADWINLLB registers.
(3) If the result of comparison meets the condition set in the ADCMPLR0 and ADCMPLER registers, the Window-A flag bits ADCMPSR0.CMPSTCHA0n, ADCMPSR1.CMPSTCHA1n, ADCMPSER.CMPSTTSA, and ADCMPSER.CMPSTOCA are set to 1. If the setting of the ADCMPCR.CMPAIE bit is 1 at this time, an S12ADCMPAI0 interrupt request (level) is also generated.
If the result of comparison meets the condition set in the ADCMPBNSR.CMPLB bit, the Window-B flag bit ADCMPBSR.CMPSTB is set to 1. If the setting of the ADCMPCR.CMPBIE bit is 1 at this time, an S12ADCMPBI0 interrupt request (level) is also generated.
(4) When comparison is completed for all selected A/D conversion, A/D conversion is started again.
(5) If the setting of the ADCSR.ADST bit is 0 (A/D conversion stop) after the S12ADCMPAI0 or S12ADCMPBI0 interrupt is accepted, processing proceeds for channels that have the compare flag.
(6) The S12ADCMPAI0 interrupt signal is deasserted when all of the Window-A compare flags have been cleared. The S12ADCMPBI0 interrupt signal is deasserted when all of the Window-B compare flags have been cleared. To start further comparison, start A/D conversion again.

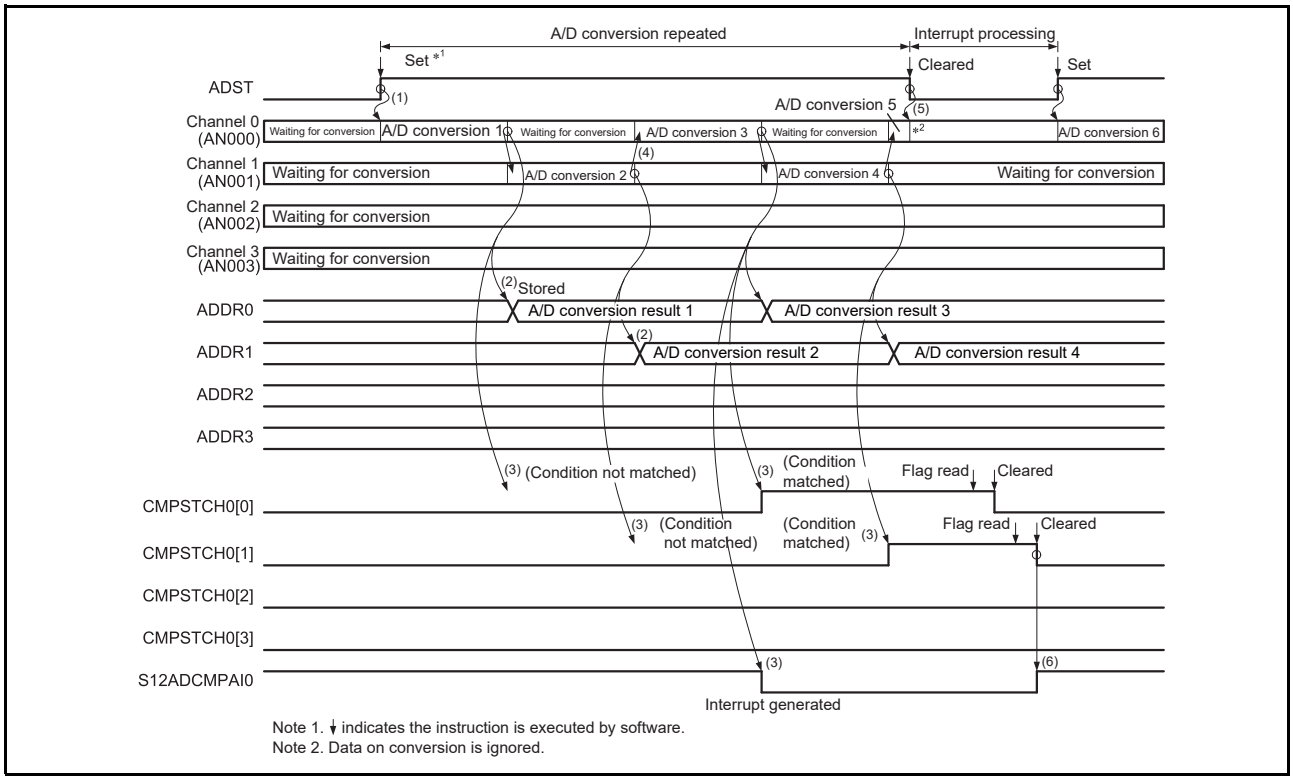


Figure 30.22 Operation Example of Comparison (AN000 to AN003 Comparison Targets)

30.3.5.2 Restrictions on the Compare Function

The following restrictions apply to the compare function:

1. Self-diagnosis function and double-triggered mode are not available while comparison is in use. (ADRD, ADDBLDR, ADDBLDRA, and ADDBLDRB are not subject to the compare function.)
2. An identical channel cannot be set in window A and window B.
3. Specify the settings so that upper-side reference value ≥ lower-side reference value.

30.3.6 Analog Input Sampling and Scan Conversion Time

Scan conversion can be activated either by a software trigger; a synchronous trigger; or ADTRG# (an external trigger). After the start-of-scanning-delay time (tD) has elapsed, processing for disconnection detection assistance, and processing of conversion for self-diagnosis proceed, and this is followed by A/D conversion.

Figure 30.23 shows the scan conversion timing in which scan conversion is activated by a software trigger or a synchronous trigger. Figure 30.24 shows the scan conversion timing in which scan conversion is activated by ADTRG# (an external trigger). The scan conversion time (tSCAN) includes the start-of-scanning-delay time (tD), disconnection detection assistance processing time (tDIS)*2, self-diagnosis A/D conversion processing time (tDIAG)*3, A/D conversion processing time (tCONV), and end-of-scanning-delay time (tED).

The A/D conversion processing time (tCONV) consists of input sampling time (tSPL) and time for conversion by successive approximation (tSAM). The sampling time (tSPL) is used to charge sample-and-hold circuits in the A/D converter. If there is not sufficient sampling time due to the high impedance of an analog input signal source, or if the A/D conversion clock (ADCLK) is slow, sampling time can be adjusted using the ADSSTR register.

The time for conversion by successive approximation (tSAM) is at 13 ADCLK states with 12-bit accuracy selected, 11 ADCLK states with 10-bit accuracy selected, and 9 ADCLK states with 8-bit accuracy selected. Table 30.16 shows the scan conversion time.

The scan conversion time (tSCAN) in single scan mode for which the number of selected channels is n can be determined as follows:

$tSCAN = tD + (tDIS \times n) + tDIAG + (tCONV \times n)^{*3} + tED$

The scan conversion time for the first cycle in continuous scan mode is tSCAN for single scan minus tED. The scan conversion time for the second and subsequent cycles in continuous scan mode is fixed to $(tDIS \times n) + tDIAG + tDSD + (tCONV \times n)^{*3}$.

Note 1. When disconnection detection assistance is not selected, tDIS = 0.
Note 2. When the self-diagnosis function is not used, tDIAG = 0 and tDSD = 0.
Note 3. Although when the sampling time (tSPL) of the selected channels is identical, "tCONV × n" is applied, when it differs according to the channel, the sum of the sampling time (tSPL) and successive approximation (tSAM) of each channel is applied.

Table 30.16 Times for Conversion during Scanning (in Numbers of Cycles of the ADCLK and PCLKH)

Item			Symbol		Type/Conditions			Unit
					Synchronous Trigger*3	External (Asynchronous) Trigger	Software Trigger	
Scan start processing time*1*2	A/D conversion on group under group priority control.	Low priority groups are to be interrupted (The priority group is activated after the low priority group is stopped by an A/D conversion source of the priority group)	tD		2 PCLK + 6 ADCLK	—	—	Cycle
		Low priority groups are not to be interrupted (Activation by an A/D conversion source of the priority group)			2 PCLK + 4 ADCLK	—	—	
	A/D conversion when self-diagnosis is enabled	A/D conversion for self-diagnosis is to be started.			2 PCLK + 6 ADCLK	4 PCLK + 6 ADCLK	6 ADCLK	
	Other than above				2 PCLK + 4 ADCLK	4 PCLK + 4 ADCLK	4 ADCLK	
Disconnection detection assistance processing time			tDIS		The setting of ADNDIS[3:0] (initial value = 00h) × ADCLK			
Self-diagnosis conversion processing time*1	Sampling time		tDIAG	tSPL	The setting of ADSSTR0 (initial value = 0Bh) × ADCLK			
	Time for conversion by successive approximation	12-bit conversion accuracy		tSAM	15 ADCLK			
		10-bit conversion accuracy			13 ADCLK			
		8-bit conversion accuracy			11 ADCLK			
	After completion of self-diagnosis conversion. Normal A/D conversion is to be started.			tDED	2 ADCLK			
	Self-diagnosis conversion is to be started after completion of conversion for continuous scan on the last channel.		tDSD		2 ADCLK			
A/D conversion processing time*1	Sampling time		tCONV	tSPL	The setting of ADSSTRn (n = 0 to 7) (initial value = 0Bh) × ADCLK			
	Time for conversion by successive approximation	12-bit conversion accuracy		tSAM	13 ADCLK			
		10-bit conversion accuracy			11 ADCLK			
		8-bit conversion accuracy			9 ADCLK			
Scan end processing time*1			tED		1 PCLK + 3 ADCLK			

Note 1. Refer to Figure 30.23 and Figure 30.24 for illustration of times tD, tDIAG, tCONV, and tED.
Note 2. This is the maximum time required from software writing or trigger input to A/D conversion start.
Note 3. This does not include the time consumed in the path from timer output to trigger input.

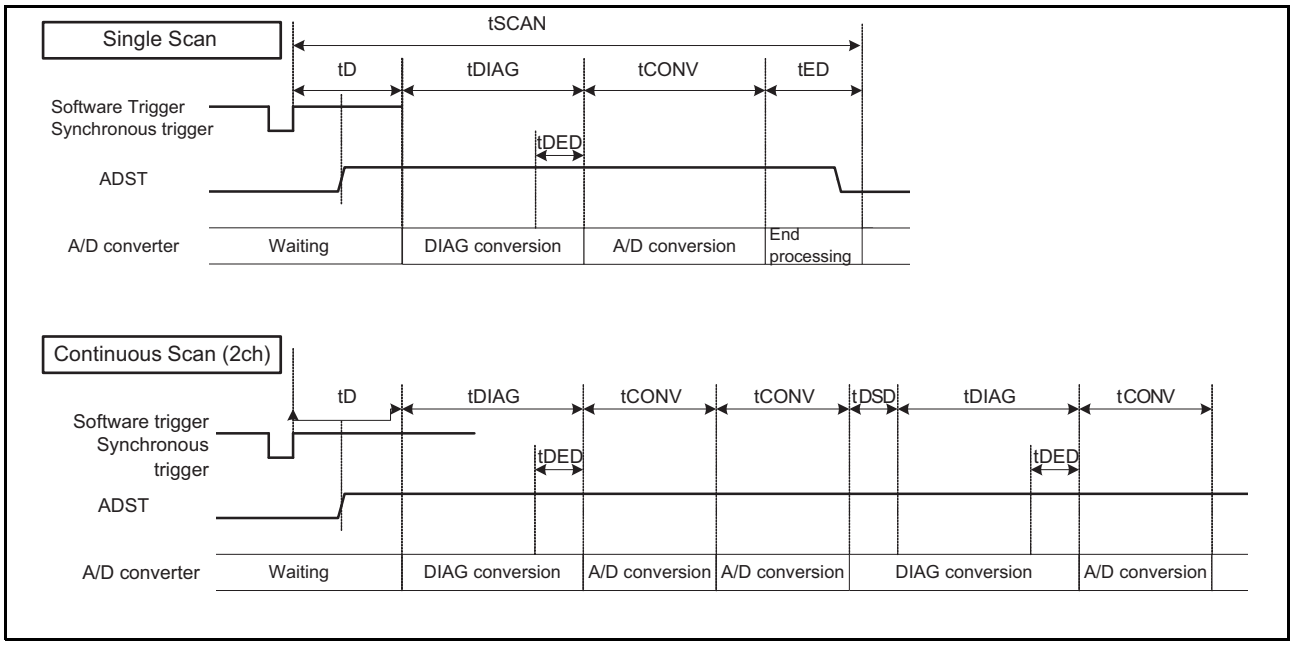


Figure 30.23 Scan Conversion Timing (Activated by Software or Synchronous Trigger Input)

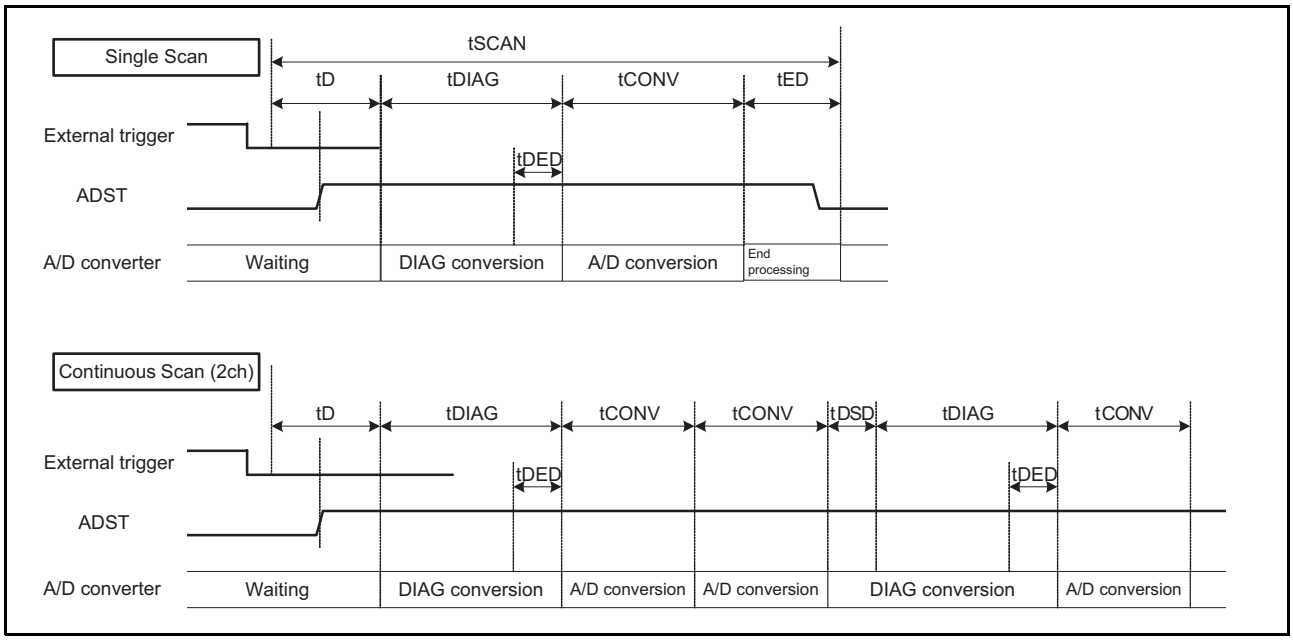


Figure 30.24 Scan Conversion Timing (Activated by Asynchronous Trigger Input (ADTRG#))

30.3.6.1 Timing of Scan Interruption and Start under Group Priority Control

Under group priority control, the timing of scan interruption and start is as follows.

1. Timing of interruption of scanning for the low priority group and start of scan for the priority group
2. Timing of restart of scan for the interrupted low priority group, and timing of start of scan for the low priority group after completion of scan for the high priority group in response to a trigger for the low priority group accepted during scanning for the high priority group
3. Timing of continuous operation of single scan for the low priority group

Figure 30.25 shows these timings, respectively.
The timing of interruption/start of scan for group A and group C or group B and group C is the same as the timing of interruption/start of scan for group A and group B shown in Figure 30.25. The timing of continuous operation of single scan is the same for both group B (using two groups) and group C (using three groups).

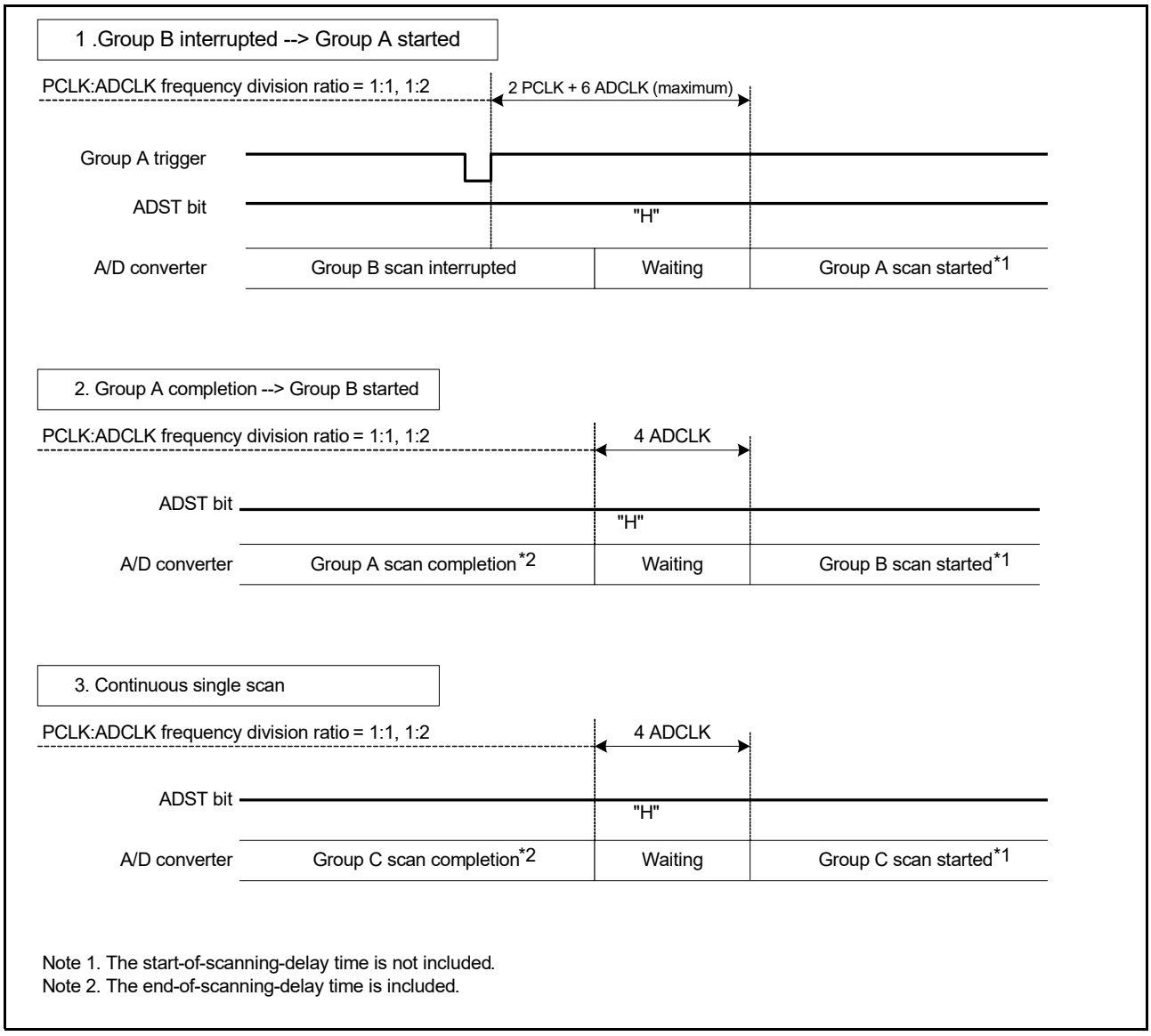


Figure 30.25 Timing of Scan Interruption and Start under Group Priority Control

30.3.7 Usage Example of Register Automatic Clearing Function

Setting the ACE bit in ADCER register to 1 automatically clears the A/D Data Register y (ADDRy, ADRD, ADDBLDR, ADDBLDRA, ADDBLDRB) to 0000h when the A/D data registers are read by the CPU or DMAC.
This function enables detection of update failures of the A/D data registers (ADDRy, ADRD, ADDBLDR, ADDBLDRA, ADDBLDRB). The following describes the examples in which the function to automatically clear the ADDRy register is enabled and disabled.
In a case where the ACE bit in ADCER register is 0 (automatic clearing is disabled), if the A/D conversion result (0222h) is not written to the ADDRy register for some reason, the old data (0111h) will be the ADDRy value. Furthermore, if this ADDRy value is read into a general register using an A/D scan end interrupt, the old data (0111h) can be saved in the general register. When checking whether there is an update failure, it is necessary to frequently save the old data in the RAM or a general register.
In a case where the ACE bit in ADCER register is 1 (automatic clearing is enabled), when ADDRy = 0111h is read by the CPU or DMAC, ADDRy is automatically cleared to 0000h. After that, if the A/D conversion result 0222h cannot be transferred to ADDRy for some reason, the cleared data (0000h) remains as the ADDRy value. If this ADDRy value is read into a general register using an A/D scan end interrupt at this point, 0000h will be saved in the general register. Occurrence of an ADDRy update failure can be determined by simply checking that the read data value is 0000h.

30.3.8 A/D-Converted Value Addition/Average Function

In A/D-converted value addition function, the same channel is A/D-converted 2, 3, 4, or 16 consecutive times and the sum of the converted values is stored in the data register. In A/D-converted value average function, the same channel is A/D converted 2 or 4 consecutive times and the mean of the converted values is stored in the data register. The use of the average of these results can improve the accuracy of A/D conversion, depending on the types of noise components that are present. This function, however, cannot always guarantee an improvement in A/D conversion accuracy.
The A/D-converted value addition/average function can be specified when A/D conversion of the channel select analog input is selected.

Note 1. The addition count can be set to 16 only when 12-bit conversion accuracy is selected.

30.3.9 Disconnection Detection Assist Function

This converter incorporates the function to fix the charge for sampling capacitance to the specified state before start of A/D conversion. This function enables disconnection detection in wiring of analog inputs.
Figure 30.26 illustrates the A/D conversion operation when the disconnection detection assist function is used. Figure 30.27 shows an example of disconnection detection when precharge is selected. Figure 30.28 shows an example of disconnection detection when discharge is selected.

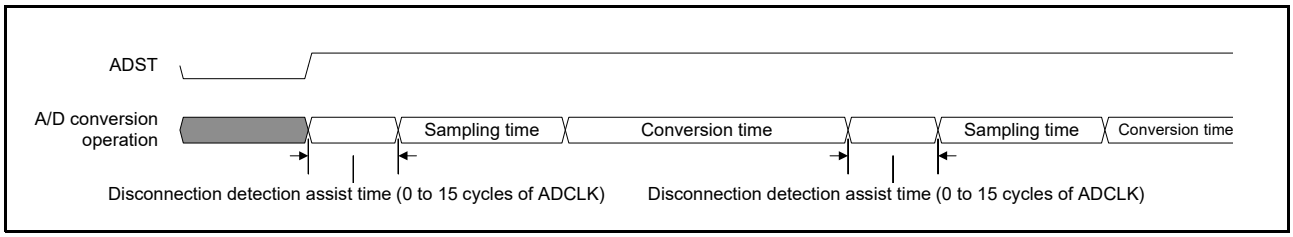


Figure 30.26 Operation of A/D Conversion when the Disconnection Detection Assist Function is Used

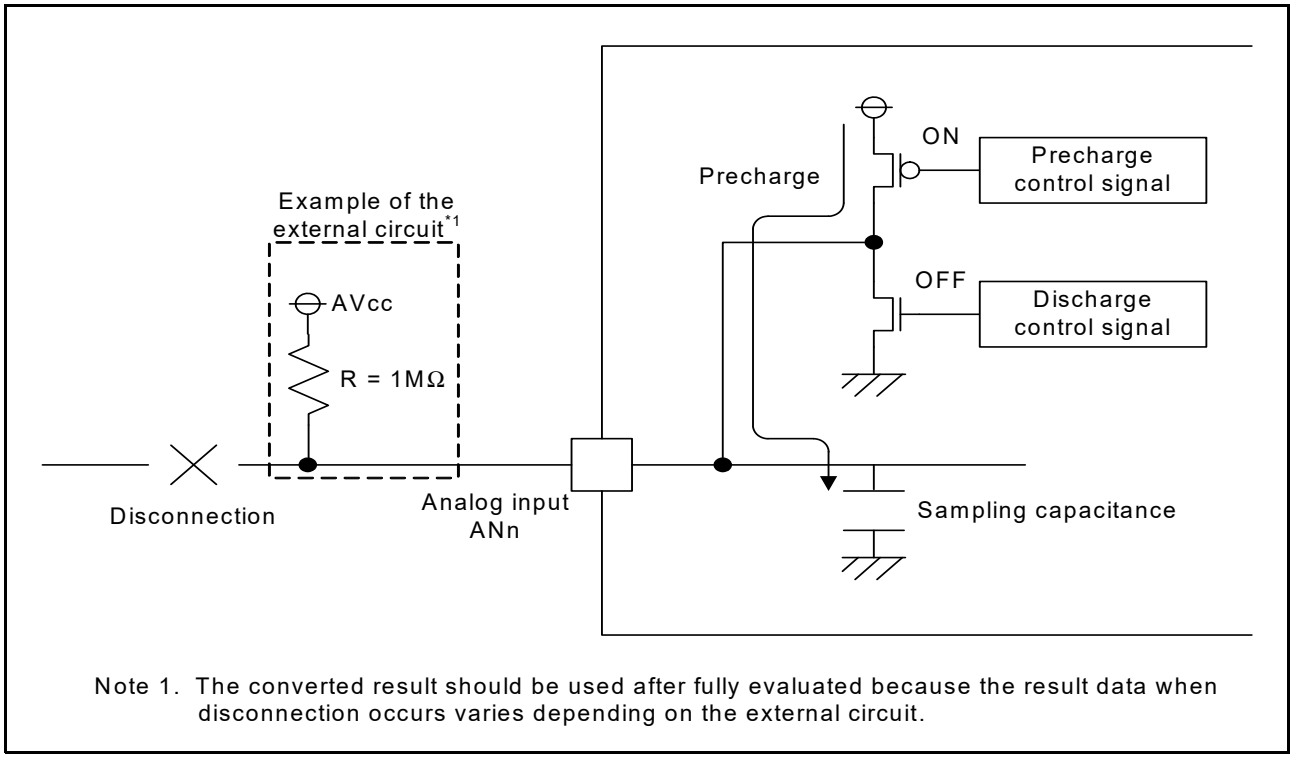
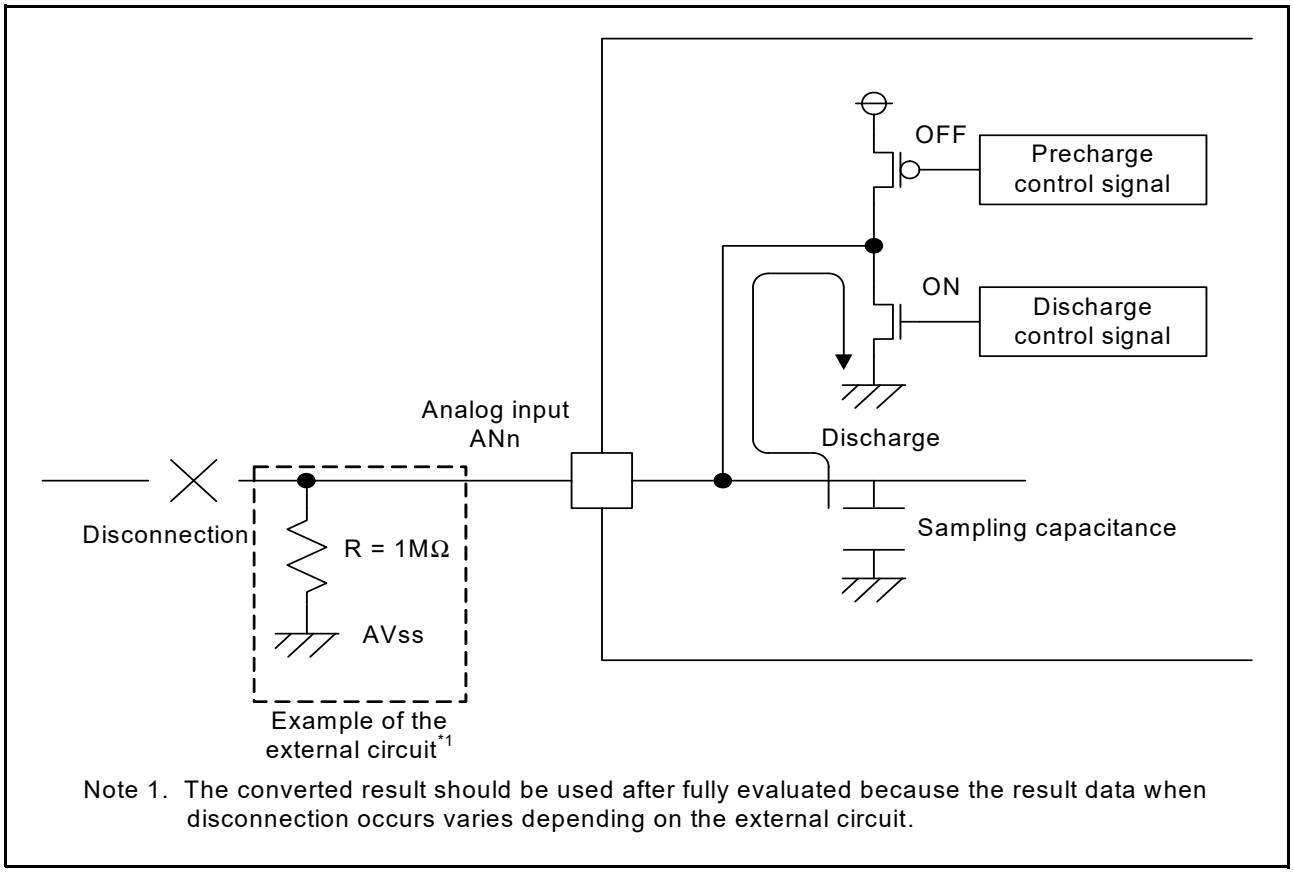


Figure 30.27 Example of Disconnection Detection when Precharge is Selected



Note 1. The converted result should be used after fully evaluated because the result data when disconnection occurs varies depending on the external circuit.

Figure 30.28 Example of Disconnection Detection when Discharge is Selected

30.3.10 Starting A/D Conversion with Asynchronous Trigger

The A/D conversion can be started by the input of an asynchronous trigger.
To start up the A/D converter by an asynchronous trigger, after setting general-purpose I/O pin function, the A/D conversion start trigger select bits (ADSTRGR.TRSA[5:0]) should be set to 0h and a high-level signal should be input to the asynchronous trigger (ADTRG# pin). Both the ADCSR.TRGE and ADCSR.EXTRG bits then should be set to 1.
Figure 30.29 shows a timing of the asynchronous trigger input.
An asynchronous trigger cannot be selected for group B and group C used in group scan mode. For details about the settings of the pin function, see section 51, GPIO.

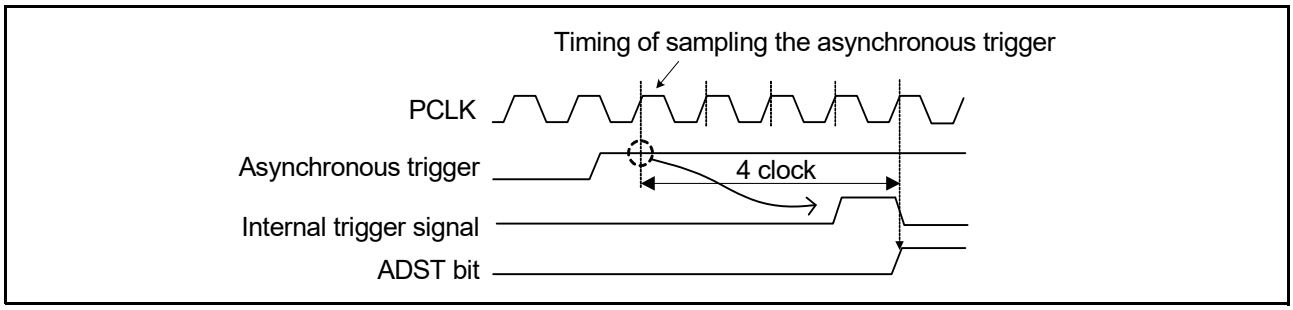


Figure 30.29 Asynchronous Trigger Input Timing

30.3.11 Starting A/D Conversion with Synchronous Trigger from Peripheral Module

The A/D conversion can be started by a synchronous trigger (one pulse of "L" in PCLK synchronization) from MTU or GPT. To start the A/D conversion by a synchronous trigger, the ADCSR.TRGE bit should be set to 1, the ADCSR.EXTRG bit should be cleared to 0, and the relevant sources should be selected by the ADSTRGR.TRSA[5:0] and ADSTRGR.TRSB[5:0] bits.

30.4 Interrupt Sources and DMAC Transfer Requests

30.4.1 Interrupt Requests

The A/D converter can send scan end interrupt requests S12ADI0, S12GBADI0, and S12GCADI0 to the CPU. The module also generates the S12ADCMPAI0 and S12ADCMPBI0 interrupt for the CPU in response to matches with a condition for comparison.
Setting the ADCSR.ADIE bit to 1 and 0 enables and disables an S12ADI0 interrupt, respectively; similarly, setting the ADCSR.GBADIE bit to 1 and 0 enables and disables an S12GBADI0 interrupt, respectively.
Setting the ADCMPCR.CMPAIE bit to 1 and 0 enables and disables an S12ADCMPAI0 interrupt, respectively.
Setting the ADCMPCR.CMPBIE bit to 1 and 0 enables and disables an S12ADCMPBI0 interrupt, respectively.
In addition, the DMAC can be started up when an S12ADI0, an S12GBADI0, or an S12GCADI0 interrupt is generated. Using an S12ADI0, an S12GBADI0, or an S12GCADI0 interrupt to allow the DMAC to read the converted data enables continuous conversion without burden on software.
For details on DMAC settings, see section 9, Direct Memory Access Controller.

30.5 Usage Notes

30.5.1 Notes on Reading Data Registers

The A/D data registers, A/D data duplication registers, A/D data duplication register A, A/D data duplication register B, and A/D self-diagnosis data register should be read in word units. If a register is read twice in byte units, that is, the upper byte and lower byte are separately read, the A/D-converted value having been read first may disagree with the A/D-converted value having been read for the second time. To prevent this, the data registers should never be read in byte units.

30.5.2 Notes on Stopping A/D Conversion

30.5.2.1 A/D Conversion Stopping Procedure

To stop A/D conversion when an asynchronous trigger or a synchronous trigger has been selected as the condition for starting A/D conversion, follow the procedure in Figure 30.30.

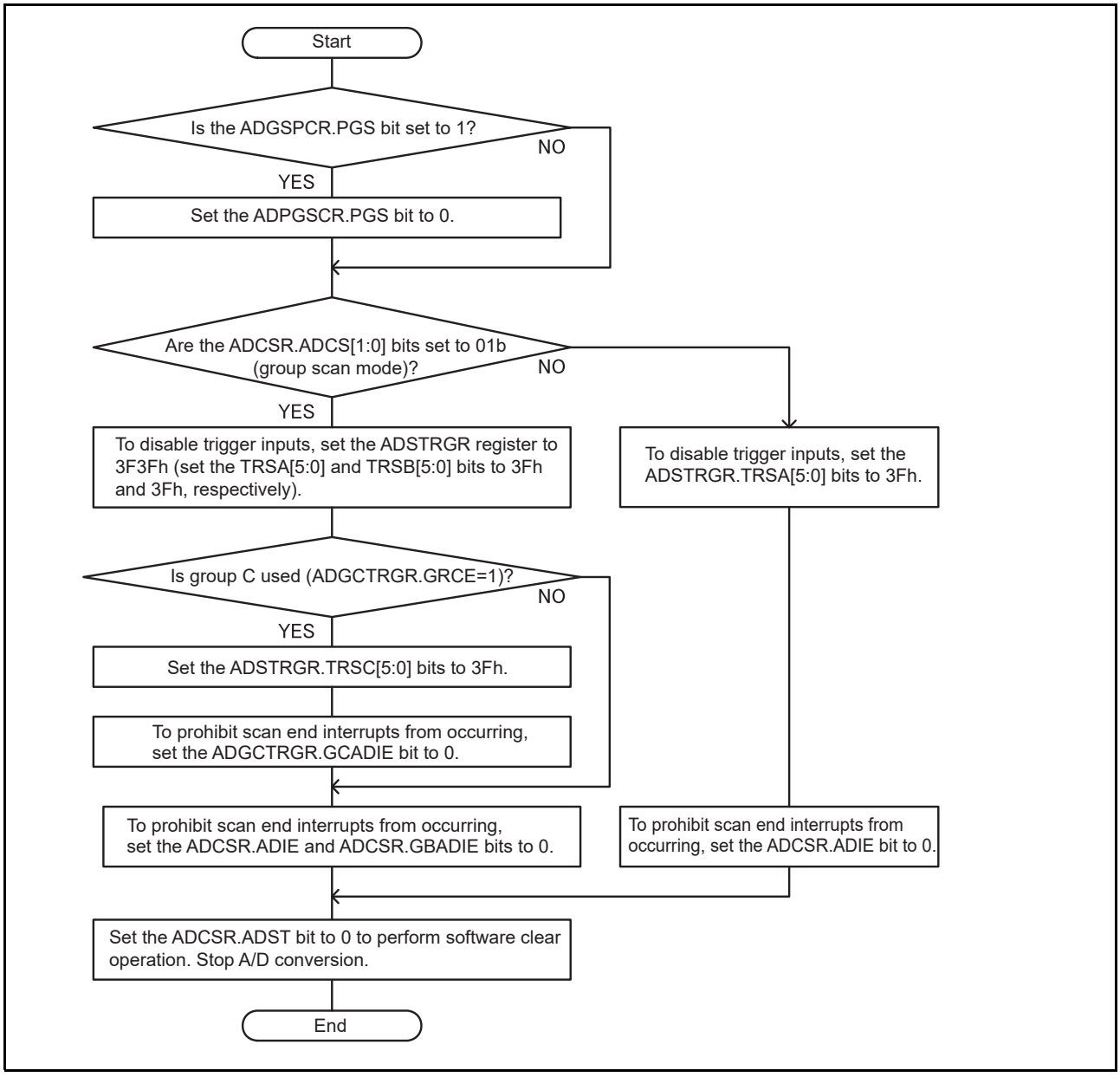


Figure 30.30 Procedures for Clear Operation by Software through the ADCSR.ADST Bit

Note: It takes two ADCLK cycles from software clear operation to the stop of scanning. To set the following operation after executing the software clear operation, a wait of at least two ADCLK cycles should be included.
- Scan end interrupt enable
- A/D conversion start by software
- Trigger input enable setting

30.5.2.2 Notes on Modes and Status Bits

Initialize or reset individually the voltage status in self-diagnosis, odd/even determination in double trigger mode, and the monitor bits in the compare function, if necessary.

- The reset value of voltage status in self-diagnosis can be selected in ADCER.DIAGVAL[1:0] after ADCER.DIAGLD is set to 1.
- Double trigger mode operates from the first scan after ADCSR.DBLE is set to 1 from 0.
- To initialize the monitor bits (MONCMPA, MONCMPB, MONCMPA) in the compare function, set ADCMPCR.CMPAE and ADCMPCR.CMPBE to 0.

30.5.3 A/D Conversion Restarting Timing and Termination Timing

It takes a maximum of six ADCLK cycles for the idle analog unit of the A/D converter to be restarted by setting the ADST bit in ADCSR register to 1. It takes a maximum of two ADCLK cycles for the operating analog unit of the 12-bit A/D converter to be terminated by setting the ADST bit in ADCSR register to 0.

30.5.4 Notes on Scan End Interrupt Handling

When scanning the same analog input twice using any trigger, the first A/D-converted data is overwritten with the second A/D-converted data in the case that the CPU does not complete reading out the A/D-converted data by the time the A/D conversion of the first analog input for the second scan ends after the first scan end interrupt is generated.

30.5.5 Module Standby Function Setting

Operation of the A/D converter can be disabled or enabled using the standby control register. The initial setting is for operation of the A/D converter to be halted. Releasing the module standby mode enables access to the registers. After canceling module standby mode, wait for at least 1 µs before starting A/D conversion.
For details, see section 52, Power-Down Modes.

30.5.6 Notes on Entering Low Power Consumption States

Before entering the Power-Down mode, make sure to stop A/D conversion. Here, set the ADST bit in ADCSR register to 0, and secure certain period of time until the analog unit of the A/D converter is stopped.
Follow the procedure given below to secure this time.
Follow the procedure for clear operation by software through the ADCSR.ADST bit, shown in Figure 30.30. After that, wait for two clock cycles of ADCLK before entering the Power-Down mode.

30.5.7 Notes on Canceling Software Standby Mode

After canceling software standby mode, wait for the oscillation stabilization time to elapse and then wait for at least 1 µs before starting A/D conversion. For details, see section 52, Power-Down Modes.

30.5.8 Error in Absolute Accuracy When Disconnection Detection Assistance is in Use

Using disconnection detection assistance leads to an error in absolute accuracy of the A/D converter. This is because an error voltage is input to the analog input pins due to the resistive voltage division between the pull-up or pull-down resistor (Rp) and the resistance of the signal source (Rs). This error in absolute accuracy is calculated from the following formula. Only use disconnection detection assistance after thorough evaluation.

Maximum error in absolute accuracy (LSB) = 4095 × Rs / Rp

30.5.9 Range of Voltage on the Analog Input Pins

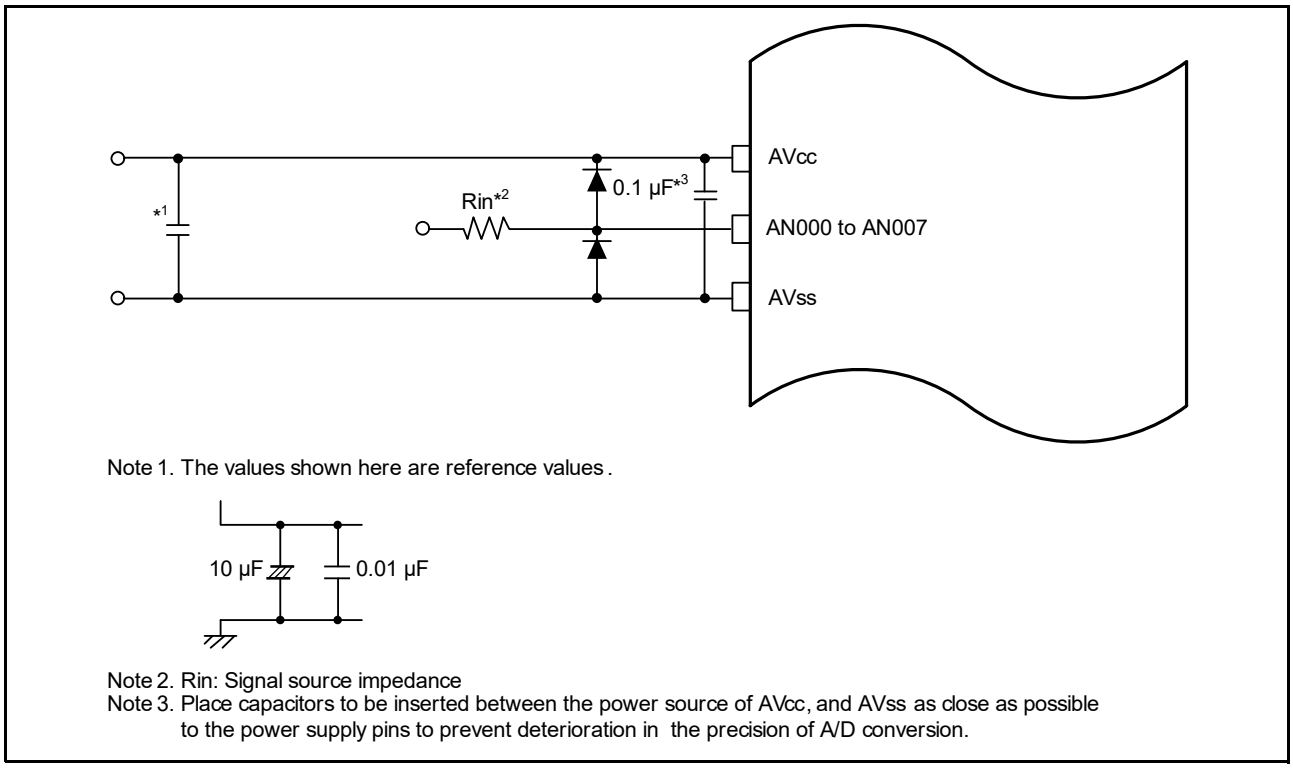
The voltage to be applied on the analog input pin should be in the range AVss ≤ AN00n (n = 0 to 7) ≤ AVcc.
Using this LSI chip with a voltage out of the allowed range may adversely affect the reliability of the LSI.

30.5.10 Notes on Board Design

When designing a board, as much as possible make sure that the signal wiring runs for the digital and analog circuits do not overlap or are not placed close to each other in the layout. Otherwise, electromagnetic induction and so on will occur, and this may adversely affect the precision of the values produced by A/D conversion.
The analog signals (AN000 to AN007) and analog power source (AVcc) are separated from the digital circuits by using the analog ground (AVss). Be sure to connect a 0.1µF capacitor between the analog power source and analog ground.
The analog ground (AVss) should be connected to a stable digital ground (Vss) on the board (single-point ground plane connection).

30.5.11 Notes on Noise Prevention

To avoid damage to the analog input pins (AN000 to AN007) due to abnormal voltages such as excessive surges, connect protection circuits to the analog input pins (AN000 to AN007) with reference to that shown in Figure 30.31.



Note 1. The values shown here are reference values.

Note 2. Rin: Signal source impedance
Note 3. Place capacitors to be inserted between the power source of AVcc, and AVss as close as possible to the power supply pins to prevent deterioration in the precision of A/D conversion.

Figure 30.31 Example of a Protection Circuit for the Analog Inputs

31. NAND Flash Controller

31.1 Overview

The NAND flash controller supports the functionality of the high-level interface described in the ONFI 1.x specifications.

31.1.1 Features

- Compatibility
  - Compatible with the ONFi 1.x specification. (8 bits)
    Note that this does not apply in cases where timing mode 3, 4, or 5 is in use.
  - Support for the Small Block devices
  Note: Only devices that allow to disable CE signal when device is in the busy state are supported.
  - Support for the low capacity devices (which use four address bytes) with the help of generic sequence.
- DMA Controller
  - Scatter-gather mode/registers managed mode
  - 32-bit system memory addressing.
  - Programmable FIFO threshold as a trigger for the DMA controller
- Interrupt Controller
  - Each interrupt can be masked.
  - Each interrupt has its own status flag.
  - The status flags are also valid when the given interrupt is masked, and can be checked by the software polling mechanism.
  - Single interrupt signal output
- Data buffering
  - FIFO module based on dual port memory.
- ECC unit
  - The ECC module is based on the BCH algorithms.
  - The syndrome calculation, error detection and error correction phases in the BCH-based modules are pipelined.
  - The ECC unit supports the following options: 256-, 512-, and 1024-byte data blocks. The data block size depends on the selected ECC option.
- Advanced features
  - The page cache read/write sequences are supported.
  - The multiple planes read/write sequences are supported.
  - The multi-LUN work mode is supported.
  - Command queuing mechanism.

Note: A queue is used when a host writes to a register. The queue is used when writing is to registers other than those listed below.
– INT_STATUS
– STATUS
– FIFO_INIT
– ECC_STAT
– FIFO_DATA
Writing to the registers listed above is executed immediately. Writing to the other registers, on the other hand, is executed through command queuing.

- The advanced Bad Blocks Management system
  - Record tables are stored in the system memory.
  - Implementation of hardware with the search algorithm

31.1.2 Block Diagram

Figure 31.1 is a block diagram of this module.

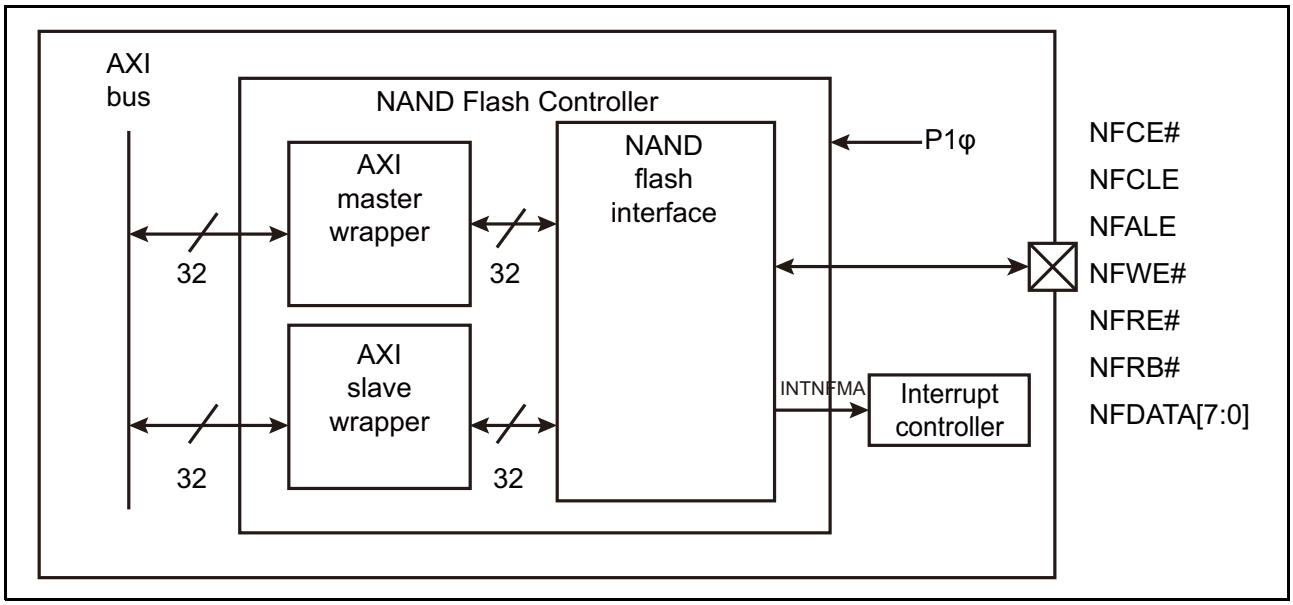


Figure 31.1 Block Diagram

The INTNFMA interrupt signal described in this section corresponds to the NAND interrupt sources of this controller.

31.1.3 External Pins

Table 31.1 lists the pin configuration.

Table 31.1 Pin Configuration

Name	I/O	Function
NFRE#	O	Read Enable
NFCE#	O	Chip Enable
NFCLE	O	Command Latch Enable
NFALE	O	Address Latch Enable
NFWE#	O	Write Enable
NFDATA[7:0]	I/O	Data
NFRB#	I	Ready/busy

31.1.4 Register Configuration

Table 31.2 shows the register configuration.

Table 31.2 Register Configuration

Name	Abbreviation	R/W	Address	Initial Value	Access Size
Controller commands register	COMMAND	R/W	H'E822C000	H'0000 0000	32
Main configurations register	CONTROL	R/W	H'E822C004	H'0000 0000	32
Controller status register	STATUS	R	H'E822C008	H'0000 0001	32
Mask register for the READ STATUS commands	STATUS_MASK	R/W	H'E822C00C	H'0000 4040	32
Interrupts mask register	INT_MASK	R/W	H'E822C010	H'0000 0000	32
Interrupts status register	INT_STATUS	R/W	H'E822C014	H'0000 0000	32
ECC module control register	ECC_CTRL	R/W	H'E822C018	H'0000 0000	32
ECC offset in the spare area register	ECC_OFFSET	R/W	H'E822C01C	H'0000 0000	32
ECC module status register	ECC_STAT	R/W	H'E822C020	H'0000 0000	32
Column address register 0	ADDR0_COL	R/W	H'E822C024	H'0000 0000	32
Row address register 0	ADDR0_ROW	R/W	H'E822C028	H'0000 0000	32
Column address register 1	ADDR1_COL	R/W	H'E822C02C	H'0000 0000	32
Row address register 1	ADDR1_ROW	R/W	H'E822C030	H'0000 0000	32
FIFO module interface register	FIFO_DATA	R/W	H'E822C038	—	32
Data register	DATA_REG	R/W	H'E822C03C	H'0000 0000	32
Data register size selection register	DATA_REG SIZE	R/W	H'E822C040	H'0000 0000	32
Records table pointer register	DEV0_PTR	R/W	H'E822C044	H'0000 0000	32
DMA base address register	DMA_ADDR_L	R/W	H'E822C064	H'0000 0000	32
DMA counter initial value register	DMA_CNT	R/W	H'E822C06C	H'0000 0000	32
DMA control register	DMA_CTRL	R/W	H'E822C070	H'0000 0001	32
Bad block management (BBM) control register	BBM_CTRL	R/W	H'E822C074	H'0000 0000	32
Page size value register	DATA_SIZE	R/W	H'E822C084	H'0000 0000	32
Timing configuration register	TIMINGS_ASYN	R/W	H'E822C088	H'0000 0000	32
Command sequence timing configuration register 0	TIME_SEQ_0	R/W	H'E822C090	H'0000 0000	32
Command sequence timing configuration register 1	TIME_SEQ_1	R/W	H'E822C094	H'0000 0000	32
Generic sequence timing configuration register 0	TIME_GEN_SEQ_0	R/W	H'E822C098	H'0000 0000	32
Generic sequence timing configuration register 1	TIME_GEN_SEQ_1	R/W	H'E822C09C	H'0000 0000	32
Generic sequence timing configuration register 2	TIME_GEN_SEQ_2	R/W	H'E822C0A0	H'0000 0000	32
FIFO control register	FIFO_INIT	R	H'E822C0B0	H'0000 0000	32
FIFO status register	FIFO_STATE	R	H'E822C0B4	H'0000 0095	32
GENERIC_SEQ register	GEN_SEQ_CTRL	R/W	H'E822C0B8	H'0000 0000	32
MLUN register	MLUN	R/W	H'E822C0BC	H'0000 0000	32
Records table size register	DEV0_SIZE	R/W	H'E822C0C0	H'0000 0000	32
DMA trigger level value register	DMA_TRIG_TLVL	R/W	H'E822C114	H'0000 0000	32
CMD ID initial value register	CMD_MARK	W	H'E822C124	H'0000 0000	32
LUN status register	LUN_STATUS_0	R	H'E822C128	H'0000 00FF	32
Generic sequence timing configuration register 3	TIME_GEN_SEQ_3	R/W	H'E822C134	H'0000 0000	32
ECC error level counter register	ECC_CNT	R/W	H'E822C14C	H'0000 0000	32

31.2 Register Description

31.2.1 Controller commands register (COMMAND)

COMMAND specifies the command sequence code and selects data.
The write of the command sequence code to this register triggers the programmed command sequence execution. If the execution cannot be started immediately, the transfer to this register is prolonged by the series of the WAIT responses. Each command sequence can generate an interrupt when it is completed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMD_2[7:0]								CMD_1[7:0]/CMD_3[7:0]*1							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMD_0[7:0]								DATA_SEL	INPUT_SEL	CMD_SEQ[5:0] OPC[1:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CMD_2[7:0]	H'00	R/W	Code of the third command in a sequence
23 to 16	CMD_1[7:0]/ CMD_3[7:0]*1	H'00	R/W	Code of the second command in a sequence
15 to 8	CMD_0[7:0]	H'00	R/W	Code of the first command in a sequence
7	DATA_SEL	0	R/W	Data/FIFO selection 0: FIFO selected 1: DATA register selected
6	INPUT_SEL	0	R/W	Input module selection 0: select the registers in the Slave I/F Unit (SIU) module as input 1: select the DMA module as input
5 to 0	CMD_SEQ[5:0]	H'00	R/W	Command code

Note 1. Depending on the selected command sequence, this field will store CMD1 or CMD3 code.
In the generic sequence, both commands are used in a single sequence.

31.2.2 Main configurations register (CONTROL)

CONTROL stores the common configuration parameters.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	AUTO_READ_STAT_EN	MLUN_EN	SMALL_BLOCK_EN	—	—	—	ADDR1_AUTO_INCR	ADDR0_AUTO_INCR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	RW	R	R	R	R/W	R/W

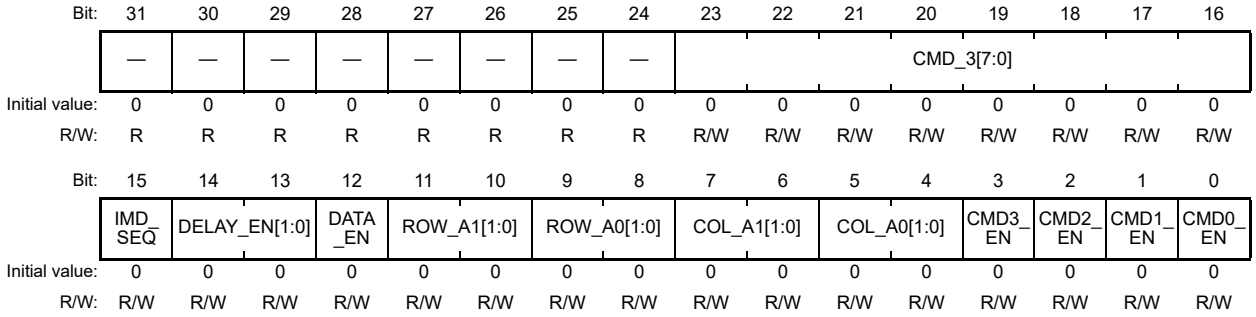
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PROT_EN	BBM_EN	—	—	—	—	—	BLOCK_SIZE[1:0]		ECC_EN	INT_EN	—	ECC_BLOCK_SIZE[1:0]		READ_STATUS_EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
23	AUTO_READ_STAT_EN	0	R/W	Auto Read Status mode enable If active, the controller reads the status after the PROGRAM PAGE and ERASE BLOCK commands. It can trigger an interrupt. The ERROR_MASK field in the STATUS_MASK register must be configured when this feature is enabled. After executing a PROGRAM PAGE or ERASE BLOCK command, you can use the field to check the state following the operation (in terms of success or failure). 0: Auto Read Status mode disabled 1: Auto Read Status mode enabled
22	MLUN_EN	0	R/W	Multi LUN mode enables If active, it enables controller multi LUN work mode 0: Multi LUN mode disabled 1: Multi LUN mode enabled For more details, see section 31.3.4, Multi LUN Work Mode.
21	SMALL_BLOCK_EN	0	R/W	Enable small block mode In this mode, the controller sends only a single byte as the column address instead of two bytes, as done in the big block NAND Flash devices. 0: big block mode enabled 1: small block mode enabled Note: Only devices that allow to disable CE signal when device is in the busy state are supported.
20 to 18	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
17	ADDR1_AUTO_INCR	0	R/W	Address auto increment for row address register 1 (ADDR1_ROW) When this bit is set, sending any command sequence using address register 1 causes the increment of address register 1. 0: auto increment disabled 1: auto increment enabled
16	ADDR0_AUTO_INCR	0	R/W	Address auto increment for row address register 0 (ADDR0_ROW) When this bit is set, sending any command sequence using address register 0 causes the increment of address register 0 0: auto increment disabled 1: auto increment enabled
15	—	0	R	Reserved bit This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
14	PROT_EN	0	R/W	Protect mechanism enable 0: protect disabled 1: protect enabled

Bit	Bit Name	Initial Value	R/W	Description
13	BBM_EN	0	R/W	Bad Block Management enable flag For more details, see section 31.3.5, Remapping Mechanism.
12 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7, 6	BLOCK_ SIZE[1:0]	H'0	R/W	The Block Size 00: 32 pages per block 01: 64 pages per block 10: 128 pages per block 11: 256 pages per block
5	ECC_EN	0	R/W	Hardware ECC support enable 0: ECC disabled 1: ECC enabled Hardware ECC can be used only when m* (ECC_BLOCK_SIZE) ≤ DATA_SIZE ≤ m* (ECC_BLOCK_SIZE + 32), where m is 1,2,3...
4	INT_EN	0	R/W	Global Interrupt enable 0: Interrupts disabled 1: Interrupts enabled For more details, see section 31.3.6, Interrupts Mechanism.
3	—	0	R	Reserved bit This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
2, 1	ECC_BLOCK_ SIZE[1:0]	H'0	R/W	The ECC Block Size: 00: 256 bytes 01: 512 bytes 10: 1024 bytes 11: not available The ECC block size can be changed only when all memory devices are ready.
0	READ_STATUS_ EN	0	R/W	Automatically READ STATUS/check RnB lines The STATUS_MASK field in the STATUS_MASK register must be configured when this feature is enabled. This bit is used to select how the controller detects the ready or busy state of the device. 0: The controller checks RnB lines. 1: The controller sends READ STATUS commands. Note: Automatically sent READ STATUS command is only available in devices compatible with ONFI 1.0.

31.2.3 GENERIC_SEQ register (GEN_SEQ_CTRL)

GENERIC_SEQ is used to parameterize the generic sequences. For more details, see section 31.3.2, Generic Sequence. For examples, see Table 31.7.



Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CMD_3[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IMD_SEQ	DELAY_EN[1:0]		DATA_EN	ROW_A1[1:0]		ROW_A0[1:0]		COL_A1[1:0]		COL_A0[1:0]		CMD3_EN	CMD2_EN	CMD1_EN	CMD0_EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
23 to 16	CMD_3[7:0]	H'00	R/W	Command 3 code value
15	IMD_SEQ	0	R/W	Enable immediate command execution This bit allows the command sequence to be executed without checking the selected target state. 0: feature disabled 1: feature enable
14, 13	DELAY_EN[1:0]	H'0	R/W	Enable the busy 0 or 1 phase These bits allow enabling or disabling the presence of the "busy" phase in the generic sequence. 00: disable both delays 01: enable delay 0 10: enable delay 1 11: disable both delays
12	DATA_EN	0	R/W	Enable data part sequence This bit allows enabling or disabling the data phase of the generic sequence. 0: disable data phase 1: enable data phase
11, 10	ROW_A1[1:0]	H'0	R/W	Row Address Cycles The number of bytes of row addresses to be sent to the NAND Flash device 00: 0 address cycles 01: 1 address cycle 10: 2 address cycles 11: 3 address cycles
9, 8	ROW_A0[1:0]	H'0	R/W	Row Address Cycles The number of bytes of row addresses to be sent to the NAND Flash device 00: 0 address cycles 01: 1 address cycle 10: 2 address cycles 11: 3 address cycles
7, 6	COL_A1[1:0]	H'0	R/W	Column Address Cycles The number of bytes of column addresses to be sent to the NAND Flash device 00: 0 address cycles 01: 1 address cycle 10: 2 address cycles 11: setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
5, 4	COL_A0[1:0]	H'0	R/W	Column Address Cycles The number of bytes of column addresses to be sent to the NAND Flash device 00: 0 address cycles 01: 1 address cycle 10: 2 address cycles 11: setting prohibited
3	CMD3_EN	0	R/W	Enable Command 3 Phase This bit allows enabling or disabling the presence of the “command 3” phase in the generic sequence. 1: enable 0: disable
2	CMD2_EN	0	R/W	Enable Command 2 phase This bit allows enabling or disabling the presence of the “command 2” phase in the generic sequence. 1: enable 0: disable
1	CMD1_EN	0	R/W	Enable Command 1 phase This bit allows enabling or disabling the presence of the “command 1” phase in the generic sequence. 1: enable 0: disable
0	CMD0_EN	0	R/W	Enable Command 0 phase This bit allows enabling or disabling the presence of the “command 0” phase in the generic sequence. 1: enable 0: disable

31.2.4 Controller status register (STATUS)

STATUS stores the NAND Flash controller and connected device status flags. These flags can be used to obtain the current controller internal state.

The CTRL_STAT flag is set after the controller starts to execute the requested command for the selected NAND Flash device, and it is active while the command execution is not completed. Command execution can be divided into two phases. In the first phase, the command sequence is executed at the moment when the NAND Flash device goes into the busy state. After that, the controller stores information about the pending operation on the selected device. In the second phase, the controller automatically finishes the pending command execution based on the previously stored data. As long as this flag is set, the controller does not accept any new commands.

The MEM0_ST flag indicates the state of the NAND flash device. The flag has the same function as the NAND Flash device RnB line.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CMD_ID[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DATA_REG_ST	DATASIZE_ERROR_ST	CTRL_STAT	—	—	—	—	—	—	—	MEM0_ST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
23 to 16	CMD_ID[7:0]	H'00	R	Command ID The current command under execution identification marker added before command was put in to the command FIFO.
15 to 11	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
10	DATA_REG_ST	0	R	The DATA_REG: Resetting of this flag is possible only by reading the data from the DATA_REG register. 1: data in DATA_REG is available 0: data in DATA_REG is not available
9	DATASIZE_ERROR_ST	0	R	The Data Size value error This bit indicates that the value written to the DATA_SIZE register is not correct when the ECC is enabled. For details on the values of the correct size of data, see the description of the DATA_SIZE register. 0: correct value 1: incorrect value
8	CTRL_STAT	0	R	The main controller status bit 0: controller ready 1: controller busy
7 to 1	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	MEM0_ST	1	R	Device 0 status flag 1: device ready 0: device busy

Note: When using the READ PARAMETER PAGE command, the block size for the ECC memory (ECC_BLOCK_SIZE) must be the same as that set in the DATA_SIZE register.
When using the SET FEATURES, GET FEATURES, or READ ID command, disable the ECC function.

31.2.5 LUN status register (LUN_STATUS_0)

LUN_STATUS_0 allows to access the LUN status information. Each bit of the LUN status field contain the status of single LUN of device. The busy state is indicated as "0", and the ready state is indicated as "1".

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	MEM0_LUN[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7 to 0	MEM0_LUN[7:0]	H'FF	R	Memory 0 LUN-s status field.

31.2.6 Interrupts mask register (INT_MASK)

INT_MASK allows masking of the selected interrupts source in the NAND Flash controller. The masked interrupts still set the corresponding bits in the status register, but do not assert the interrupt signal.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	ECC_ INT0_EN	—	—	—	—	—	—	—	STAT_ERR_ INT0_EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MEM0RDY_ INT_EN	—	PG_SZ_ERR_ INT_EN	—	TRANS_ ERR_EN	DMA_INT_ EN	DATA_REG_ INT_EN	CMD_END_ INT_EN	PROT_ INT_EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R/W	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
24	ECC_ INT0_EN	0	R/W	Enables the interrupt from the ECC module status for Memory device 0. 0: interrupt disabled 1: interrupt enabled
23 to 17	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	STAT_ERR_ INT0_EN	0	R/W	Enables the interrupt when the most recently finished operation on the Memory device 0 failed. This applies to PROGRAM PAGE and ERASE BLOCK operations. It is not valid following a READ-series operation. 0: interrupt disabled 1: interrupt enabled
15 to 9	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8	MEM0_RDY_ INT_EN	0	R/W	The memory device 0 is ready for the new command. 0: interrupt disabled 1: interrupt enabled For more details see Figure 31.31.
7	—	0	R	Reserved bit This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
6	PG_SZ_ERR_ INT_EN	0	R/W	Data Size error occur. 0: interrupt disabled 1: interrupt enabled
5	—	0	R	Reserved bit This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
4	TRANS_ERR_ EN	0	R/W	The transfer on the slave interface error 0: interrupt disabled 1: interrupt enabled
3	DMA_INT_EN	0	R/W	DMA transfer ended. 0: interrupt disabled 1: interrupt enabled
2	DATA_REG_ INT_EN	0	R/W	Data in DATA_REG is available. 0: interrupt disabled 1: interrupt enabled

Bit	Bit Name	Initial Value	R/W	Description
1	CMD_END_ INT_EN	0	R/W	Command sequence ended. 0: interrupt disabled 1: interrupt enabled For more details see Figure 31.31, Command Sequence End and Memory-Ready Interrupts.
0	PROT_ INT_EN	0	R/W	Erase/Write protected area interrupt enable 0: interrupt disabled 1: interrupt enabled

31.2.7 Interrupts status register (INT_STATUS)

INT_STATUS stores the NAND Flash controller interrupt flags. If the given bit is 0, the corresponding interrupt condition is not met. If the given bit is 1, the corresponding interrupt condition is met.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	ECC_INT0_FL	—	—	—	—	—	—	—	STAT_ERR_INT0_FL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MEM0_RDY_INT_FL	—	PG_SZ_ERR_INT_FL	—	TRANS_ERR_FL	DMA_INT_FL	DATA_REG_INT_FL	CMD_END_INT_FL	PROT_INT_FL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R/W	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
24	ECC_INT0_FL	0	R/W	Selected flag (source is ECC_UNC_0, ECC_ERROR_0 or ECC_OVER_0) in the ECC module is set.
23 to 17	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	STAT_ERR_INT0_FL	0	R/W	Most recently finished operation on the Memory device 0 failed. This applies to PROGRAM PAGE and ERASE BLOCK operations. It is not valid following a READ-series operation.
15 to 9	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8	MEM0_RDY_INT_FL	0	R/W	The memory device 0 is ready for the new command.
7	—	0	R	Reserved bit This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
6	PG_SZ_ERR_INT_FL	0	R/W	Data Size error flag When the ECC is enabled, the value written into the DATA_SIZE register has some restrictions. Interrupt condition is met when the value written to the DATA_SIZE register is not correct.
5	—	0	R	Reserved bit This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
4	TRANS_ERR_FL	0	R/W	The transfer on the slave interface error The flag is set when the access to the FIFO memory has the opposite direction to the current FIFO configuration.
3	DMA_INT_FL	0	R/W	DMA transfer ended flag.
2	DATA_REG_INT_FL	0	R/W	Data in DATA_REG is available.
1	CMD_END_INT_FL	0	R/W	Transfer sequence ended.
0	PROT_INT_FL	0	R/W	Erase/Write protected area interrupt enable

31.2.8 ECC module control register (ECC_CTRL)

ECC_CTRL stores all configuration parameters required by the ECC.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ECC_SEL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	ERR_THRESHOLD[5:0]						—	—	—	—	—	ECC_CAP[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
17,16	ECC_SEL[1:0]	H'0	R/W	The ECC interrupt source select These bits select the ECC module flag that will be used as a source for the interrupt signal: 00: select ECC_ERROR (correctable error) flag as interrupt source. 01: select ECC_UNC (uncorrectable error) flag as interrupt source. 1x: select ECC_OVER (acceptable errors level overflow) flag as interrupt source. The ECC_OVER flag is not set in response to an uncorrectable error (ECC_UNC_0 = 1).
15, 14	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
13 to 8	ERR_ THRESHOLD [5:0]	H'00	R/W	The acceptable errors level The value of this field contains the number of errors that is acceptable. This field must be initialized by the application program.
7 to 3	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2 to 0	ECC_CAP[2:0]	H'0	R/W	The ECC module correction ability The correction ability can be changed only when memory devices are ready. 000: 2 001: 4 010: 8 011: 16 100: 24 All others: 32

31.2.9 ECC module status register (ECC_STAT)

ECC_STAT stores all ECC module status information.

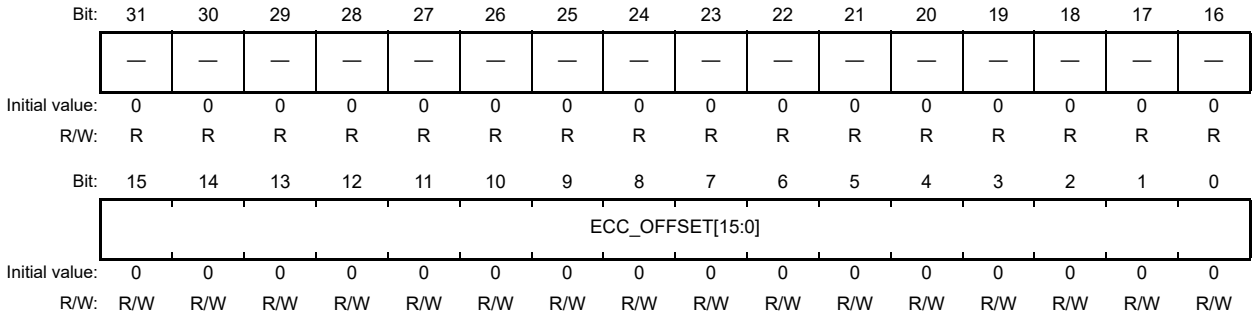
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ECC_OVER_0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECC_UNC_0	—	—	—	—	—	—	—	ECC_ERROR_0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	ECC_OVER_0	0	R/W	The Memory device 0 acceptable errors level overflow The bit is set when the number of errors is bigger than the value of declared ERR_THRESHOLD bits. Uncorrectable errors are not counted.
15 to 9	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8	ECC_UNC_0	0	R/W	The Memory device 0 uncorrectable error flag The bit is set when the uncorrectable error occur during the read operation.
7 to 1	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	ECC_ERROR_0	0	R/W	The Memory device 0 correctable error flag The bit is set when a correctable or uncorrectable error (ECC_UNC_0 = 1) occurs during reading.

31.2.10 ECC offset in the spare area register (ECC_OFFSET)

ECC_OFFSET stores the offset value from the beginning of the page to the place where correction words will be stored. The value of the ECC_OFFSET register must be bigger than the value in the DATA_SIZE register.
In small block mode, the value in ECC_OFFSET is ignored and the correction words are located in the NAND Flash memory device just behind the data.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	ECC_ OFFSET[15:0]	H'0000	R/W	Correction words block offset

31.2.11 ECC error level counter register (ECC_CNT)

ECC_CNT stores number of value detected during last page read operation. This register content is not automatically cleared, it must be done by software. The new page read operation does not overwrite previous register value. Register contain value of the largest error level detected in processed ECC blocks.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	ERR_LVL[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5 to 0	ERR_LVL[5:0]	H'00	R/W	Detected error level Uncorrectable errors are not counted.

31.2.12 Column/row address registers (ADDR[1:0]_COL, ADDR[1:0]_ROW)

ADDR[1:0]_COL and ADDR[1:0]_ROW store the address that will be used by the next command sequence during access to the NAND Flash device.

ADDR [1:0] _COL:

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADDR[1:0]_COL[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	ADDR[1:0]_ COL[15:0]	H'0000	R/W	Column address, A15-A0 address bits

ADDR [1:0] _ROW:

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADDR[1:0]_ROW[23:16]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADDR[1:0]_ROW[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
23 to 0	ADDR[1:0]_ ROW[23:0]	H'00 0000	R/W	Row address, A39-A16 address bits (Page address, Block address and LUN address in the ONFI case)

Note 1. There is no register that defines the total memory size of the NAND Flash memory chip, thus the controller is not able to determine which address bits in ADDR [1:0]_COL and ADDR[1:0]_ROW are important and which must be zero. For this reason, the software must take special care with the values written to these registers. Incorrect values of unused address bits (none '0' values) can cause errors in memory access.
A relation between address registers and memory device address width is configured by the command sequence field of the COMMAND register. This field determines which command sequence has to be used and how many address bytes are used when addressing a NAND Flash memory device.

(Example: In order to erase blocks, the three address cycles containing the row address are written into the NAND Flash memory device. The NAND Flash Controller automatically writes bits A39-A16 to the NAND Flash device).
Refer to section 31.3.1 (2), Command Sequence Encoding and Table 31.6, Command Sequence Encoding in order to see how many address cycles are written into the NAND Flash memory device by each Command Sequence.
The address written to the address register must be aligned according to the NAND Flash device. Unused bits must be padded with zeros.

Note 2. When the auto increment for the row address register is enabled, the proper value of this register can be read only when the bit CTRL_STAT in STATUS register is clear.

Table 31.3 Address Registers and Address Bytes Relationship

Address Cycle	I/O 0	I/O 1	I/O 2	I/O 3	I/O 4	I/O 5	I/O 6	I/O 7
1st cycle	A0	A1	A2	A3	A4	A5	A6	A7
2nd cycle	A8	A9	A10	A11	A12	A13	A14	A15
3rd cycle	A16	A17	A18	A19	A20	A21	A22	A23
4th cycle	A24	A25	A26	A27	A28	A29	A30	A31
5th cycle	A32	A33	A34	A35	A36	A37	A38	A39

31.2.13 Page size value register (DATA_SIZE)

DATA_SIZE stores the value of the data block size. The data size value is remembered as the number of bytes per transferred block, but its size must be declared as the multiple of the chosen NAND Flash word size. The unused bits for the given word size configuration are ignored and replaced with zeros.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DATA_SIZE[14:0]														
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
14 to 0	DATA_ SIZE[14:0]	H'0000	R/W	Data size The value of this field defines data size.

Note: Write the proper value to the DATA_SIZE register when ECC is enabled:
(ECC_BLOCK_SIZE) × m ≤ DATA_SIZE ≤ (ECC_BLOCK_SIZE + 32) × m, Where m is 1, 2, 3 …

Here is an example of correct DATA_SIZE values when ECC_BLOCK_SIZE equals 512 bytes.

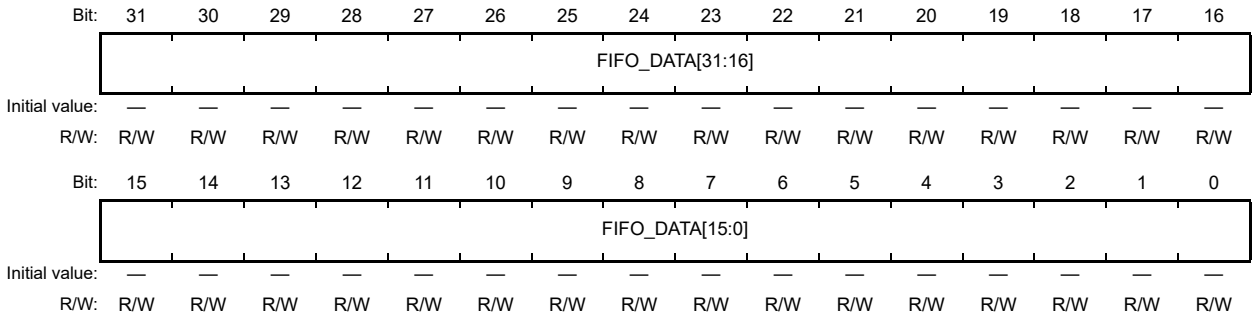
Table 31.4 Example of Correct DATA_SIZE Values

DATA_SIZE Value	Setting
H'0000	Not allowed
….	….
H'001F	Not allowed
H'0020	Not allowed
H'0021	Not allowed
….	….
H'01FF	Not allowed
H'0200	Allowed
H'0201	Allowed
….	….
H'0220	Allowed
H'0221	Not allowed
….	….
H'03FF	Not allowed
H'0400	Allowed
H'0401	Allowed
….	….
H'0440	Allowed
H'0441	Not allowed
….	….

Note 1. ECC_BLOCK_SIZE equals 512 bytes.

31.2.14 FIFO module interface register (FIFO_DATA)

FIFO_DATA is used as an entry point to the FIFO module. The CPU can access the FIFO module by reading from or writing to FIFO_DATA in the same way as it accesses any other register.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	FIFO_DATA [31:0]	Undefined	R/W	FIFO Data

Note: FIFO_DATA can be accessed after the read or write command is issued.

31.2.15 Bad block management (BBM) control register (BBM_CTRL)

BBM_CTRL stores the BBM specific control parameters.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RMP_INIT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0.
0	RMP_INIT	0	R/W	Remap initial flag If set, this flag forces the BBM module to reread the remapping table after it was updated by software. This flag is set by software and cleared by hardware after the remapping table is re-read.

31.2.16 Records table pointer register (DEV0_PTR)

The bad block management mechanism uses the tables in the system memory to store the remapping records. DEV0_PTR stores the table of the remapping record address.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PTR_ADDR[11:2]										—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
11 to 2	PTR_ ADDR[11:2]	H'000	R/W	Remap table pointer The field contains an address of the remap table in the internal memory.
1 to 0	—	ALL 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.

31.2.17 Records table size register (DEV0_SIZE)

The bad block management mechanism implemented in the controller uses the tables in the system memory to store the remapping records. Each table can store a variable number of records depending on the number of bad blocks in the NAND Flash device.

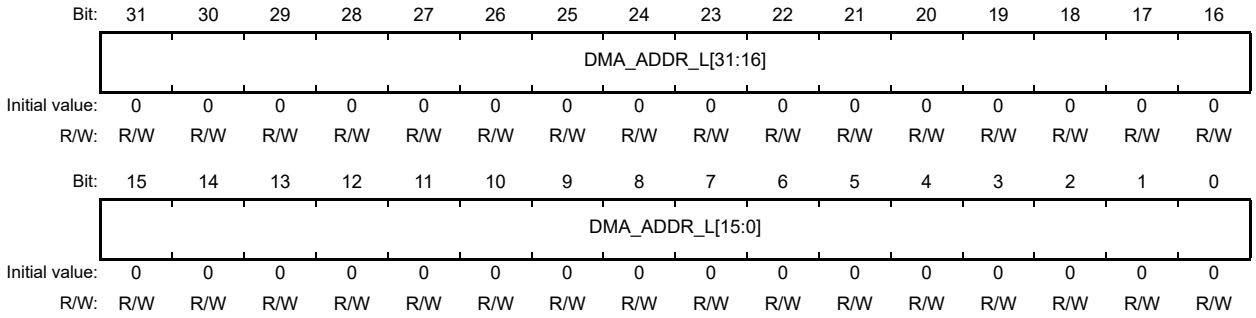
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DEV_SIZE[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
11 to 0	DEV_SIZE[11:0]	H'000	R/W	Number of records

31.2.18 DMA base address register (DMA_ADDR_L)

DMA_ADDR_L holds the DMA base address. The register contains the address of the first data in the data block in the system memory, or the address of the first descriptor. The DMA module can read data from the memory location set by DMA_ADDR and write it to the FIFO module, or read data from the FIFO module and write it to the memory, starting from the location indicated by the DMA_ADDR.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_ADDR_L [31:0]	H'0000 0000	R/W	DMA base address The two least significant bits are ignored, thus the address must be aligned to 32-bit words.

31.2.19 DMA counter initial value register (DMA_CNT)

DMA_CNT defines the number of bytes that will be transferred by the DMA module. The register remains unchanged during transfer.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNT_INIT[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNT_INIT[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CNT_INIT [31:0]	H'0000 0000	R/W	Bytes counter initial value The field contains data page length in bytes (H'0000 0004 to H'FFFF FFFC). Specify a multiple of four as the number of bytes to be transferred.

31.2.20 DMA control register (DMA_CTRL)

DMA_CTRL is a control register for the DMA channel. This register defines the parameters of the DMA transfer.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DMA_ START	—	DMA_ MODE	DMA_ BURST[2:0]			—	DMA_ READY
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7	DMA_START	0	R/W	Set bit DMA_START to start DMA when the command sequence is sent to the NAND Flash memory. For more details see section 31.4.2 (2), DMA Description.
6	—	0	R	Reserved bit This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
5	DMA_MODE	0	R/W	DMA work mode 0: the registers managed mode 1: the Scatter-Gather mode
4 to 2	DMA_BURST [2:0]	H'0	R/W	Burst Type Specify the DMA transfer type. 000: incrementing burst transfer (address increment) Others: Setting prohibited.
1	—	0	R	Reserved bit This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
0	DMA_READY	1	R	DMA ready flag The flag is set when transfer is completed.

31.2.21 DMA trigger level value register (DMA_TRIG_TLVL)

DMA_TRIG_TLVL specifies the data FIFO occupancy level that will trigger the DMA module. For more details see section 31.4.2 (2), DMA Description.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DMA_TRIG_TLVL[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7 to 0	DMA_TRIG_TLVL[7:0]	H'00	R/W	DMA trigger level The trigger level is counted using the 32-bit words as entity.

31.2.22 Mask register for the READ STATUS commands (STATUS_MASK)

The STATE_MASK bits are used to mark the ready/busy bits of the status byte in the NAND Flash device. These bits are used during the internal read status operation. In the ONFI specification, the user must mask all fields except RDY or ARDY (depending on the application).
The ERROR_MASK bits are used to mask unused fields when the controller automatically reads the status of the NAND Flash memory device. In the ONFI specification, the user must mask all fields except FAIL or FAILC (depending on the application).

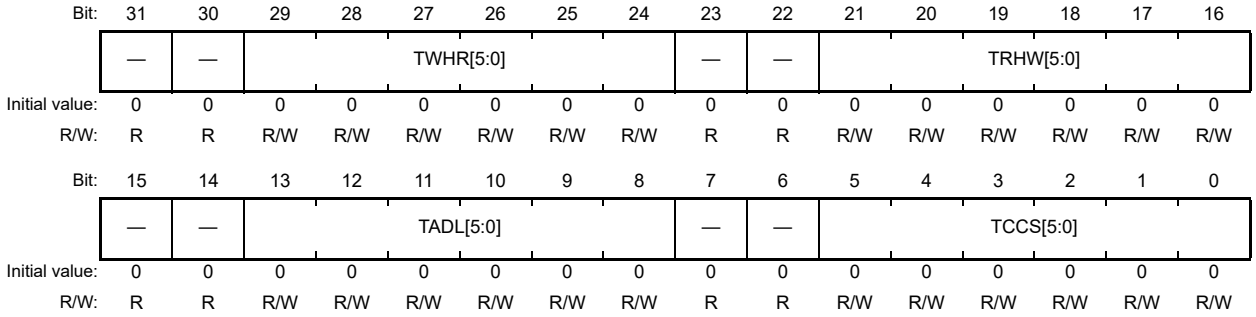
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERROR_MASK[7:0]								STATE_MASK[7:0]							
Initial value:	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 8	ER-ROR_MASK[7:0]	H'40	R/W	Error State Mask Used to mask the error bits if automatic read status feature is enabled.
7 to 0	STATE_ MASK[7:0]	H'40	R/W	State Mask Used to mask status bits when the read status command is used to obtain the NAND flash status.

31.2.23 Command sequence timing configuration register 0 (TIME_SEQ_0)

TIME_SEQ_0 controls the waveform timing parameters. All timings are generated from the P1φ clock signal, and delays are added to the timings in clock cycle units.



Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
29 to 24	TWHR[5:0]	H'00	R/W	NFWE# high to NFRE# low time
23, 22	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
21 to 16	TRHW[5:0]	H'00	R/W	NFRE# high to NFWE# low time
15, 14	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
13 to 8	TADL[5:0]	H'00	R/W	ALE to data start time
7, 6	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5 to 0	TCCS[5:0]	H'00	R/W	Change column setup

31.2.24 Command sequence timing configuration register 1 (TIME_SEQ_1)

TIME_SEQ_1 controls the waveform timing parameters. All timings are generated from the P1φ clock signal, and delays are added to the timings in clock cycle units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TRR[5:0]						—	—	TWB[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
13 to 8	TRR[5:0]	H'00	R/W	Read High to Read Low Time TRR is the time between a rising edge on the read/busy input line and the assertion of the read enable signal.
7, 6	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5 to 0	TWB[5:0]	H'00	R/W	tWB delay The time measured from a rising edge of the NFWE# signal to a falling edge on the RnB line.

31.2.25 Generic sequence timing configuration register 0 (TIME_GEN_SEQ_0)

TIME_GEN_SEQ_0 controls the waveform timing parameters for the generic sequence. For more detail see section 31.3.2, Generic Sequence. All timings are generated from the P1φ clock signal, and delays are added to the timings in clock cycle units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	t0_d3[5:0]						—	—	t0_d2[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	t0_d1[5:0]						—	—	t0_d0[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
29 to 24	t0_d3[5:0]	H'00	R/W	Command to Data Time The time between the sending of a command and the data transfer.
23, 22	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
21 to 16	t0_d2[5:0]	H'00	R/W	Command to Delay Time The wait time between the sending of a command to the NAND flash memory device and the memory becoming ready.
15, 14	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
13 to 8	t0_d1[5:0]	H'00	R/W	Command to Command time The time between two subsequent commands sent to the NAND Flash memory device.
7, 6	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5 to 0	t0_d0[5:0]	H'00	R/W	Command to Address time The time between sending a command and sending the address to the NAND Flash memory device.

31.2.26 Generic sequence timing configuration register 1 (TIME_GEN_SEQ_1)

TIME_GEN_SEQ_1 controls the waveform timing parameters for the generic sequence. For more detail see section 31.3.2, Generic Sequence. All timings are generated from the P1φ clock signal, and delays are added to the timings in clock cycle units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	t0_d7[5:0]						—	—	t0_d6[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	t0_d5[5:0]						—	—	t0_d4[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
29 to 24	t0_d7[5:0]	H'00	R/W	Address to Data time The time between sending the address and data transferring.
23, 22	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
21 to 16	t0_d6[5:0]	H'00	R/W	Address to Delay Time The wait time after the sending of an address to the NAND flash memory device and the memory becoming ready.
15, 14	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
13 to 8	t0_d5[5:0]	H'00	R/W	Address to Address Time The time between the sending of an address to the NAND flash memory device and that of the next address.
7, 6	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5 to 0	t0_d4[5:0]	H'00	R/W	Address to Command time The time between sending the address and sending a command to the NAND Flash memory device.

31.2.27 Generic sequence timing configuration register 2 (TIME_GEN_SEQ_2)

TIME_GEN_SEQ_2 controls the waveform timing parameters for the generic sequence. For more detail see section 31.3.2, Generic Sequence. All timings are generated from the P1φ clock signal, and delays are added to the timings in clock cycle units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	t0_d11[5:0]						—	—	t0_d10[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	t0_d9[5:0]						—	—	t0_d8[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
29 to 24	t0_d11[5:0]	H'00	R/W	Data to Delay Time The wait time between the data transfer to the NAND flash memory device and the memory becoming ready.
23, 22	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
21 to 16	t0_d10[5:0]	H'00	R/W	Data to Command time The time between data transferring and sending a command to the NAND Flash memory device.
15, 14	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
13 to 8	t0_d9[5:0]	H'00	R/W	Delay to Command Time The time between the memory becoming ready and the sending of a command to the NAND flash memory device.
7, 6	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5 to 0	t0_d8[5:0]	H'00	R/W	Delay to Data time The time between waiting until the memory is ready and data transferring.

31.2.28 Generic sequence timing configuration register 3 (TIME_GEN_SEQ_3)

TIME_GEN_SEQ_3 controls the waveform timing parameters for the generic sequence. For more detail see section 31.3.2, Generic Sequence. All timings are generated from the P1φ clock signal, and delays are added to the timings in clock cycle units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	t0_d12[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5 to 0	t0_d12[5:0]	H'00	R/W	Data to Sequence End Time The time between the data transferring phase and the end of the sequence. The sequence is ended when any other sequence phases are not enabled.

31.2.29 Timing configuration register (TIMINGS_ASYN)

TIMINGS_ASYN controls the timing parameters for two waveforms shown in Figure 31.2 (upper waveform: read; lower waveform: write).

All timings are generated from the P1φ clock signal, and delays are added to the timings in clock cycle units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TRWH[3:0]				TRWP[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7 to 4	TRWH[3:0]	H'0	R/W	NFRE# or NFWE# high hold time
3 to 0	TRWP[3:0]	H'0	R/W	NFRE# or NFWE# pulse width

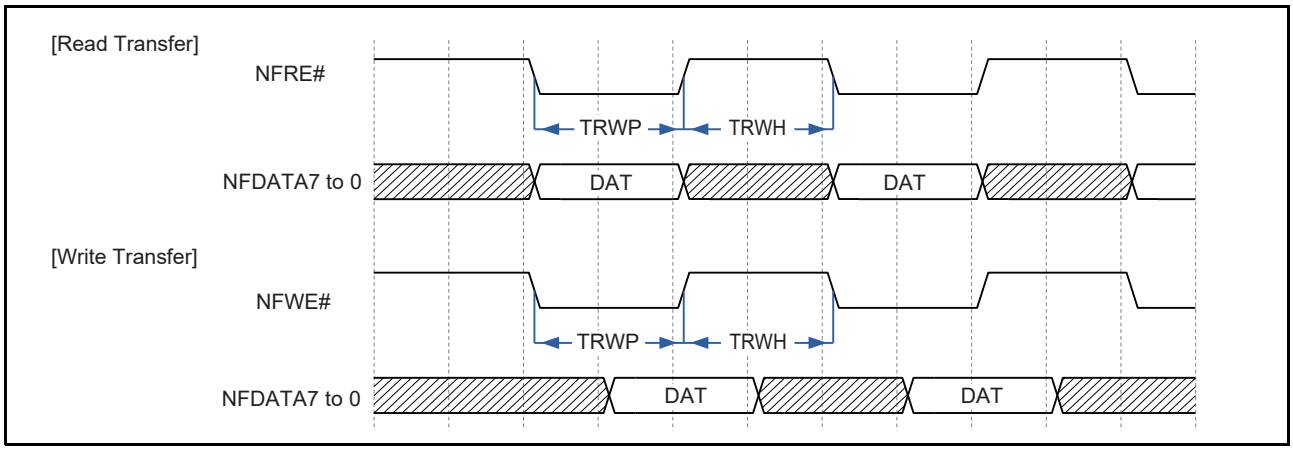
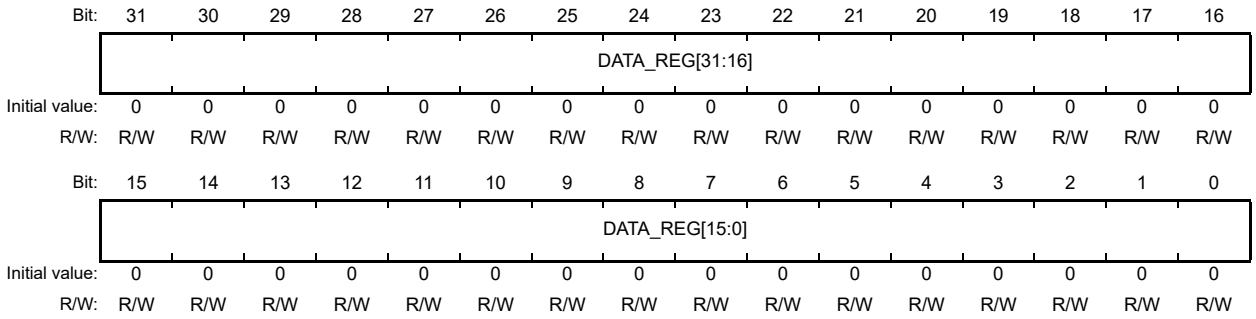


Figure 31.2 Asynchronous Timings

31.2.30 Data register (DATA_REG)

DATA_REG is used for storage of the data that is read in the registered mode. The registered mode is allowed in the read direction only.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA_REG [31:0]	H'0000 0000	R/W	DATA_REG

31.2.31 Data register size selection register (DATA_REG_SIZE)

DATA_REG_SIZE allows the selection of data size in the registered work mode. The data size in the registered mode is limited to four bytes.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DATA_REG_SIZE[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1, 0	DATA_REG_SIZE[1:0]	H'0	R/W	DATA_REG_SIZE register Allows selection of the number of valid bytes in the DATA register: 00: single byte valid 01: two lower bytes valid 10: three lower bytes valid 11: all four bytes valid

31.2.32 FIFO control register (FIFO_INIT)

FIFO_INIT is used to clear (flush) the FIFO.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FIFO_INIT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	FIFO_INIT	0	W	FIFO Initialization bit The setting of this bit causes the flushing of FIFO. It is not necessary to set this bit before sending each command to the NAND Flash memory device. This bit is only used when the previous FIFO contents need to be cleared before a new operation.

31.2.33 FIFO status register (FIFO_STATE)

FIFO_STATE contains the data buffer status.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DF_W_EMPTY	DF_R_FULL	CF_ACCPT_W	CF_ACCPT_R	CF_FULL	CF_EMPTY	DF_W_FULL	DF_R_EMPTY
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7	DF_W_EMPTY	1	R	FIFO empty state bit This bit indicates that there is no data in the FIFO available. This flag is valid for the write direction.
6	DF_R_ FULL	0	R	FIFO full state bit This bit indicates that there is no free space for the data in FIFO available. This flag is valid for the read direction.
5	CF_ACCPT_W	0	R	Command FIFO accept flag – write direction If this flag is set, the next write access will finish with an additional delay.
4	CF_ACCPT_R	1	R	Command FIFO accept flag – read direction If this flag is set, the command FIFO will accept a read transfer with a delay.
3	CF_FULL	0	R	Command FIFO full flag This bit indicates the status of the command FIFO. Do not use this bit to check if the FIFO can accept the next transfer.
2	CF_EMPTY	1	R	Command FIFO empty flag This bit indicates the status of the command FIFO. Do not use this bit to check if the FIFO can accept the next transfer.
1	DF_W_FULL	0	R	FIFO full state bit This bit indicates that there is no free space for the data in FIFO available. This flag is valid for the write direction.
0	DF_R_EMPTY	1	R	FIFO empty state bit This bit indicates that there is no data in the FIFO available. This flag is valid for the read direction.

31.2.34 MLUN register (MLUN)

MLUN contains the LUN address offset bits and number of available LUNs.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	LUN_SEL[1:0]		—	—	—	—	—	MLUN_IDX[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9, 8	LUN_SEL[1:0]	00	R/W	Number of the logical units (LUNs) 00: Two 01: Four 10: Eight
7 to 3	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2 to 0	MLUN_IDX[2:0]	000	R/W	LUN address offset These bits hold the offset to the LUN address from bit zero of the last address byte. These bits indicate the bit index for the address that identifies valid LUNs.

31.2.35 CMD ID initial value register (CMD_MARK)

CMD_MARK allows to write initial value to the command marking generator in the register Slave I/F Unit (SIU).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CMD_ID[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7 to 0	CMD_ID[7:0]	H'00	W	CMD ID initial value

31.3 Operation

31.3.1 Command Generation

This NAND flash controller provides parameterized command sequences to support future functional expansion (new commands) of NAND flash devices.

(1) Instruction Encoding

The controller instruction field is constant and has 32 bits. The instruction field contains the command sequence code and optional parameters. Those parameters are:

- The command codes present in the instruction sequence.
- The flag used to select data destination; possible options are data register and FIFO module.
- The flag used to select data source/sink for the command sequence. The possible choices are the registers or the DMA unit.
- The command sequence code.

If the given command sequence does not use all parameter fields, unused fields are ignored. The instruction encoding scheme is presented in the table below:

Table 31.5 Instruction Encoding

Field Name	Bits	Description
CMD_2	[31:24]	Code of the third command in a sequence
CMD_1/CMD3*1	[23:16]	Code of the second or fourth command in a sequence
CMD_0	[15:8]	Code of the first command in a sequence
DATA_SEL	[7]	Data register/FIFO select 0 – FIFO selected 1 – DATA register selected
INPUT_SEL	[6]	Input module select flag 0 – Select the SIU module as input 1 – Select the DMA module as input
CMD_SEQ	[5:0]	Command code.

Note 1. Depending on the selected command sequence, this field will store the CMD1 or CMD3 code. Both commands are never used in a single sequence.

(2) Command Sequence Encoding

The NAND Flash controller defines the set of commands, addresses and data sequences that allows implementation of all present and many future instructions. The following description of those sequences is adequate to define most of the future NAND Flash device instructions.

Table 31.6 contains the command sequence encoding details. Each sequence is encoded according to the fields defined in GENERIC_SEQUENCE_CTRL.
For more details, section 31.2.3, GENERIC_SEQ register (GEN_SEQ_CTRL).

Table 31.6 Command Sequence Encoding

Sequence Symbol	Sequence Encoding	CMD0	CMD1	CMD2	CMD3	COL _A0*2	COL _A1	ROW _A0	ROW _A1	DATA _EN	DELAY _EN	IMD _SEQ*3
SEQ_0	000000	√	—	—	—	—	—	—	—	—	DELAY_1	—
SEQ_1	100001	√	—	—	—	1	—	—	—	√	—	—
SEQ_2	100010	√	—	—	—	1	—	—	—	√	DELAY_0	—
SEQ_3	000011	√	—	—	—	1	—	—	—	√	DELAY_1	—
SEQ_4	100100	√	—	—	—	—	—	—	—	√	—	√
SEQ_5	100101	√	—	—	—	—	—	3	—	√	—	√
SEQ_6	100110	√	—	√	—	2(1)	—	—	—	√	—	—
SEQ_7	100111	√	—	√	—	2(1)	—	3	—	√	DELAY_0	—
SEQ_8	001000	√	—	—	—	2(1)	—	—	—	√	—	—
SEQ_9	101001	√	√	—	—	2(1)	—	3	—	—	DELAY_1	—
SEQ_10	101010	√	—	√	—	2(1)	—	3	—	√	DELAY_0	—
SEQ_11	101011	√	—	—	—	—	—	—	—	√	DELAY_0	—
SEQ_12	001100	√	√	—	—	2(1)	—	3	—	√	DELAY_1	—
SEQ_13	001101	√	—	—	—	2(1)	—	3	—	√	DELAY_1	—
SEQ_14	001110	√	√	—	—	—	—	3	—	—	DELAY_1	—
SEQ_15	101111	√	—	√	√	2	2	3	3	√	DELAY_0	—
SEQ_17	110001	√	—	—	—	2(1)	—	3	—	√	DELAY_1	—
SEQ_18	110010	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
SEQ_19	010011	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
SEQ_20	110100	√	—	—	—	—	—	3	—	—	DELAY_1	—
SEQ_21	110101	√	—	—	—	1	—	—	—	—	—	—
SEQ_22	110110	√	—	√	—	2(1)	2	—	3	√	DELAY_0	—
SEQ_23	010111	√	√	—	—	—	—	3	—	√	DELAY_1	—
SEQ_24	011000	√	—	√	√	—	—	3	3	—	DELAY_0	—
SEQ_25	111001	√	—	√	√	2(1)	2	3	—	√	—	—

Note 1. SEQ_18 and SEQ_19 are the parameterized Generic Sequences. GEN_SEQ_CTRL defines which parts of sequences are executed.

Note 2. The value given in the brackets relates to the small block mode. In this mode, the controller sends only a single byte as the column address.

Note 3. MD_SEQ - The command will be sent immediately.

Note: Gray rows – read from NAND Flash memory;
White rows – write to NAND Flash memory;
Blue rows – Non- directional commands

(a) SEQ_0 Sequence

This non-directional sequence is composed from only one command. After the command is written to the NAND Flash device, the controller waits until the device goes into the busy state and drives the RnB line low, or sends the READ STATUS command. When delay time (tWB) passes or the device is ready, the sequence ends. The figure below shows the sequence.

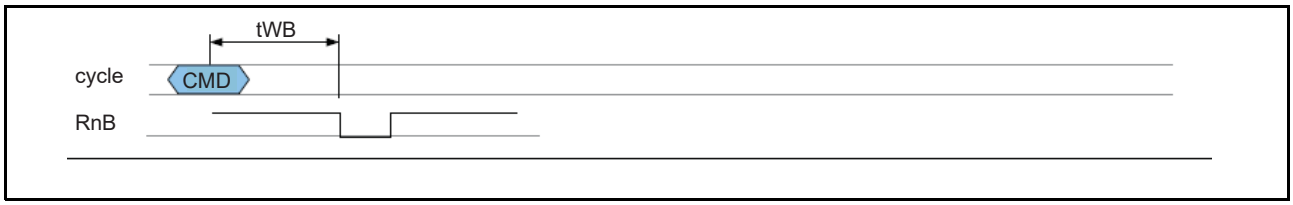


Figure 31.3 SEQ_0 Sequence

(b) SEQ_1 Sequence

This is a read-sequence that is composed from a single command cycle, single address cycle and the single data cycle with a programmable number of read sequences. After the address sequence is finished, the controller measures the standard delay of first data read after the last write (tWHR). Next, the read data words are written to the FIFO module. The input module is selected by the INPUT_SEL field of the COMMAND register, the source for the address is placed in the ADDR0_COL register, and the commend code is stored in the CMD_0 field. The figure below shows the sequence execution:

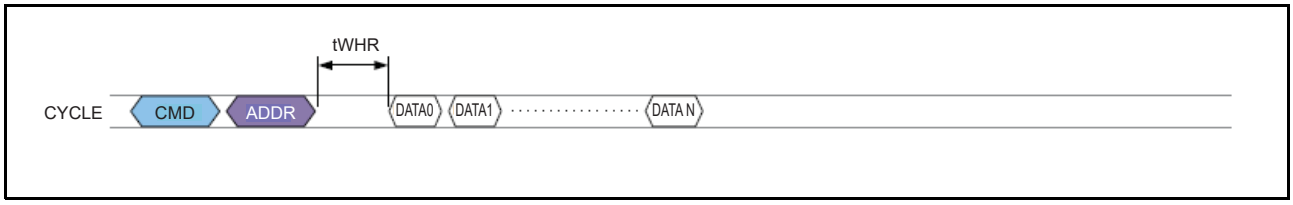


Figure 31.4 SEQ_1 Sequence

(c) SEQ_2 Sequence

This is a read-sequence and it is similar to the SEQ_1 sequence except that after the address cycle the controller expects that device goes to the busy state. The controller checks the state of the RnB line or sends the READ STATUS command as required to obtain the state of the NAND flash device. The figure below shows the sequence execution:

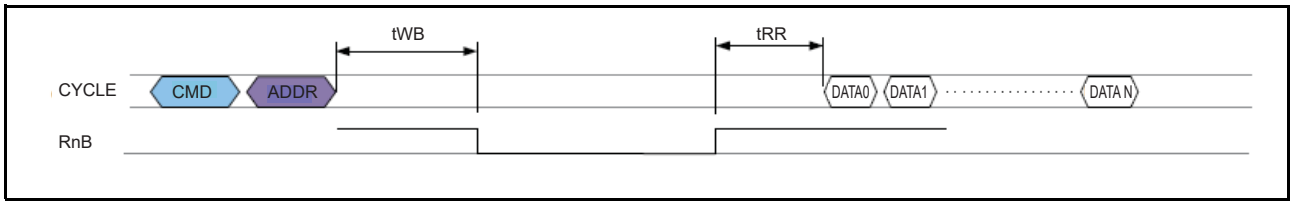


Figure 31.5 SEQ_2 Sequence

(d) SEQ_3 Sequence

This is a write-sequence that is composed from a single command cycle, single address cycle and single data cycle with a programmable number of write sequences. After the address sequence is finished, the controller measures the standard delay of first data write after the last address cycle (tADL). The written words are read from the FIFO module.
The controller checks the state of the RnB line or sends the READ STATUS command as required to obtain the state of the NAND flash device. The input module is selected by the INPUT_SEL field of the COMMAND register; the source for the address is placed in the ADDR0_COL register, and the commend code is stored in the CMD_0 field. The figure below shows the sequence execution:

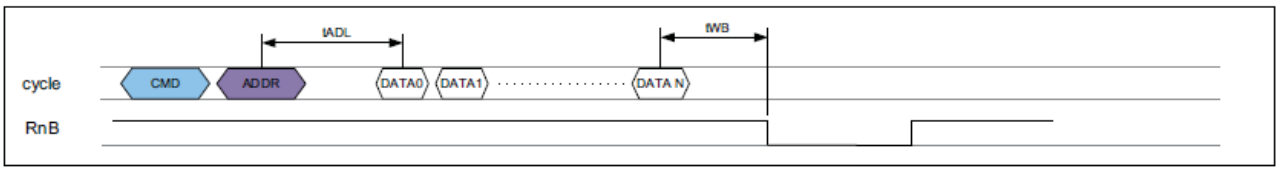


Figure 31.6 SEQ_3 Sequence

(e) SEQ_4 Sequence

This is a special read-sequence that is used to implement the read status command sequences. The command is sent immediately. The sequence is composed of a single command cycle and a single data cycle. Between those cycles, the delay is counted (tWHR). The command code is read from the CMD_0 field.
When the DATA register is selected in the COMMAND register, the data is stored in the DATA register. The user defines the number of data in the DATA_REG_SIZE register. The registered mode is allowed only for the read direction.
When the FIFO register is selected in the COMMAND register, the data is stored in the FIFO. Because user has to change DATA_SIZE register, the command will be sent when all memories are ready and the Controller is in the idle state. The figure below shows the sequence execution:

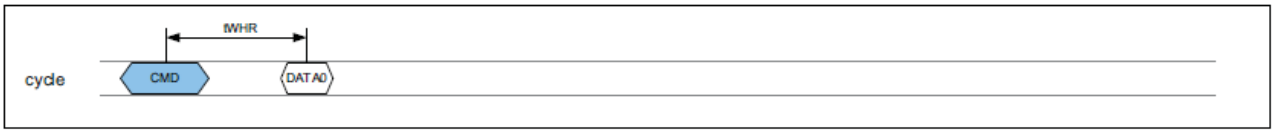


Figure 31.7 SEQ_4 Sequence

(f) SEQ_5 Sequence

This is a read-sequence and it is similar to the SEQ_4 sequence. The command is sent immediately. The only difference is that after the command cycle, an additional address cycle is performed. The ADDR0_ROW register is used in this sequence. The figure below shows the sequence execution:

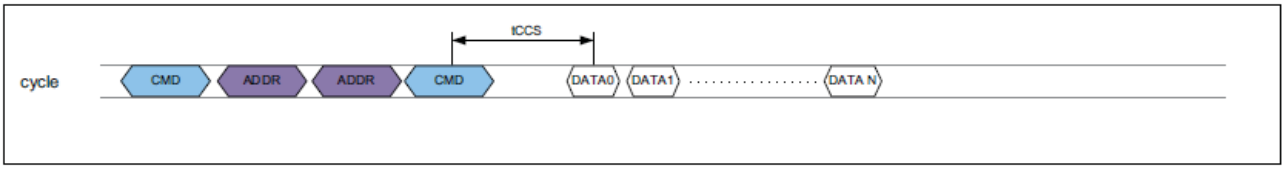


Figure 31.8 SEQ_5 Sequence

(g) SEQ_6 Sequence

This is a read-sequence. The sequence of command cycle, address cycle, command cycle is executed. After that, the delay from the change column to the next operation (tCCS) is measured. Finally, the read data cycle is executed. The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_2 instruction field; the ADDR0_COL register is used in this sequence; the input module is selected by the INPUT_SEL field. The figure below shows the sequence execution:

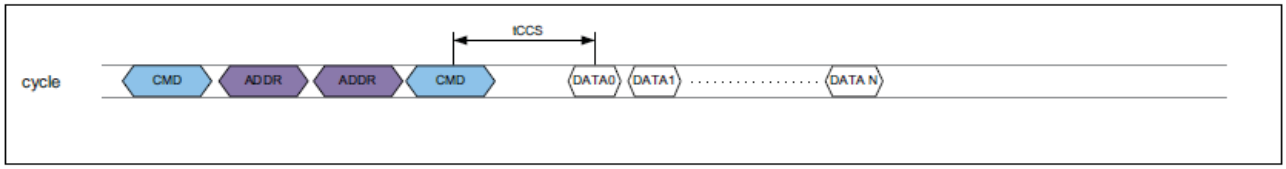


Figure 31.9 SEQ_6 Sequence

(h) SEQ_7 Sequence

This read-sequence is similar to the SEQ_6 sequence, differing only because the address cycle in this sequence is composed of five bytes (ADDR0_COL and ADDR0_ROW) instead of two bytes. All else is the same as in the SEQ_6 sequence. The figure below shows the sequence execution:

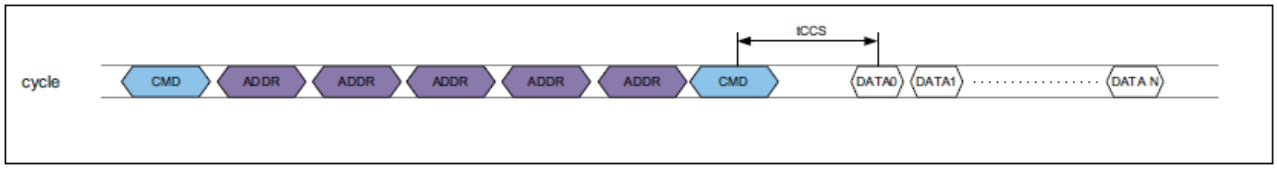


Figure 31.10 SEQ_7 Sequence

(i) SEQ_8 Sequence

This is a write-sequence. First, the sequence of command cycle and two bytes address cycle is executed. Next, the delay after the column address changes (tCCS) is measured. Finally, the single data cycle with programmable number of write sequences is executed. The first command code is encoded in the CMD_0 instruction field; the ADDR0_COL register is used in this sequence; the input module is selected by the INPUT_SEL field. The figure below shows the sequence execution:

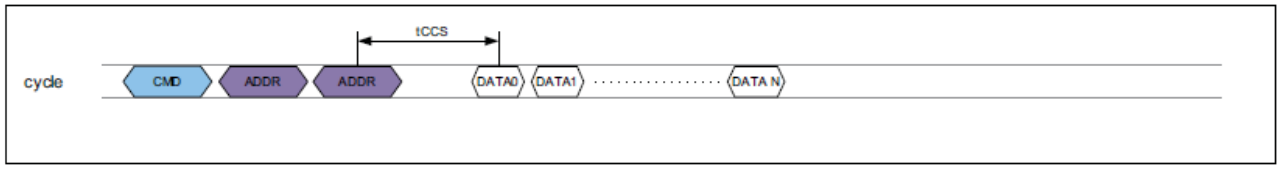


Figure 31.11 SEQ_8 Sequence

(j) SEQ_9 Sequence

This is a non-directional sequence. The first step is to execute the five bytes address command cycle. The controller checks the state of the RnB line or sends the READ STATUS command as required to obtain the state of the NAND flash device. The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the ADDR0_COL and ADDR0_ROW registers are used in this sequence. The figure below shows the sequence execution:

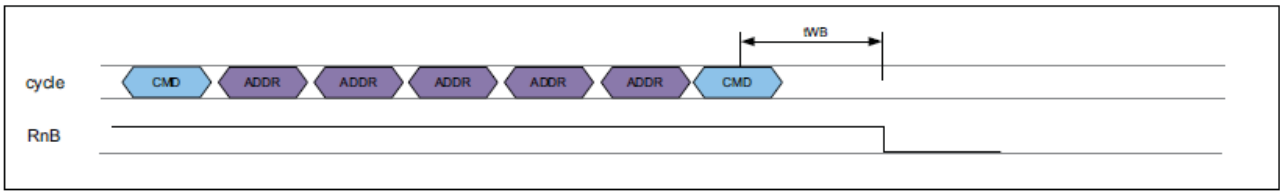


Figure 31.12 SEQ_9 Sequence

(k) SEQ_10 Sequence

This is a non-directional sequence. The first step is to execute the five bytes address command cycle. The controller checks the state of the RnB line or sends the READ STATUS command as required to obtain the state of the NAND flash device. Finally data block is read. The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_2 instruction field; the ADDR0_COL and ADDR0_ROW registers are used in this sequence. The figure below shows the sequence execution:

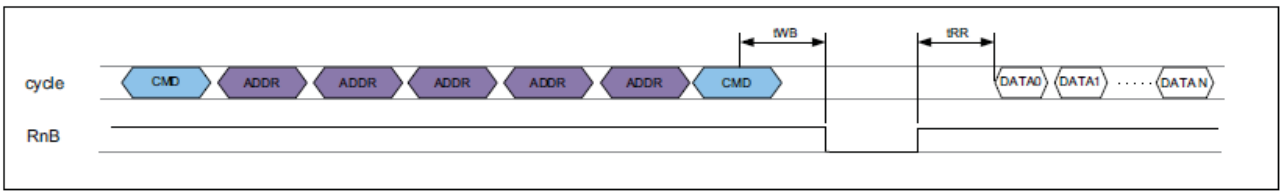


Figure 31.13 SEQ_10 Sequence

(l) SEQ_11 Sequence

This is the read-sequence. The first step is to execute the command cycle. Next, the device goes to the busy state. The controller checks the state of the RnB line or sends the READ STATUS command as required to obtain the state of the NAND flash device. As soon as the device reaches the ready state, the write data cycle with configurable read sequences is executed. The command code is encoded in the CMD_0 instruction field; the input module is selected by the INPUT_SEL field. The number of transferred bytes are configured using the DATA_SIZE register. The figure below shows the sequence execution:

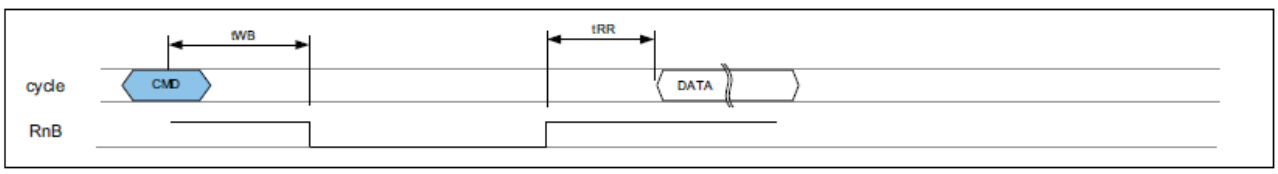


Figure 31.14 SEQ_11 Sequence

(m) SEQ_12 Sequence

This is a write-sequence. The SEQ_12 sequence is a series of command cycle, address cycle, and data cycle with a configurable number of write operations and another command cycle. Between the last address cycle and first data cycle, a delay is measured (tADL) and, after the second command cycle, another delay is measured (tWB). The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the ADDR0_COL and ADDR0_ROW registers are used in this sequence; the input module is selected by the INPUT_SEL field. The figure below shows the sequence execution:

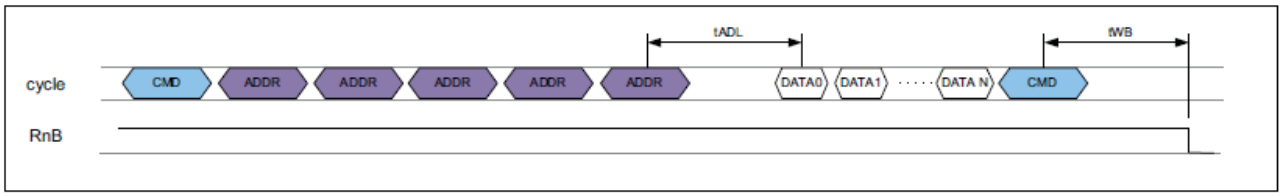


Figure 31.15 SEQ_12 Sequence

(n) SEQ_13 Sequence

This is a write-sequence. The SEQ_13 sequence is a series of command cycle and address cycles, data cycle with a configurable number of write operations. Between the last address cycle and first data cycle, a delay is measured (tADL). The command code is encoded in the CMD_0 instruction field; the ADDR0_COL and ADDR0_ROW registers are used in this sequence; the input module is selected by the INPUT_SEL field. The figure below shows the sequence execution:

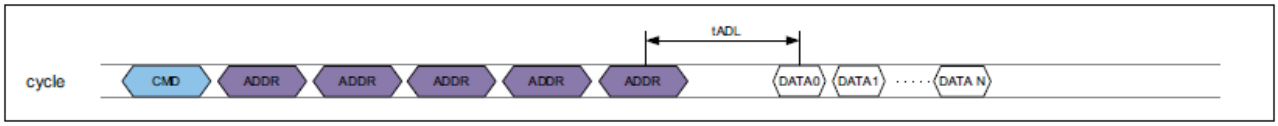


Figure 31.16 SEQ_13 Sequence

(o) SEQ_14 Sequence

This is a non-directional sequence. First, the series of command cycle, address cycle, command cycle is executed. The controller checks the state of the RnB line or sends the READ STATUS command as required to obtain the state of the NAND flash device. The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the ADDR0_ROW and ADDR0_COL registers are used in this sequence. The figure below shows the sequence execution:

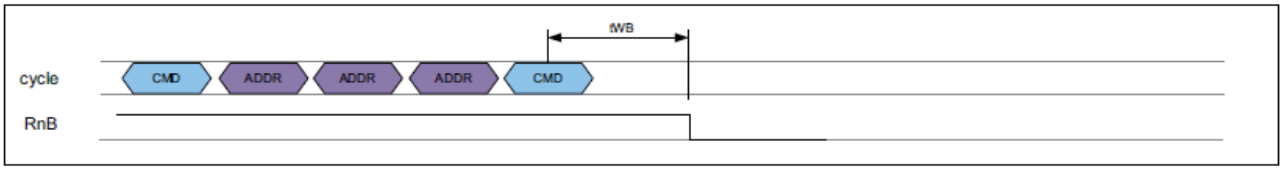


Figure 31.17 SEQ_14 Sequence

(p) SEQ_15 sequence

This is a read-sequence. First, the series of command cycle, address cycle, second command cycle, second address cycle, third command cycle is executed. The controller checks the state of the RnB line or sends the READ STATUS command as required to obtain the state of the NAND flash device. After the NAND Flash device returns to the ready state, the data sequence, with a configurable number of read operations, is executed. The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the third command code is encoded in the CMD_2 instruction field. In this sequence, both address registers are used. The ADDR0_ROW and ADDR0_COL register content is sent after the first command in the sequence, the ADDR1_ROW and ADDR1_COL register content is sent after the second command in the sequence. The figure below shows the sequence execution:

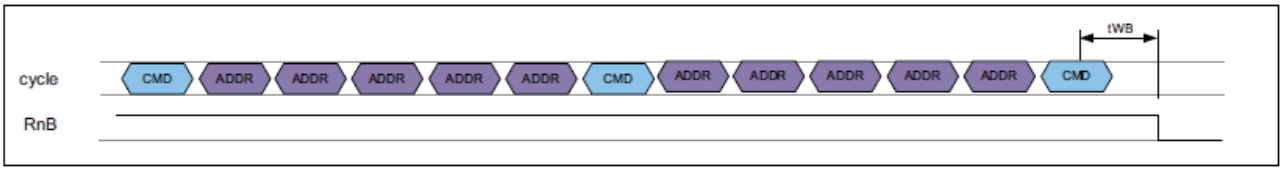


Figure 31.18 SEQ_15 Sequence

(q) SEQ_17 sequence

This is a read-sequence. This sequence is similar to the SEQ_10 sequence, except that the second command cycle is omitted. This sequence is implemented to use small block memories. The controller sends only four bytes of the address when the small block mode is enabled. The figure below shows the sequence execution:

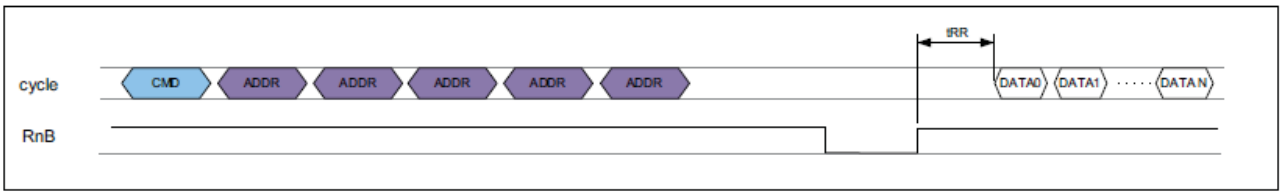


Figure 31.19 SEQ_17 Sequence

(r) SEQ_18 sequence

This generic read sequence is described in detail in section 31.3.2, Generic Sequence.

(s) SEQ_19 sequence

This generic write sequence is described in detail in section 31.3.2, Generic Sequence.

(t) SEQ_20 sequence

This non-directional sequence is composed from one command and three addresses bytes. After the command and addresses are written to the NAND Flash device, the controller waits until the device goes into the busy state and drives the RnB line low, or sends the READ STATUS command. When delay time (tWB) passes or the device is ready, the sequence ends. The figure below shows the sequence execution:

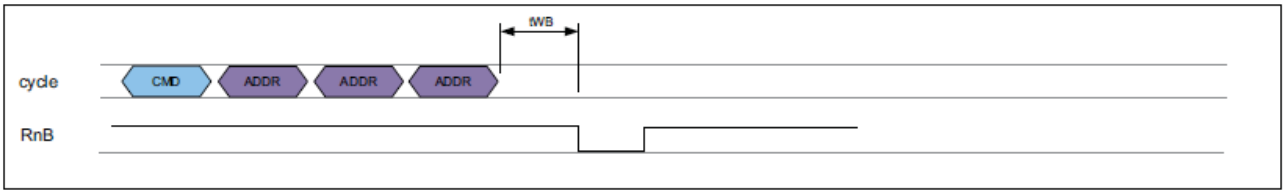


Figure 31.20 SEQ_20 Sequence

(u) SEQ_21 sequence

This non-directional sequence is composed from one command and one addresses byte. After the command and address are written to the NAND Flash device, the sequence ends. The figure below shows the sequence execution:

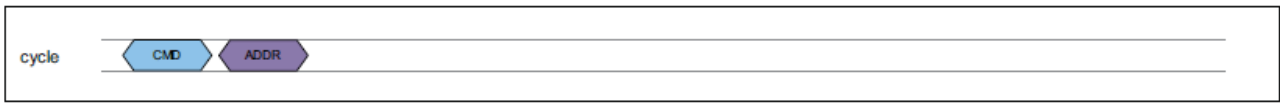


Figure 31.21 SEQ_21 Sequence

(v) SEQ_22 sequence

This is a read-sequence. The first step is to execute the five bytes address command cycle. The controller checks the state of the RnB line or sends the READ STATUS command as required to obtain the state of the NAND flash device. The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_2 instruction field; the ADDR0_COL, ADDR1_COL and ADDR1_ROW registers are used in this sequence. The figure below shows the sequence execution:

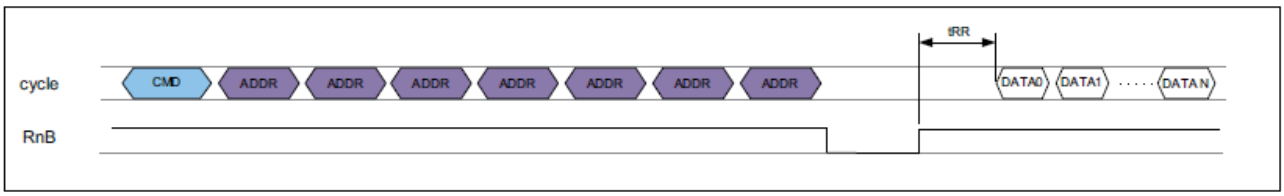


Figure 31.22 SEQ_22 Sequence

(w) SEQ_23 sequence

This is a write-sequence. The SEQ_23 sequence is a series of command cycle, address cycle, and data cycle with a configurable number of write operations and another command cycle. Between the last address cycle and first data cycle, a delay is measured (tADL) and, after the second command cycle, another delay is measured (tWB). The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the ADDR0_ROW registers is used in this sequence; the input module is selected by the INPUT_SEL field. The figure below shows the sequence execution:

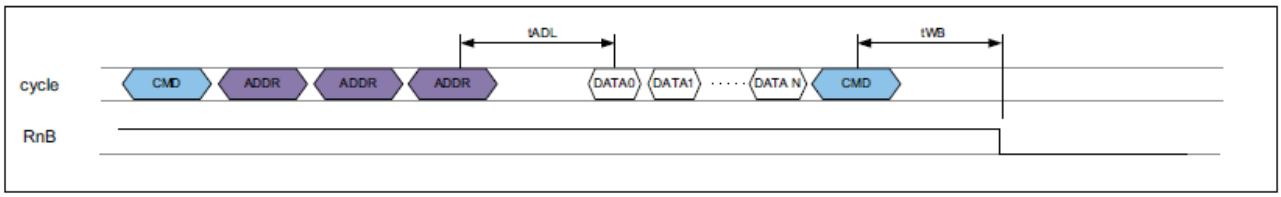


Figure 31.23 SEQ_23Sequence

(x) SEQ_24 sequence

This is a write-sequence. It is composed from the three commands cycles and two addresses cycles. Both addresses cycle contain the row address part. After the last command cycle the tWB delay is measured. The figure below shows the sequence execution.

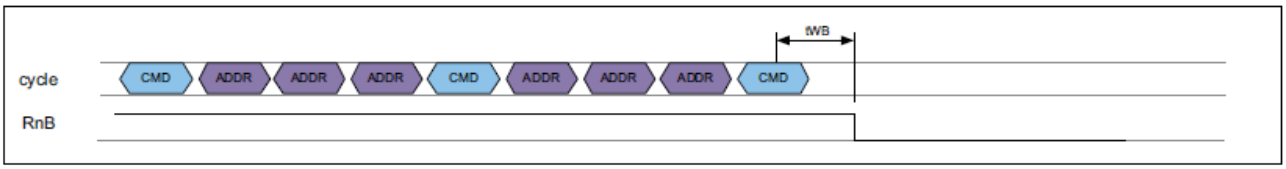


Figure 31.24 SEQ_24 Sequence

(y) SEQ_25 sequence

This is a read-sequence. It is composed from the three commands cycles and two addresses cycles. the first addresses cycle contain the column and row address part. The second address cycle contain only the column address part. After the last command cycle the tWHR delay is measured. The figure below shows the sequence execution.

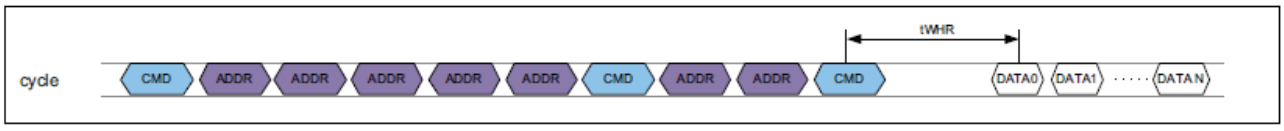


Figure 31.25 SEQ_25 Sequence

(z) SEQ_26 sequence

This is a write-sequence. The SEQ_26 sequence is a series of command cycle, address cycle, and data cycle with a configurable number of write operations and another command cycle. Between the last address cycle and first data cycle, a delay is measured (tADL). The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the ADDR0_ROW registers is used in this sequence; the input module is selected by the INPUT_SEL field. The figure below shows the sequence execution:

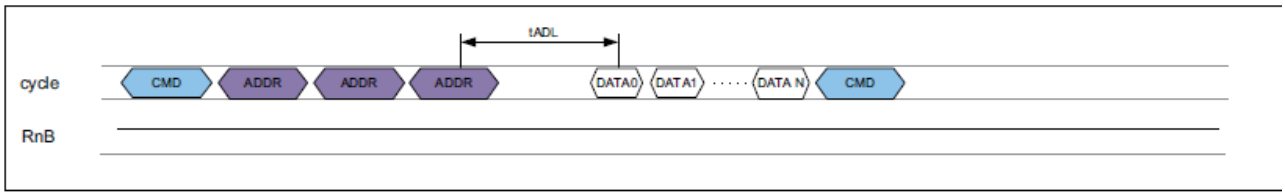


Figure 31.26 SEQ_26 Sequence

31.3.2 Generic Sequence

There will be cases where the set of predefined sequences above will not be sufficient to handle a new command sequence. If that occurs, the generic sequence feature of the NAND flash controller can be used.
This sequence is designed to mimic almost every available command supported by the NAND Flash devices; however, additional effort required to trigger such commands.
Generic sequence is executed in the following steps:

CMD0 – The first command in the sequence

The value of this command is stored in the CMD_0 field of the COMMAND register, described in section 31.2.1, Controller commands register (COMMAND).

ADDR0 – The first address sequence

This is enabled if the COL_A0 and ROW_A0 fields in the GEN_SEQ_CTRL register described in section 31.2.3, GENERIC_SEQ register (GEN_SEQ_CTRL) have values other than zero. In this phase, the address is sent to the NAND Flash device and is read from the ADDR0_COL and ADDR0_ROW registers. The number of the bytes in the address cycle is configured by the COL_A0 and ROW_A0 fields of the GEN_SEQ_CTRL register.

CMD1 – The fourth command in the sequence

The value of this command is stored in the CMD_1 field of the COMMAND register, described in section 31.2.1, Controller commands register (COMMAND). This is enabled by the CMD1_EN field in the GEN_SEQ_CTRL register, described in section 31.2.3, GENERIC_SEQ register (GEN_SEQ_CTRL).

ADDR1 – The second address in the sequence

This is enabled if the COL_A1 and ROW_A1 fields of the GEN_SEQ_CTRL register described in section 31.2.3, GENERIC_SEQ register (GEN_SEQ_CTRL) have a value other than zero. In this phase, the address is sent to the NAND Flash device from the ADDR1_COL and ADDR1_ROW registers.
The number of bytes in the address cycle is configured by the COL_A1 and ROW_A1 fields of the GEN_SEQ_CTRL register.

CMD2 – The third command in the sequence

The value of this command is stored in the CMD_2 field of the COMMAND register described in section 31.2.1, Controller commands register (COMMAND). This is enabled by the CMD2_EN field in the GEN_SEQ_CTRL register, described in section 31.2.3, GENERIC_SEQ register (GEN_SEQ_CTRL).

DELAY0 – Waiting for the device to return to the ready state and continue the sequence.

This is enabled by the DELAY_EN field in the GEN_SEQ_CTRL register, described in section 31.2.3, GENERIC_SEQ register (GEN_SEQ_CTRL). Only one delay phase can be present in the generic sequence. For more details, see section 31.2.3, GENERIC_SEQ register (GEN_SEQ_CTRL).

DATA – The data phase of the sequence

This is enabled by the DATA_EN field in the GEN_SEQ_CTRL register described in section 31.2.3, GENERIC_SEQ register (GEN_SEQ_CTRL). Additionally, the transfer direction must be selected by the sequence number. Sequence number 18 reads data from the NAND Flash memory, sequence number 19 writes data to the NAND Flash memory. The size of the transferred data block is configured by the DATA_SIZE register value.

CMD3 – The second command in the sequence

This is enabled by the CMD3_EN field in the GEN_SEQ_CTRL register described in the section 31.2.3, GENERIC_SEQ register (GEN_SEQ_CTRL). The value of this command is stored in the CMD_3 field of the GEN_SEQ_CTRL register.

DELAY1 – Waiting for the device to return to the ready state and finish the sequence.

This is enabled by the DELAY_EN field in the GEN_SEQ_CTRL register described in section 31.2.3, GENERIC_SEQ register (GEN_SEQ_CTRL). The controller waits for the device to return to the ready state and finish the sequence. Only one delay phase can be present in generic sequence.

The figure below presents the generic sequence composition:

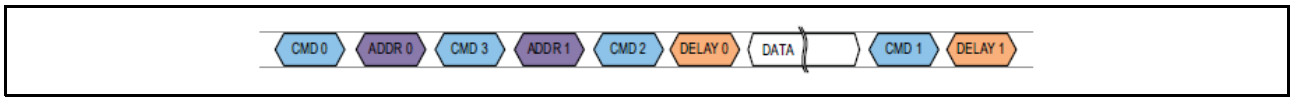


Figure 31.27 Generic Sequence

There are few constraints on the generic sequence usage:

- The DEL0 and DEL1 delay phases cannot both be enabled in a single command sequence.
- The TIMINGS _ASYN register must be set even when the generic sequence is used.

It is possible to force an immediate command sequence execution by enabling the IMD_SEQ bit in the GEN_SEQ_CTRL register. In this case, the triggered command will be executed even if the previously sent command for a selected device is not completed. If both the IMD_SEQ and DATA_EN flags are enabled in a single sequence, then the register must be selected as data source/sink. The IMD_SEQ is valid only for the read direction. This feature is intended to implement all state read operations.
After each step of the generic sequence, the programmable time delay is measured. These delays are configured using the TIME_GEN_SEQ[0-3] registers. Refer to the TIME_GEN_SEQ[0-3] registers description for further information and see figures below.

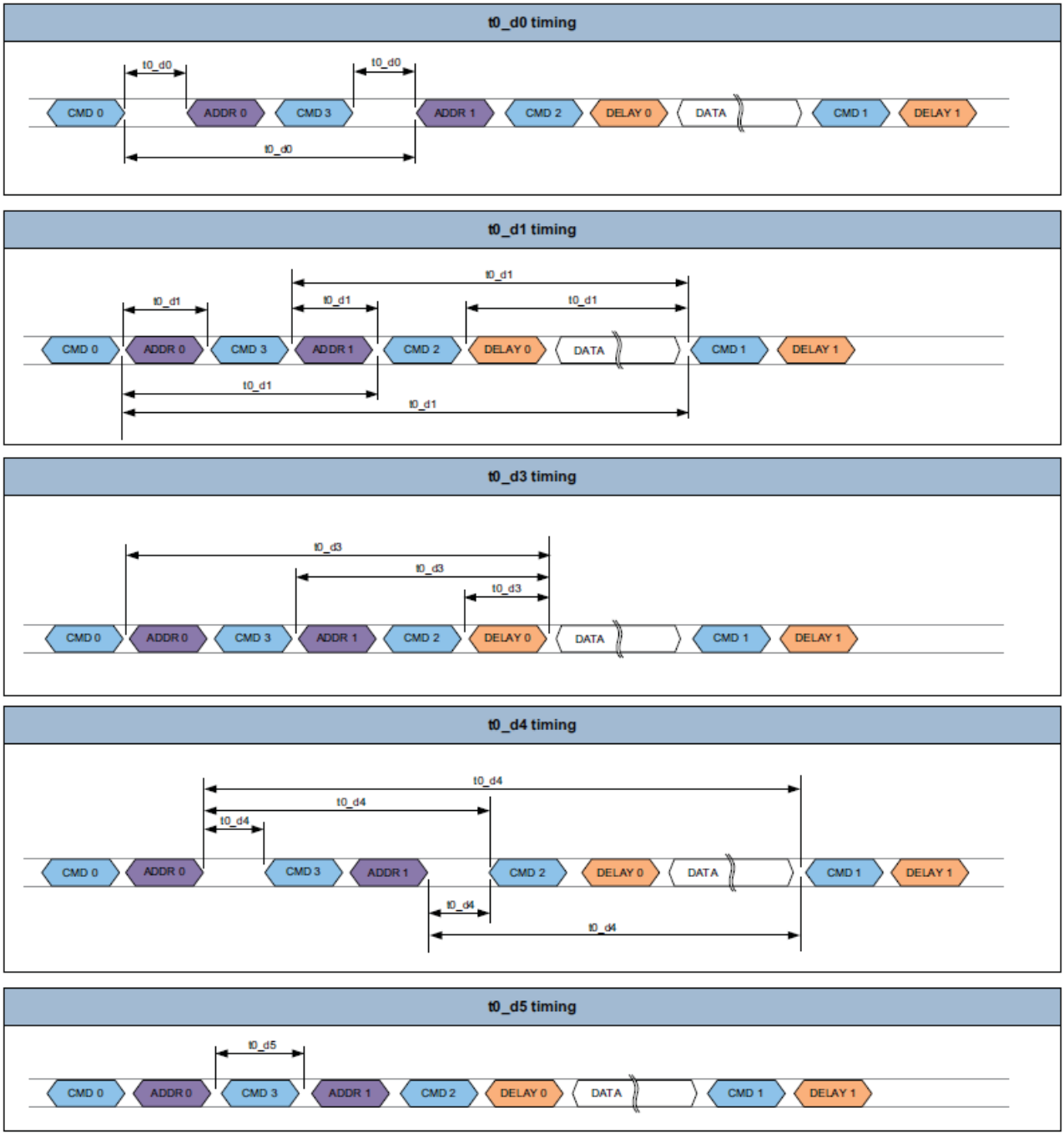


Figure 31.28 Generic Sequence Timing parameters (1)

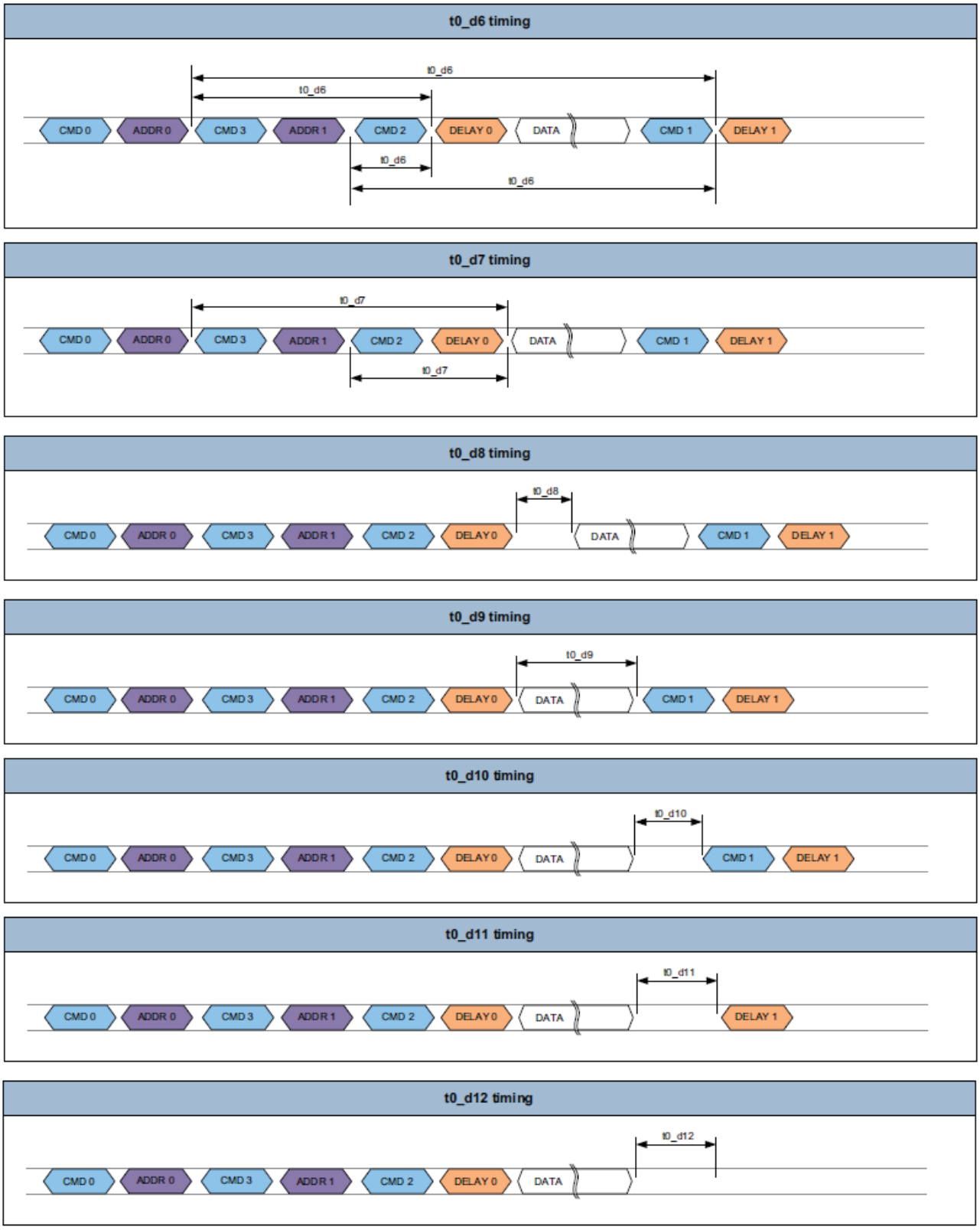


Figure 31.29 Generic Sequence Timing parameters (2)

31.3.3 Instructions

The implementation of the instruction set presented at this point is an example of how to use the instruction encoding scheme presented in section 31.3.1 (1), Instruction Encoding. This chapter and the command sequences presented previously give sufficient information to implement new commands and command sequences for future NAND Flash devices.

(1) Instructions Set

The table below contains the basic instruction defined to implement the all command sequences accessible in the ONFI 1.x standards.

Table 31.7 Instructions set

Instruction	CMD_0	CMD_1/CMD_3	CMD_2	CMD_SEQ	Send when Memory is Busy
Reset Commands					
RESET	H'FF	—	—	SEQ_0	No
Identification Operations					
READ ID	H'90	—	—	SEQ_1	No
READ PARAMETER PAGE	H'EC	—	—	SEQ_2	No
READ UNIQUE ID	H'ED	—	—	SEQ_2	No
Configuration Operations					
GET FEATURES	H'EE	—	—	SEQ_2	No
SET FEATURES	H'EF	—	—	SEQ_3	No
Status Operations					
READ STATUS	H'70	—	—	SEQ_4	Yes
SELECT LUN WITH STATUS	H'78	—	—	SEQ_5	Yes
LUN STATUS	H'71	—	—	SEQ_5	Yes
DEVICE STATUS	H'72	—	—	SEQ_4	Yes
VOLUME SELECT	H'E1	—	—	SEQ_21	Yes
Column Address Operations					
CHANGE READ COLUMN	H'05	—	H'E0	SEQ_6	No
SELECT CACHE REGISTER	H'06	—	H'E0	SEQ_7	No
CHANGE WRITE COLUMN	H'85	—	—	SEQ_8	No
CHANGE ROW ADDRESS	H'85	H'11	—	SEQ_12	No
Read Operations					
READ PAGE	H'00	—	H'30	SEQ_10	No
READ PAGE CACHE	H'31	—	—	SEQ_11	No
READ PAGE CACHE LAST	H'3F	—	—	SEQ_11	No
READ MULTIPLANE	H'00	H'32	—	SEQ_9	No
TWO PLANE PAGE READ	H'00	H'00	H'30	SEQ_15	No
QUEUE PAGE READ	H'07	—	H'37	SEQ_22	No
Program Operations					
PROGRAM PAGE	H'80	H'10	—	SEQ_12	No
PROGRAM PAGE IMD	H'80	H'10	—	SEQ_23	No
PROGRAM PAGE DEL	H'80	H'13	—	SEQ_23	No
PROGRAM PAGE 1	H'80	—	—	SEQ_13	No
PROGRAM PAGE CACHE	H'80	H'15	—	SEQ_12	No
PROGRAM MULTIPLATE	H'80	H'11	—	SEQ_12	No
WRITE PAGE	H'10	—	—	SEQ_0	No

Table 31.7 Instructions set

Instruction	CMD_0	CMD_1/CMD_3	CMD_2	CMD_SEQ	Send when Memory is Busy
WRITE PAGE CACHE	H'15	—	—	SEQ_0	No
WRITE MULTIPLATE	H'11	—	—	SEQ_0	No
Erase Operation					
ERASE BLOCK	H'60	H'D0	—	SEQ_14	No
ERASE MULTIPLATE	H'60	H'D1	—	SEQ_14	No
Copyback Operation					
COPYBACK READ	H'00	—	H'35	SEQ_10	No
COPYBACK PROGRAM	H'85	H'10	—	SEQ_9	No
COPYBACK PROGRAM1	H'85	—	—	SEQ_13	No
COPYBACK MULTIPLATE	H'85	H'11	—	SEQ_12	No
OTP Operation					
PROGRAM OTP	H'A0	H'10	—	SEQ_12	No
DATA PROTECT OTP	H'A5	H'10	—	SEQ_9	No
PAGE READ OTP	H'AF	—	H'30	SEQ_10	No

(2) RESET Command

(a) Command Description

The RESET command is used to put a target into a known condition and to abort command sequences in progress.

(b) Command Encoding

The RESET instruction uses the SEQ_0 commands sequence. The command is encoded, as shown in the table below:

Table 31.8 RESET Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'FF	—	—	SEQ_0_ID

(3) READ ID Command

(a) Command Description

The READ ID command is used to read identifier codes programmed into the target. This command is accepted by the target only when all LUNs on the target are in the IDLE state.
When the command is followed by an address cycle of H'00, the target returns a 5-byte identifier code that includes the manufacturer's ID, device configuration, and part-specific information.
When the READ ID command is followed by an address cycle of H'20, the target returns the 4-byte ONFI identifier code.

(b) Command Encoding

The READ ID instruction uses the SEQ_1 commands sequence. The command is encoded, as shown in the table below:

Table 31.9 READ ID Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'90	0/1	0/1	SEQ_1_ID

(4) READ PARAMETER PAGE Command

(a) Command Description

The READ PARAMETER PAGE command is used to read the ONFI parameter page programmed into the target. This command is accepted by the target only when all LUNs on the target are idle.
When the command is followed by an address cycle of H'00, the target goes into busy state. After the read process is completed, the controller enables the data output mode to read the parameter page.

(b) Command Encoding

The READ PARAMETER PAGE instruction uses the SEQ_2 commands sequence. The command is encoded, as shown in the table below:

Table 31.10 READ PARAMETER PAGE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'EC	0/1	0/1	SEQ_2_ID

(5) READ UNIQUE ID Command

(a) Command Description

The READ UNIQUE ID instruction is used to read a unique identifier programmed into the target. This command is accepted by the target only when all LUNs on the target are idle.
When the address cycle of H'00 is written to the target, then the target goes into busy state. After the read process is complete, the controller enables the data output mode to read the unique ID. Sixteen copies of the unique ID data are stored in the device. Each copy is 32 bytes. The first 16 bytes are unique data and the second 16 bytes are the complement of the first 16 bytes. The application program will XOR the first 16 bytes with the second 16 bytes. If the result is 16 bytes of H'FF, then that copy of the unique ID data is correct. In the event that there is a non-H'FF result, the application program repeats the XOR operation on a subsequent copy of the unique ID data.

(b) Command Encoding

The READ UNIQUE ID instruction uses the SEQ_2 commands sequence. The command is encoded, as shown in the table below:

Table 31.11 READ UNIQUE ID Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'ED	0/1	0/1	SEQ_2_ID

(6) GET FEATURES Command

(a) Command Description

The GET FEATURES instruction reads the sub-feature parameters (P1-P4) from the specified feature address. This command is accepted by the target only when all LUNs on the target are idle.
When the H'EE command is followed by a feature address, the target goes into busy state. After the target internal read operation completes, the controller enables the data output mode to read the sub-feature parameters.

(b) Command Encoding

The GET FEATURES instruction uses the SEQ_2 commands sequence. The command is encoded, as shown in the table below:

Table 31.12 GET FEATURES Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'EE	0/1	0/1	SEQ_2_ID

(7) SET FEATURES Command

(a) Command Description

The SET FEATURES instruction writes the sub-feature parameters (P1-P4) to the specified feature address to enable or disable target-specific features. This command is accepted by the target only when all LUNs on the target are idle.
The H'EF command is followed by a valid feature address. The possible address value depends on the features set implemented in the target device. The address cycle is followed by the configurable number of data cycles. Values of the address and data encoding scheme allowed are found in the device vendor documentation.

(b) Command Encoding

The SET FEATURES instruction uses the SEQ_3 commands sequence. The command is encoded, as shown in the table below:

Table 31.13 SET FEATURES Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'EF	0/1	0/1	SEQ_3_ID

(8) READ STATUS Command

(a) Command Description

Each LUN provides its status independently of other LUNs on the same target through its 8-bit status register. Once the READ STATUS instruction is issued, status register output is enabled. The contents of the status register are returned on DQ[7:0] for each data output request.
The READ STATUS command returns the status of the most recently selected LUN.

(b) Command Encoding

The READ STATUS instruction uses the SEQ_4 commands sequence. The command is encoded, as shown in the table below:

Table 31.14 READ STATUS Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'70	0/1	—	SEQ_4_ID

This instruction has special meaning because it can be executed when the target is in the BUSY state.
The FIFO cannot be used to access the read data because it can be occupied by the operation under execution. For the sequence the data field of command must select the DATA_REG register as data destination. The DATA_REG_SIZE must be select single byte.

(9) DEVICE STATUS Command

(a) Command Description

Each LUN provides its status independently of other LUNs on the same target through its 8-bit status register. Once the DEVICE STATUS instruction is issued, status register output is enabled. The contents of the status register are returned on DQ[7: 0] for each data output request.
The DEVICE STATUS command returns the status of the most recently selected LUN.

(b) Command Encoding

The DEVICE STATUS instruction uses the SEQ_4 commands sequence. The command is encoded, as shown in the table below:

Table 31.15 DEVICE STATUS Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'72	0/1	—	SEQ_4_ID

This instruction has special meaning because it can be executed when the target is in the BUSY state.
The FIFO cannot be used to access the read data because it can be occupied by the operation under execution. For the sequence the data field of command must select the DATA_REG register as data destination. The DATA_REG_SIZE must be select single byte.

(10) VOLUME SELECT Command

(a) Command Description

This command is accepted by all initialized devices that share a CE pin. The command may be executed with any volume on the target in any state. When the VOLUME SELECT command is issued, all volumes with unselected volume addresses will be deselected to save power. If the Volume address entered is invalid or does not match any appointed volume address, all volume addresses will be deselected. If the VOLUME SELECT command is not issued after CE high time then all volumes revert to their previous state.

(b) Command Encoding

The VOLUME SELECT instruction uses the SEQ_21 commands sequence. The command is encoded, as shown in the table below:

Table 31.16 VOLUME SELECT Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'E1	0/1	—	SEQ_21_ID

(11) SELECT LUN WITH STATUS Command

(a) Command Description

Each LUN provides its status independently of other LUNs on the same target through its 8-bit status register. Once the SELECT LUN WITH STATUS instruction is issued, status register output is enabled. The contents of the status register are returned on DQ[7:0] for each data output request.
The SELECT LUN WITH STATUS command returns the state of the selected LUN.

(b) Command Encoding

The SELECT LUN WITH STATUS instruction uses the SEQ_5 commands sequence. The command is encoded, as shown in the table below:

Table 31.17 SELECT LUN WITH STATUS Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'78	0/1	—	SEQ_5_ID

This instruction has special meaning because it can be executed when the target is in the BUSY state.
The FIFO cannot be used to access the read data because it can be occupied by the operation under execution. For the sequence the data field of command must select the DATA_REG register as data destination. The DATA_REG_SIZE must be select single byte.

(12) LUN STATUS Command

(a) Command Description

Each LUN provides its status independently of other LUNs on the same target through its 8-bit status register. Once the SELECT LUN WITH STATUS instruction is issued, status register output is enabled. The contents of the status register are returned on DQ[7: 0] for each data output request.
The LUN STATUS command returns the status of the selected LUN.revert to their previous state.

(b) Command Encoding

The LUN STATUS instruction uses the SEQ_5 commands sequence. The command is encoded, as shown in the table below:

Table 31.18 LUN STATUS Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H’71	0/1	—	SEQ_5_ID

This instruction has special meaning because it can be executed when the target is in the BUSY state.
The FIFO cannot be used to access the read data because it can be occupied by the operation under execution. For the sequence the data field of command must select the DATA_REG register as data destination. The DATA_REG_SIZE must be select single byte.

(13) CHANGE READ COLUMN Command

(a) Command Description

The CHANGE READ COLUMN command changes the column address of the selected cache register and enables data output of the last selected LUN. This command is accepted by the selected LUN when it is ready. Writing H’05 to the target command register, followed by two column address cycles containing the column address, followed by the H’E0 command puts the selected LUN into data output mode. The selected LUN stays in data output mode until another valid command is issued.

(b) Command Encoding

The CHANGE READ COLUMN instruction uses the SEQ_6 commands sequence. The command is encoded, as shown in the table below:

Table 31.19 CHANGE READ COLUMN Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
H’E0	—	H’05	0/1	—	SEQ_6_ID

(14) SELECT CACHE REGISTER Command

(a) Command Description

The SELECT CACHE REGISTER command enables data output from the addressed LUN and the cache register at the specified column address. This command is accepted by a LUN when it is ready. Writing H'06 to the target command register, followed by two column address cycles, three row address cycles, and finally H'E0 enables data output mode on the addressed LUN and the cache register at the specified column address.

(b) Command Encoding

The SELECT CACHE REGISTER instruction uses the SEQ_7 commands sequence. The command is encoded, as shown in the table below:

Table 31.20 SELECT CACHE REGISTER Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
H'E0	—	H'06	0/1	—	SEQ_7_ID

(15) CHANGE WRITE COLUMN Command

(a) Command Description

The CHANGE WRITE COLUMN command changes the column address of the selected cache register and enables data input on the last selected LUN. Writing the H'85 to the target internal command register, followed by two column address cycles containing the column address puts the selected LUN into data input mode.

(b) Command Encoding

The CHANGE WRITE COLUMN instruction uses the SEQ_8 commands sequence. The command is encoded, as shown in the table below:

Table 31.21 CHANGE WRITE COLUMN Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'85	—	—	SEQ_8_ID

(16) CHANGE ROW ADDRESS Command

(a) Command Description

The CHANGE ROW ADDRESS command changes the row address (block and page) where the cache register contents are to be programmed in the NAND array. It also changes the column address of the selected cache register and enables data input on the specified LUN.

(b) Command Encoding

The CHANGE ROW ADDRESS instruction uses the SEQ_12 commands sequence. The command is encoded, as shown in the table below:

Table 31.22 CHANGE ROW ADDRESS Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H'11	H'85	0	—	SEQ_12_ID

(17) READ PAGE Command

(a) Command Description

The READ PAGE command copies a page from the NAND Flash array to its respective cache register and enables data output. This command is accepted by the LUN when it is ready. To read a page from the NAND Flash array, the controller writes the H'00 command to the target internal command register, then writes 5 address cycles to the address registers, and concludes with the H'30 command. The selected LUN is in the busy state while the data are transferred from the NAND flash array to the corresponding cache register. When the LUN is ready, data output is enabled for the cache register linked to the plane addressed in the READ PAGE command. The controller reads the programmed number of bytes to the FIFO.

(b) Command Encoding

The READ PAGE instruction uses the SEQ_10 commands sequence. The command is encoded, as shown in the table below:

Table 31.23 READ PAGE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
H'30	—	H'00	0/1	0/1	SEQ_10_ID

(18) READ PAGE CACHE Command

(a) Command Description

The READ PAGE CACHE command reads the next sequential page within a block into the data register, while the previous page is output from the cache register. To issue this command, the controller writes H'31 to the target internal command register. After this command is issued, the R/B# goes LOW and the LUN goes into busy state. After that, the R/B# signal goes high and the LUN becomes busy and enters the cache operation state, in which the cache register is available and the specified page is copied from the NAND flash array to the data register. At this point, data is read from the cache register.

(b) Command Encoding

The READ PAGE CACHE instruction uses the SEQ_11 commands sequence. The command is encoded, as shown in the table below:

Table 31.24 READ PAGE CACHE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'31	0/1	0/1	SEQ_11_ID

(19) READ PAGE CACHE LAST Command

(a) Command Description

The READ PAGE CACHE LAST command ends the READ PAGE CACHE sequence and copies a page from the data register to the cache register. This command is accepted by the LUN when it is ready. To issue the READ PAGE CACHE LAST command, the controller writes H'3F to the target internal command register. After this command is issued, R/B# goes LOW and the LUN goes into busy state. Afterwards, the R/B# goes HIGH and the LUN is ready. At this point, data from the targets cache register is read into the FIFO.

(b) Command Encoding

The READ PAGE CACHE LAST instruction uses the SEQ_11 commands sequence. The command is encoded, as shown in the table below:

Table 31.25 READ PAGE CACHE LAST Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'3F	0/1	0/1	SEQ_11_ID

(20) READ MULTIPLANE Command

(a) Command Description

The READ MULTIPLANE command queues a plane to transfer data from the NAND array to its cache register. This command can be issued one or more times. Each time a new plane address is specified, that plane is also queued for data transfer. To select the final plane and to begin the read operation for all previously queued planes, issue the READ PAGE command. All queued planes will transfer data from the NAND array to their cache registers.

(b) Command Encoding

The READ MULTIPLANE instruction uses the SEQ_9 commands sequence. The command is encoded, as shown in the table below:

Table 31.26 READ MULTIPLANE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H'32	H'00	—	—	SEQ_9_ID

(21) QUEUE PAGE READ Command

(a) Command Description

The QUEUE PAGE READ operation allows for partial-page reads by using a 7-address cycle. The first two bytes of the address cycle indicate the length of the page to read - those bytes are stored in ADDR0 register, followed by the column and row addresses - those bytes are stored in the ADDR1 register.
This can help overall performance should only a portion of the page data be needed because only the codeword containing the requested data will have ECC decoded. This command is exclusive for the CLEAR NAND devices.

(b) Command Encoding

The QUEUE PAGE READ instruction uses the SEQ_22 commands sequence. The command is encoded, as shown in the table below:

Table 31.27 QUEUE PAGE READ Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
H'37	—	H'07	—	—	SEQ_22_ID

(22) TWO PLANE PAGE READ command

(a) Command Description

This command was implemented to preserve the compatibility with the ONFI 1.x and some older devices.
The TWO PLANE PAGE READ (H'00-H'00-H'30) operation is similar to the PAGE READ (H'00-H'30) operation. It transfers two pages of data from the NAND Flash array to the data registers. Each page must be from a different plane on the same die. The software is responsible for generating correct addresses for the requested pages. Both the ADDR0 and ADDR1 address registers are used in this case.

(b) Command Encoding

The TWO PLANE PAGE READ instruction uses the SEQ_15 commands sequence. The command is encoded, as shown in the table below. In this case, both the address registers are used.

Table 31.28 QUEUE PAGE READ Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
H'30	H'00	H'00	0/1	0/1	SEQ_15_ID

(23) PROGRAM PAGE Command

(a) Command Description

The PROGRAM PAGE command allows the application program to input data to a cache register and moves the data from the cache register to the specified block and page address in the array of the selected LUN. This command is accepted by the LUN when it is ready. To input a page to the cache register and move it to the NAND array at the block and page address specified, the controller writes H'80 to the target internal command register. Then five address cycles containing the column address and row address are written. Data input cycles follow. Serial data is input, beginning at the column address specified. When data input is complete, the controller writes H'10 to the target internal command register. The selected LUN goes into the busy state.

(b) Command Encoding

The PROGRAM PAGE instruction uses the SEQ_12 commands sequence. The command is encoded, as shown in the table below:

Table 31.29 PROGRAM PAGE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H'10	H'80	0	0/1	SEQ_12_ID

(24) PROGRAM PAGE IMMEDIATE Command

(a) Command Description

The PROGRAM PAGE IMMEDIATE command allows the application program to input data to a cache register and moves the data from the cache register to the specified block and page address in the array of the selected LUN. This command is accepted by the LUN when it is ready. To input a page to the cache register and move it to the NAND array at the block and page address specified, the controller writes H'80 to the target internal command register. Then five address cycles containing the column address and row address are written. Data input cycles follow. Serial data is input, beginning at the column address specified. When data input is complete, the controller writes H'10 to the target internal command register. The selected LUN goes into the busy state. This command writes only the row address to the NAND flash device.

(b) Command Encoding

The PROGRAM PAGE IMMEDIATE instruction uses the SEQ_23 commands sequence. The command is encoded, as shown in the table below:

Table 31.30 PROGRAM PAGE IMMEDIATE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H'10	H'80	0	0/1	SEQ_23_ID

(25) PROGRAM PAGE DELAYED Command

(a) Command Description

The device's internal controller can automatically manage multi-plane programming. It does this with PROGRAM PAGE DELAYED command. When this command issued, the controller will delay issuing the program operation to the array until the address for the subsequent program operation is examined. If that operation allows the previous operation to complete as part of multi-plane operation, the controller will issue a multi-plane program to the array. Multi-plane operations are only completed if an LUN address from plane 0 is issued prior to a LUN address from plane 1. If the subsequent program operation does not allow the multi-plane operation, the controller will immediately start the previous program. It is presumed that the application program may use this command to initiate all the program operations. In this way the application program does not have to maintain information regarding multi-plane operation usage.

(b) Command Encoding

The PROGRAM PAGE DELAY instruction uses the SEQ_12 commands sequence. The command is encoded, as shown in the table below:

Table 31.31 PROGRAM PAGE DELAY Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H'13	H’80	0	0/1	SEQ_23_ID

(26) PROGRAM PAGE 1 Command

(a) Command Description

The PROGRAM PAGE 1 command allows the application program to input data to a cache register and moves the data from the cache register to the specified block and page address in the array of the selected LUN. This command is accepted by the LUN when it is ready. To input a page to the cache register and move it to the NAND array at the block and page address specified, the controller writes H’80 to the target internal command register. Then five address cycles containing the column address and row address are written. Data input cycles follow. Serial data is input, beginning at the column address specified. When data input is complete, the commands sequence ends.

(b) Command Encoding

The PROGRAM PAGE 1 instruction uses the SEQ_13 commands sequence. The command is encoded, as shown in the table below:

Table 31.32 PROGRAM PAGE 1 Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H’80	0	0/1	SEQ_13_ID

(27) PROGRAM PAGE CACHE Command

(a) Command Description

The PROGRAM PAGE CACHE command allows the controller to input data to a cache register, copies the data from the cache register to the data register, and then moves the data register contents to the specified block and page address in the array of the selected LUN. After the data is copied to the data register, the cache register is available for additional PROGRAM PAGE CACHE or PROGRAM PAGE commands. The PROGRAM PAGE CACHE command is accepted by the LUN when it is ready.
To input a page to the cache register, the controller writes H'80 to the target internal command register. Then five address cycles containing the column address and row address are written. Data input cycles follow. Serial data is input, beginning at the column address specified. When data input is complete, H'15 is written to the command register. The selected LUN becomes busy, data is copied from the cache register to the data register that has become available through a previous program cache operation, and then the contents of the data register begins to be moved to the specified block and page address.

(b) Command Encoding

The PROGRAM PAGE CACHE instruction uses the SEQ_12 commands sequence. Command is encoded as shown in the table below.

Table 31.33 PROGRAM PAGE CACHE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H'15	H'80	0	0/1	SEQ_12_ID

(28) PROGRAM MULTIPLANE Command

(a) Command Description

The PROGRAM MULTIPLANE command enables the controller to input data to the cache register for the addressed plane and to place the cache register data in the queue to be moved to the NAND array. This command can be issued one or more times. Each time a new plane address is specified that plane is also queued for data transfer. This command is accepted by the LUN when it is ready. The controller writes H'80 to the target internal command register. Then five address cycles containing the column address and row address are written. Data input cycles follow. Serial data is input beginning at the column address specified. When data input is complete, the controller writes H'11 to the target internal command register.

(b) Command Encoding

The PROGRAM PAGE CACHE instruction uses the SEQ_12 commands sequence. Command is encoded as shown in the table below.

Table 31.34 PROGRAM MULTIPLANE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H'11	H'80	0	0/1	SEQ_12_ID

(29) WRITE PAGE Command

(a) Command Description

The WRITE PAGE command allows the controller to move data from the targets cache register to the NAND array. This command is accepted by the LUN when it is ready. The controller writes H’10 to the target internal command register.

(b) Command Encoding

The WRITE PAGE instruction uses the SEQ_0 commands sequence. The command is encoded, as shown in the table below:

Table 31.35 WRITE PAGE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H’10	—	—	SEQ_0_ID

(30) WRITE PAGE CACHE Command

(a) Command Description

The WRITE PAGE CACHE command allows the controller to move data from the targets cache register to the targets data register. This command is accepted by the LUN when it is ready. The controller writes H’15 to the target internal command register.

(b) Command Encoding

The WRITE PAGE CACHE instruction uses the SEQ_0 commands sequence. The command is encoded, as shown in the table below:

Table 31.36 WRITE PAGE CACHE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H’15	—	—	SEQ_0_ID

(31) WRITE MULTIPLANE Command

(a) Command Description

The WRITE MULTIPLANE command allows the controller to queue data from the targets cache register to the NAND array. This command is accepted by the LUN when it is ready. The controller writes H’11 to the target internal command register.

(b) Command Encoding

The WRITE MULTIPLANE instruction uses the SEQ_0 commands sequence. The command is encoded, as shown in the table below:

Table 31.37 WRITE MULTIPLANE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H’11	—	—	SEQ_0_ID

(32) ERASE BLOCK Command

(a) Command Description

The ERASE BLOCK command erases the specified block in the NAND array. This command is accepted by the LUN when it is ready. To erase a block, the controller writes H’60 to the target internal command register. Then three address cycles containing the row address are written; the column address is ignored. Finally, H’D0 is written to the target internal command register.

(b) Command Encoding

The ERASE BLOCK instruction uses the SEQ_14 commands sequence. The command is encoded, as shown in the table below:

Table 31.38 ERASE BLOCK Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H’D0	H’60	—	—	SEQ_14_ID

(33) ERASE MULTIPLANE Command

(a) Command Description

The ERASE MULTIPLANE command queues a block in the specified plane to be erased from the NAND array. This command can be issued one or more times. Each time a new plane address is specified, that plane is also queued for a block to be erased. This command is accepted by the LUN when it is ready. To queue a block to be erased, the controller writes H’60 to the command register.
Then three address cycles containing the row address are written; the column address is ignored.
Finally, H’D1 is written to the command register.

(b) Command Encoding

The ERASE MULTIPLANE instruction uses the SEQ_14 commands sequence. The command is encoded, as shown in the table below:

Table 31.39 ERASE MULTIPLANE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H’D1	H’60	—	—	SEQ_14_ID

(34) COPYBACK READ Command

(a) Command Description

The COPYBACK READ command is functionally identical to the READ PAGE command, except that H'35 is written to the target internal command register instead of H'30. For more details, see section 31.3.4 (17), READ PAGE Command.

(b) Command Encoding

The COPYBACK READ instruction uses the SEQ_10 commands sequence. The command is encoded, as shown in the table below:

Table 31.40 COPYBACK READ Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
H'35	—	H'00	0/1	0/1	SEQ_10_ID

(35) COPYBACK PROGRAM Command

(a) Command Description

The Copyback function reads a page of data from one location and then moves that data to a second location. The COPYBACK PROGRAM command is functionally identical to the PROGRAM PAGE command, except that when H'85 is written to the target internal command register, the cache register contents are not cleared. The SEQ_9 command sequence does not have the data phase, so the data from the cache register are written into the second location without modification. If data must be written with modification to the second location, the SEQ_12 command sequence, which includes the data phase, is used.

(b) Command Encoding

The COPYBACK PROGRAM instruction uses the SEQ_9 commands sequence. The command is encoded, as shown in the table below:

Table 31.41 COPYBACK PROGRAM Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H'10	H'85	—	—	SEQ_9_ID

(36) COPYBACK PROGRAM 1 Command

(a) Command Description

The COPYBACK PROGRAM 1 command is functionally identical to the PROGRAM PAGE 1 command, except that when H’85 is written to the target internal command register, the cache register contents are not cleared. See section 31.3.3 (26), PROGRAM PAGE 1 Command for further details.

(b) Command Encoding

The COPYBACK PROGRAM 1 instruction uses the SEQ_13 commands sequence. The command is encoded as shown in the table below:

Table 31.42 COPYBACK PROGRAM 1 Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	—	H'85	0	—	SEQ_13_ID

(37) COPYBACK MULTIPLANE Command

(a) Command Description

The COPYBACK MULTIPLANE command is functionally identical to the PROGRAM MULTIPLANE command, except that when H’85 is written to the target internal command register, cache register contents are not cleared. See section 31.3.3 (28), PROGRAM MULTIPLANE Command for further details.

(b) Command Encoding

The COPYBACK MULTIPLANE instruction uses the SEQ_12 commands sequence. The command is encoded, as shown in the table below:

Table 31.43 COPYBACK MULTIPLANE Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H’11	H’85	0	0/1	SEQ_12_ID

(38) PROGRAM OTP Command

(a) Command Description

The PROGRAM OTP command is used to write data to the pages within the OTP area. An entire page can be programmed at one time, or a page can be partially programmed up to four times. There is no ERASE operation for the OTP pages. To use the PROGRAM OTP command, the controller issues the H'A0 command. Next, 5 address cycles are issued. The address write is followed by a programmable number of data cycles. After data input is complete, the controller issues the H'10 command. The internal control logic automatically executes the proper programming algorithm and controls the necessary timing for programming and verification.

(b) Command Encoding

The PROGRAM OTP instruction uses the SEQ_12 commands sequence. The command is encoded as shown in the table below:

Table 31.44 PROGRAM OTP Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H'10	H'A0	0	0/1	SEQ_12_ID

(39) DATA PROTECT OTP Command

(a) Command Description

The DATA PROTECT OTP command is used to protect all the data in the OTP area. After the data is protected, it cannot be further programmed. When the OTP area is protected, the pages within the area are no longer programmable and cannot be unprotected. To use the DATA PROTECT OTP command, the controller issues the H'A5 command. Next, the controller issues the following 5 addresses cycles.
Finally, the H'10 command is issued.

(b) Command Encoding

The DATA PROTECT OTP instruction uses the SEQ_9 commands sequence. The command is encoded, as shown in the table below:

Table 31.45 DATA PROTECT OTP Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
—	H'10	H'A5	—	—	SEQ_9_ID

(40) PAGE READ OTP Command

(a) Command Description

The PAGE READ OTP command is used to read data from a page within the OTP area. An OTP page within the OTP area is available for reading data whether or not the area is protected. To use the PAGE READ OTP command, the controller issues the H'AF command. Next, 5 address cycles are issued.
Finally, the H'30 command is issued. After internal read from the NAND matrix is ended, the data is copied to the FIFO.

(b) Command Encoding

The PAGE READ OTP instruction uses the SEQ_9 commands sequence. The command is encoded, as shown in the table below:

Table 31.46 PAGE READ OTP Instruction Encoding

CMD_2	CMD_1/CMD_3	CMD_0	DATA_SEL	INPUT_SEL	CMD_SEQ
H'30	—	H'AF	0/1	0/1	SEQ_10_ID

31.3.4 Multi LUN Work Mode

The multi LUN work mode is enabled by setting the MLUN_EN bit in the CONTROL register. In this mode, the memory is divided into multiple LUNs, each of which is handled as a separate target. The active LUN number is decoded directly from the address value. The following parameters apply to the multi LUN work mode.

MLUN_IDX field in the MLUN register
This parameter provides the bit index for the last address byte, which contains the LUN selection bits. This parameter must be set according to the datasheet for the NAND Flash device that is in use.
The LUN_SEL bits in the MLUN register are used to set the number of LUNs in the device. The LUN_STATUS_0 register holds indicators of the states of the LUNs. Each effective bit corresponds to the state of a single LUN.

STATE_MASK field in the STATUS_MASK register
This parameter is used to mask parts of the LUN status byte that will be ignored during LUN ready/busy checks.

31.3.5 Remapping Mechanism

The remapping mechanism supports the bad blocks management (BBM) solution. The hardware remapping mechanism relieves software from the time-consuming operations of finding the physical address for the given linear address in the requested operation. The software initializes only the remapping tables for uses in the application of the NAND Flash device. It sorts those tables in ascending order, and then the whole operation of searching tables and substituting addresses is accomplished automatically.

The remapping solution uses two groups of the control registers:

- The pointer register DEV0_PTR. This register stores the address in the internal memory where record tables used by the BBM mechanism are placed.
- The size register DEV0_SIZE. This register stores the number of records in a table.

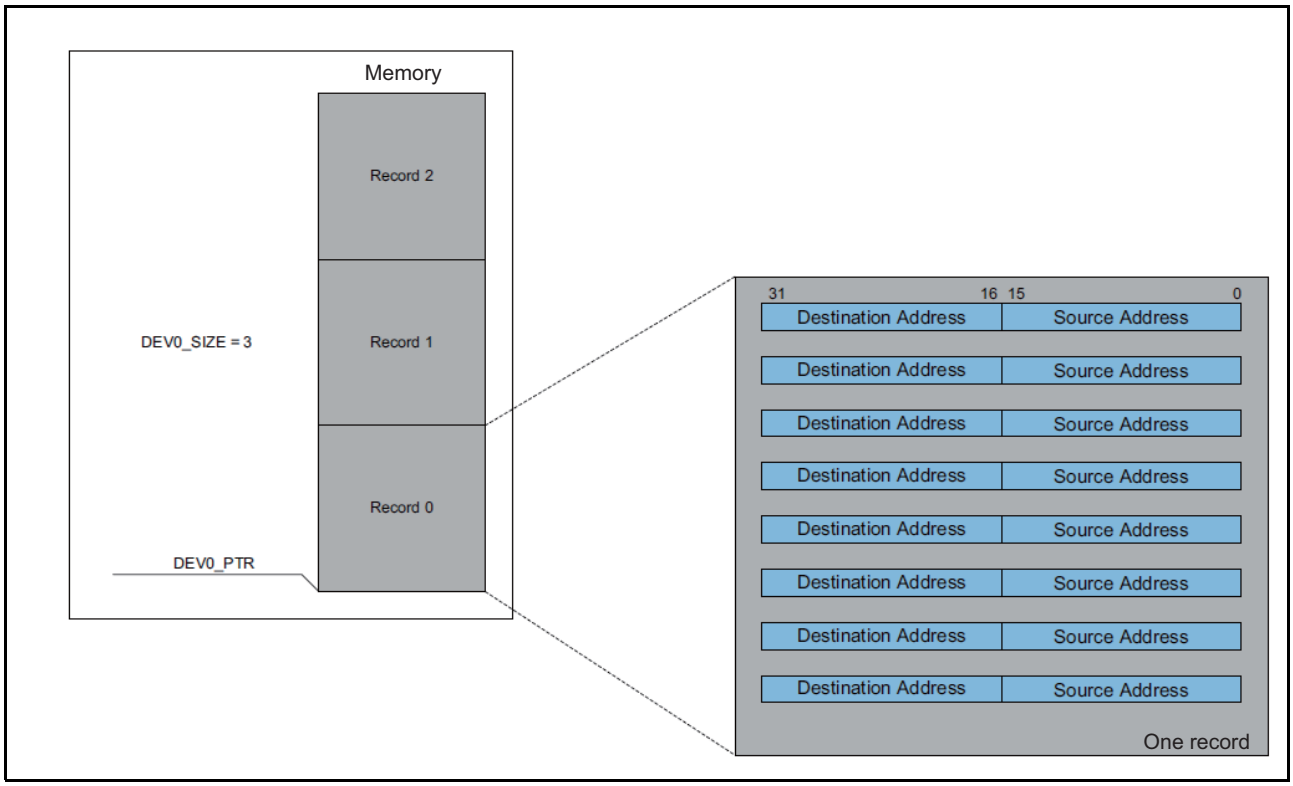


Figure 31.30 Example of BBM Records

Note: Store the block numbers of the destination and source addresses for transfer.
The source addresses for transfer should be arranged in ascending order in the record tables.
If some addresses in a record are not to be used, set the values for these addresses to H'FFFF_FFFF.

31.3.6 Interrupts Mechanism

The NAND Flash controller interrupt system uses two control registers:

The interrupt mask register – each bit of this register masks a single interrupt. The register is described in more detail in section 31.2.6, Interrupts mask register (INT_MASK).
The interrupt flag register – each bit of this register is an active flag from the single interrupt source.
The register is described in more detail in section 31.2.7, Interrupts status register (INT_STATUS).

In case an interrupt event occurs, i.e. the respective interrupt flag in INT_STATUS is 1, and its respective mask bit in INT_MASK is 1, the interrupt signal INTNFMA is asserted.
Note that the interrupt mask bits in INT_MASK affects only the interrupt signal INTNFMA, but not the flags in INT_STATUS.
A set flag in INT_STATUS must be cleared by the application program.
The available interrupt sources and their respective INT_STATUS flags are:

INT_STATUS.CMD_END_INT_FL

Command sequence finished interrupt – This interrupt occurs when the previously triggered command sequence is finished and the new one can be started. The command sequence is marked as finished when the full sequence is executed or when the NAND Flash device goes into the busy state.

INT_STATUS.ECC_INT0_FLAG

The ECC module detects an uncorrectable error in the transmitted data.
The ECC module detects when the configured errors threshold level is exceeded.

INT_STATUS.MEM0_RDY_INT_FL

The memory device is ready – This interrupt occurs when the NAND Flash device finishes executing the programmed command sequence and is ready for the new one. Each NAND Flash device has a single interrupt flag. The difference between “command sequence finished” interrupt and “memory device is ready” interrupt is presented in Figure 31.30.

INT_STATUS.TRANS_ERR_FL

The error on the slave interface during access to the controller FIFO – This interrupt occurs if the access to the FIFO memory has the opposite direction to the current FIFO configuration: the FIFO is read when it is configured for write, or the FIFO is write when it is configured for read.

INT_STATUS.STAT_ERR_INT0_FL

Most recently finished operation on the memory device failed. This applies to PROGRAM PAGE and ERASE BLOCK operations. It is not valid following a READ-series operation.

INT_STATUS.PG_SZ_ERR_INT_FL

Data Size error flag
When the ECC is enabled, the value written into the DATA_SIZE register has some restrictions.
Interrupt condition is met when the value written to the DATA_SIZE register is not correct.

INT_STATUS.DMA_INT_FL

DMA transfer ended flag

INT_STATUS.DATA_REG_INT_FL

Data in DATA_REG is available.

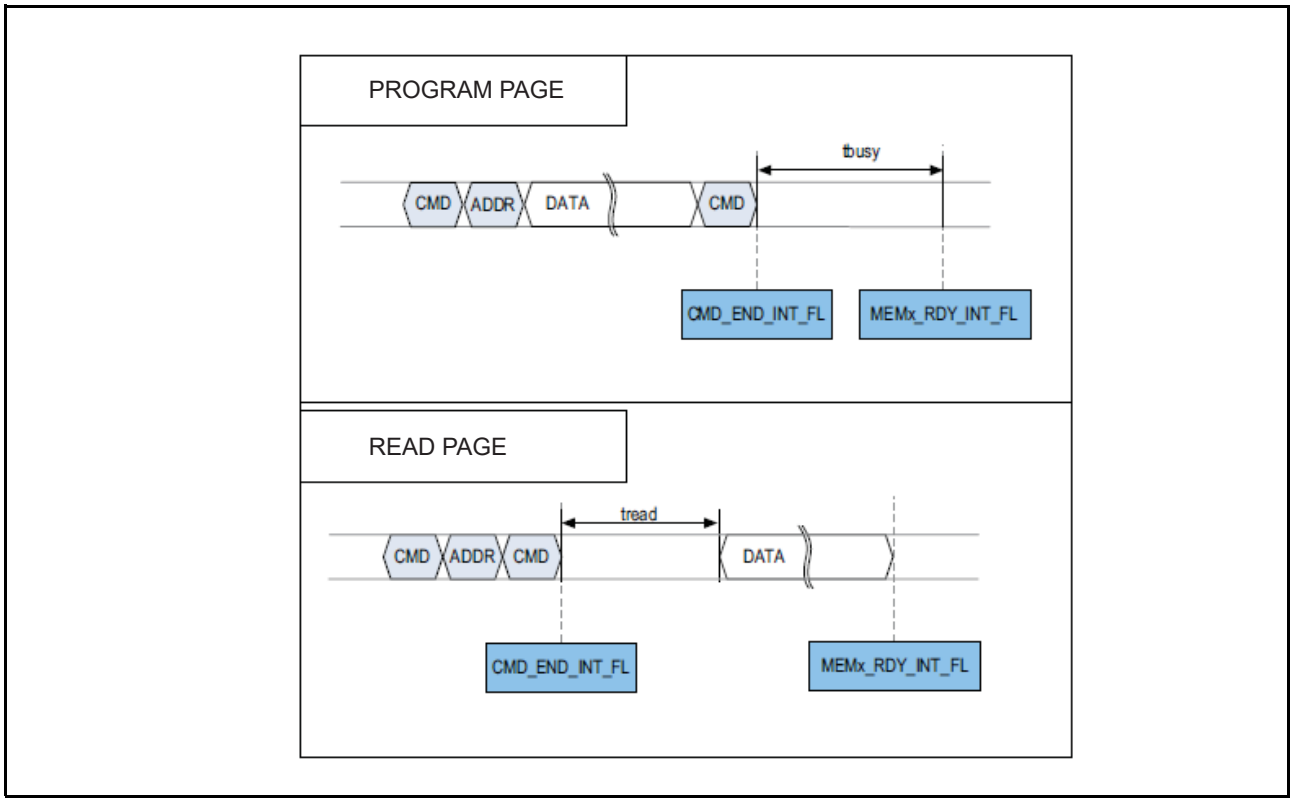


Figure 31.31 Command Sequence End and Memory-Ready Interrupts

31.3.7 Setup and Configuration

The CONTROL register is the main control register in the NAND Flash controller.
The following bits configure basic settings of the controller:

- INT_EN – Bit which enables Global Interrupt.
- ECC_EN – Bit which enables Hardware ECC.
- BLOCK_SIZE – Bits which configure block size.
- BBM_EN – Bit which enables remapping process.
- PROT_EN – Bit which enables Protect mechanism.
- ADDRx_AUTO_INCR - Addresses auto increment for row address register 0 or 1.
- SMALL_BLOCK_EN – Bit which enables Small Block Mode.
- MLUN_EN – Bit which enables Multi LUN mode.
- AUTO_READ_STAT_EN – Bit which activates automatic read status after the PROGRAM PAGE and ERASE BLOCK commands.
- READ_STATUS_EN – Bit which chooses whether the controller checks RnB lines or sends READ_STATUS commands.
- ECC_BLOCK_SIZE – Bits which define ECC Block Size.

The registers described below are configured according to the settings of other bits in the CONTROL register:

1. If INT_EN bit is set, the software must write the mask into the INT_MASK register, which masks the selected interrupts source in the NAND Flash controller.
2. If ECC_EN bit is set, the software must correctly configure the ECC module by writing the appropriate configuration into the ECC_CTRL. Additionally, the software configures the offset in the ECC_OFFSET register. In small block mode, the value in the ECC_OFFSET is ignored and the correction words are located in the NAND Flash memory device, right behind the data.
3. The write number of the data which will be transferred by the controller (DATA_SIZE).
   When ECC is enabled, there are some restrictions to the DATA_SIZE value.
4. If the BBM_EN bit is set, the software must initialize the remapping tables (DEVn_PTR and DEVn_SIZE registers).
5. Additionally, the software must configure time parameters which can be found in the TIMINGS_ASYN register. Additionally, the software must configure the TIME_SEQ_0 and TIME_SEQ_1 registers.

When the NAND Flash controller uses DMA to transfer data, the software must configure the DMA_ADDR, DMA_CTRL and DMA_CNT registers. The software can modify these registers before any transfer or during the initialization procedure.

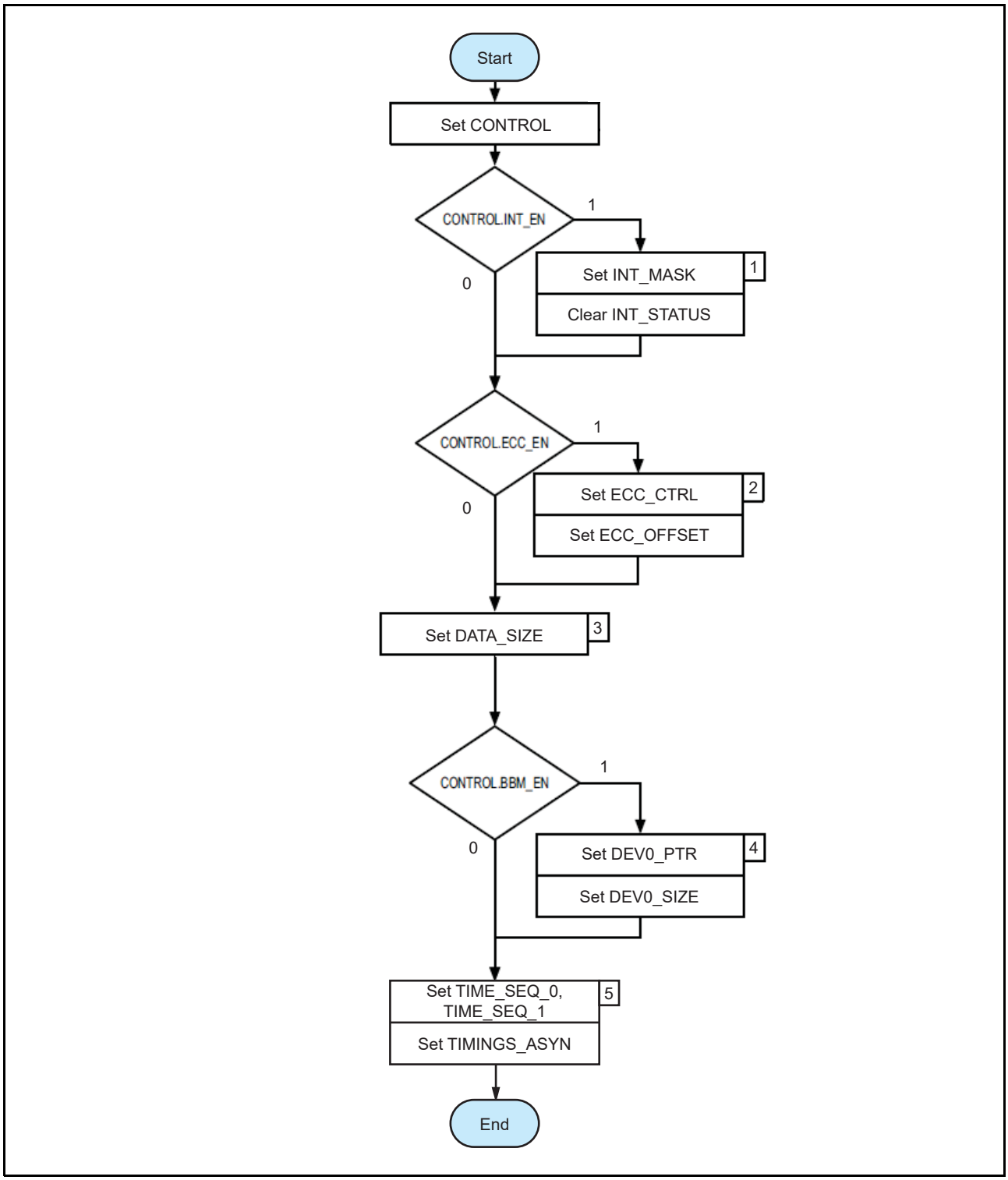


Figure 31.32 Configuration

(1) Send Data to NAND Flash via Slave Interface

1. The NAND Flash controller must be correctly configured before sending data to the NAND Flash memory device. The setup process is described in detail in section 31.3.7, Setup and Configuration and section 31.2, Register Description.
2. Write the address of the data in NAND Flash memory into the address registers 0 (ADDR0_COL and ADDR0_ROW). Write the number of data which you want to read (DATA_SIZE register).
3. To use the simplest program command, write H'0010 800C to the COMMAND register (PROGRAM PAGE command, FIFO selected, slave interface selected).
4. Write data to the FIFO using the FIFO_DATA register. Data is sent to the NAND Flash memory device.

When the memory is ready for further work, MEM0_ST is set.

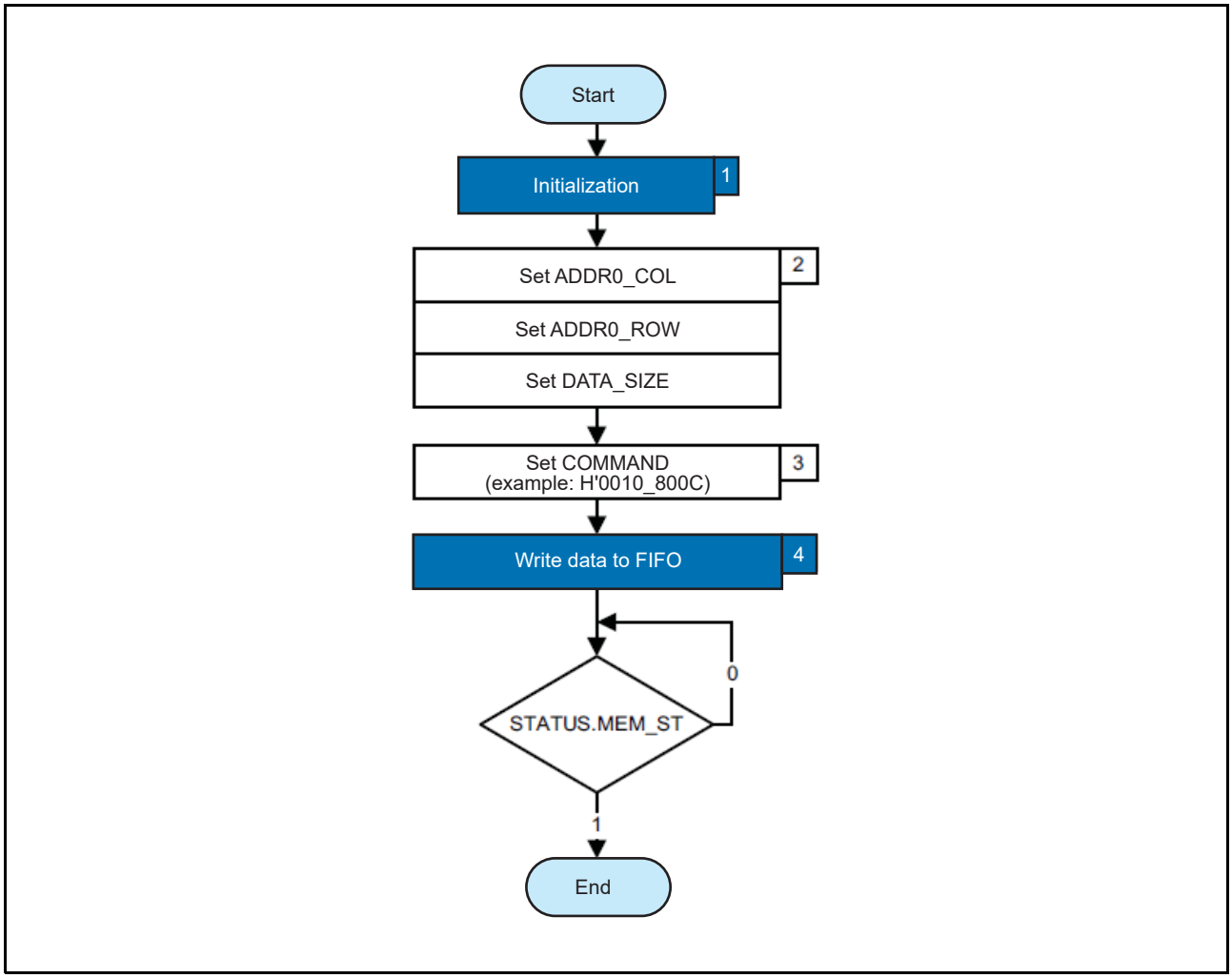


Figure 31.33 Write data to NAND Flash Memory via Slave Interface

(2) Read Data from NAND Flash via Slave Interface

1. The NAND Flash controller must be correctly configured before sending data to the NAND Flash memory device. The setup process is described in detail in section 31.3.7, Setup and Configuration and section 31.2, Register Description.
2. Write the address of the data in NAND Flash memory into the address registers 0 (ADDR0_COL and ADDR0_ROW).
3. To use the simplest read command, write H'3000 002A to the COMMAND register (READ PAGE command, FIFO selected, slave interface selected).
4. The application program can read data immediately after sending the command but in this case reading will be preceded by a very few WAIT replies from the NAND flash controller.
   If this situation shall be avoided the application program must read FIFO_STATE register and wait when CF_EMPTY bit is set. After that wait for the DF_R_EMPTY bit in the FIFO_STATE register to be clear.
5. Read data from the FIFO using the FIFO_DATA register.

When the memory is ready for further work, MEM0_ST is set.

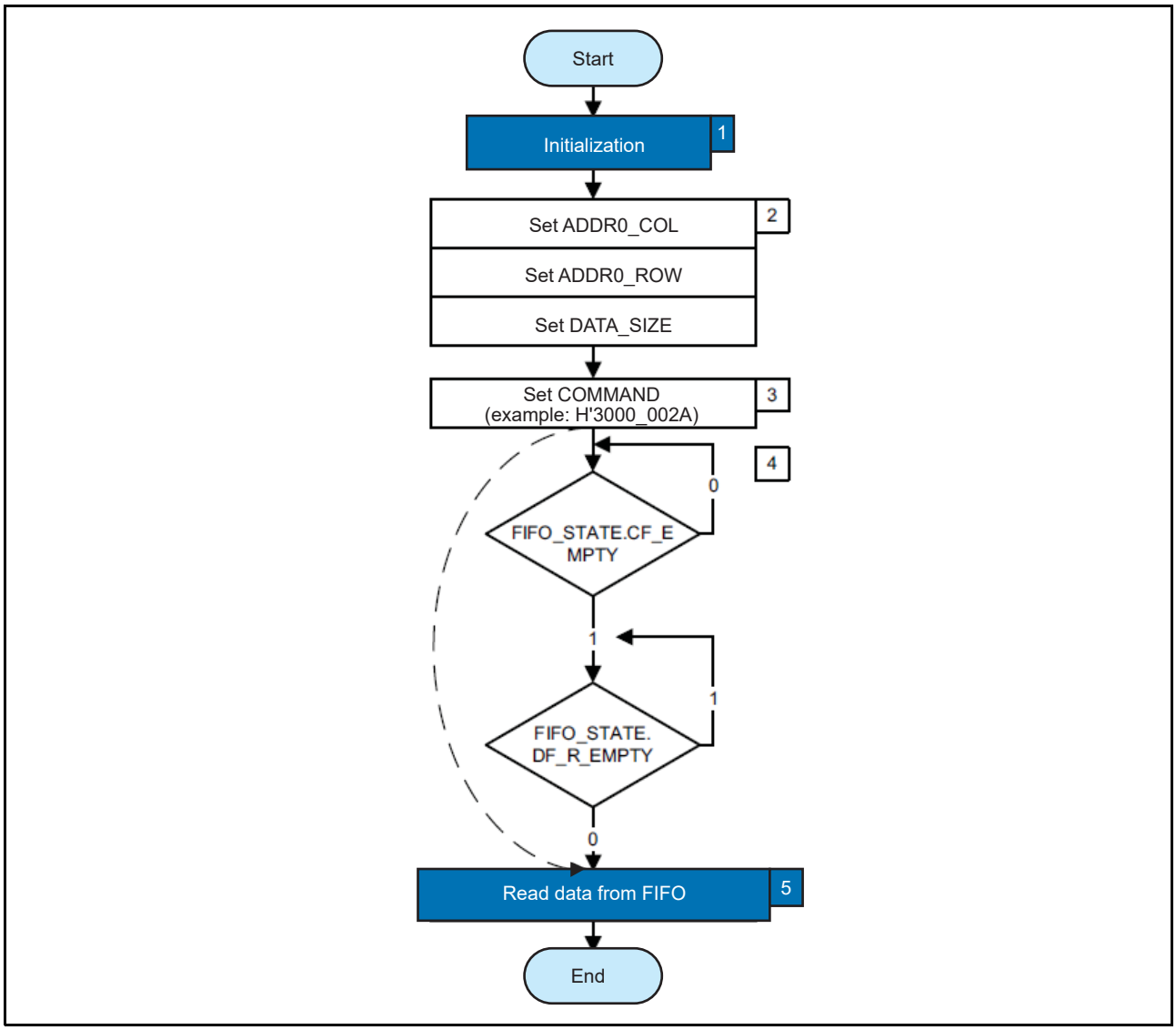


Figure 31.34 Read data from NAND Flash memory via Slave Interface

(3) Send Data to NAND Flash via Master Interface (using DMA)

1. The NAND Flash controller must be correctly configured before sending data to the NAND Flash memory device. The setup process is described in detail in section 31.3.7, Setup and Configuration and section 31.2, Register Description.
2. Set the INT_EN bit in the CONTROL register to enable global interrupts.
3. Select active interrupt in the INT_MASK register and write H'0000 0000 into the INT_STATUS register to clear all interrupts.
4. Select the DMA work mode to configure the DMA module correctly:
   In registers managed mode, the address of the data is written in the system memory (DMA_ADDR_L registers). Write the number of the transferred data into the DMA_CNT register. Set bit DMA_START to start DMA when the command sequence is sent to the NAND Flash memory. Bits ERR_FLAG and DMA_READY are read-only. The ERR_FLAG bit indicates an error on the internal bus while the DMA is transferring data; the DMA_READY bit indicates that the DMA is ready (transfer is completed).
5. Write the address of the data in the NAND Flash memory device into the address register 0.
6. To use the simplest program command, write H'0010 804C to the COMMAND register (PROGRAM PAGE command, FIFO selected, DMA selected).

When the memory is ready for further work, the appropriate MEM0_ST bit is set and interrupt is active. The recommended interrupt is MEM0_RDY_INT.

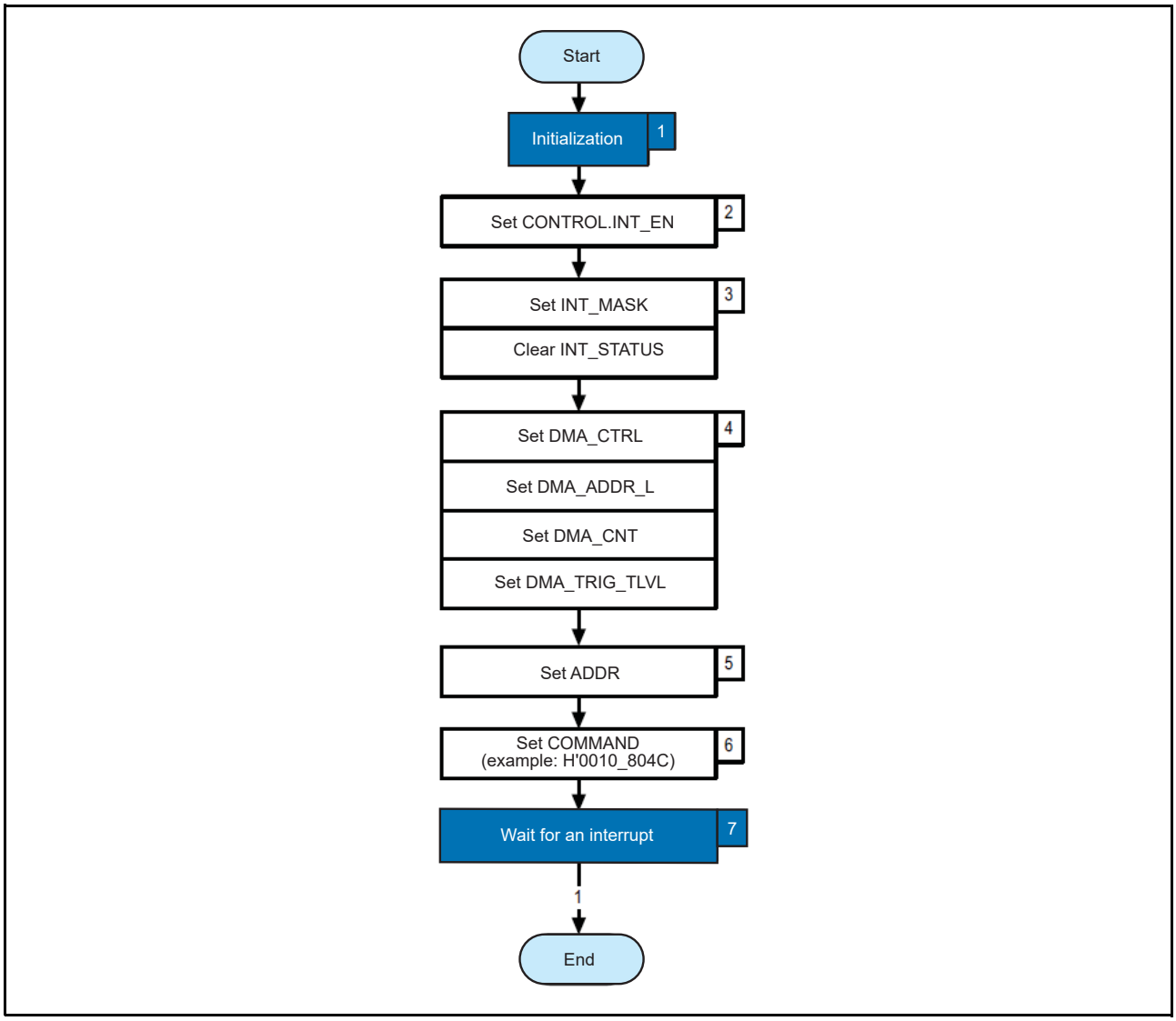


Figure 31.35 Send Data to NAND Flash Using DMA, Interrupt Enable

(4) Fast Writing and Reading of Several Pages from the Memory using DMA

1. The NAND Flash controller must be correctly configured before sending data to the NAND Flash memory device. The setup process is described in detail in section 31.3.7, Setup and Configuration and section 31.2, Register Description.
2. Set the INT_EN bit in the CONTROL register to enable global interrupts.
3. Select the active interrupt (CMD_END_INT_EN) in the INT_MASK register and write H'0000 0000 into the INT_STATUS register to clear all interrupts.
4. In Scatter-Gather mode, it is necessary to write the descriptors into the system memory. Select the DMA work mode to correctly configure the DMA module. In Scatter-Gather mode, it is not necessary to configure the DMA_CNT register. For more details, see section 31.4.2 (2) (b), Scatter-Gather Mode.
5. Set DMA_START bit to start the DMA when the command sequence is sent to the NAND Flash memory. Bits ERR_FLAG and DMA_READY are read-only. The former indicates the error on the internal bus while DMA is transferring data, the latter indicates when the DMA is ready (transfer is completed).
6. Set the ADDR0_INCR bit to auto increment row address 0 register after each command. Write address of the data in the NAND Flash memory device into the 0 address register.
7. Write H'0000 0000 into the INT_STATUS register to clear all interrupts.
8. Write address of the first descriptor into the DMA_ADDR_L registers.
9. Write PROGRAM PAGE CACHE command to the NAND Flash memory device by writing H'0015 804C into the COMMAND register (DMA selected, FIFO selected). When the controller is ready for further work, the appropriate CMD_END_INT_FL bit is set and the interrupt is activated.
10. When the number of the pages to be transferred does not equal one, go to step 7.
11. Write H'0000 0000 into the INT_STATUS register to clear all interrupts.
    Write address of the first descriptor into the DMA_ADDR_L register.
    The last command in the sequence of sending data is the PROGRAM PAGE (write H'0010804C) into the COMMAND register (DMA selected, FIFO selected).
12. Wait for interrupts and write H'0000 0000 into the INT_STATUS register to clear all interrupts.
13. Write the new descriptors to the system memory.
14. If DMA should work in the same work mode and burst type do not modify DMA_BURST and DMA_MODE bits. Set DMA_START bit to start DMA when the command sequence is sent to the NAND Flash memory.
15. Write the READ PAGE command into the NAND Flash memory device by writing H'3000 0069 into the COMMAND register (FIFO selected, DMA selected).
16. Write H'0000 0000 into the INT_STATUS register to clear all interrupts.
17. Write the address of the first descriptor to the DMA_ADDR_L registers.
18. Write READ PAGE CACHE SEQUENTIAL command to the NAND Flash memory device by writing H'0000 316B into the COMMAND register (FIFO selected, DMA selected).
19. When the controller is ready for further work, the appropriate CMD_END_INT_FL bit is set and the interrupt is activated. When the number of the pages to be transferred does not equal zero, go to step 16.
20. Write H'0000 0000 into the INT_STATUS register to clear all interrupts. Write the address of the first descriptor to the DMA_ADDR_L register. The last command in the sequence of reading data is READ PAGE CACHE LAST (write H'0000 3F6B into the COMMAND register).

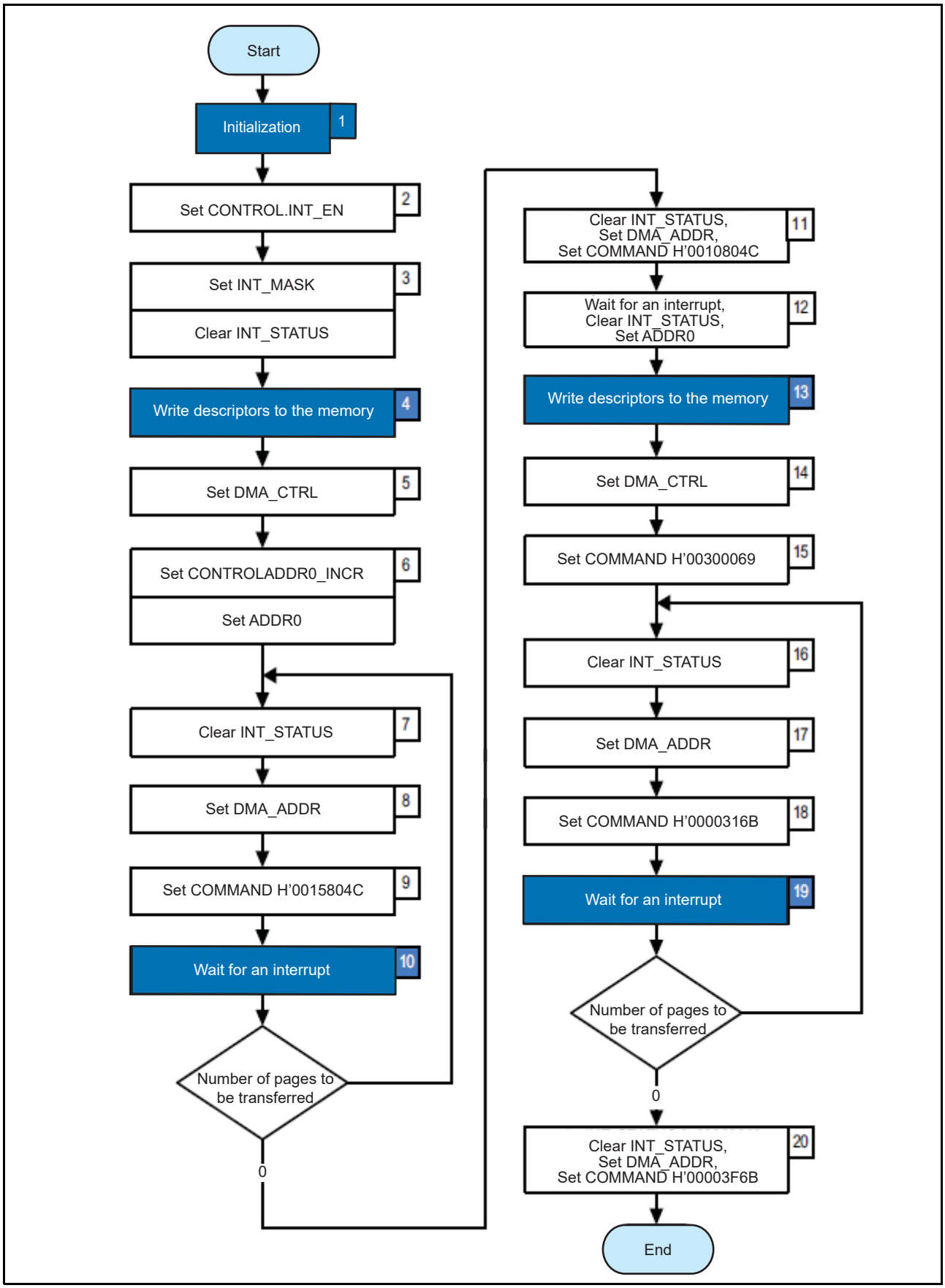


Figure 31.36 Fast Writing and Reading of Several Pages from the Memory Using DMA

(5) Writing to Partial Pages

The following procedure need to be used to write to partial pages with ECC engine enabled:

1. The NAND Flash controller must be correctly configured before sending data to the NAND Flash memory device. The setup process is described in detail in section 31.3.7, Setup and Configuration and section 31.2, Register Description.
2. Write the address of the data in NAND Flash memory into the address registers 0 (ADDR0_COL and ADDR0_ROW). You need to write the partial page offset into the ADDR0_COL register. Write the number of data that you want to write (DATA_SIZE register) – in this case you need to set the partial sector size. Write the offset value of the ECC data into the ECC_OFFSET register.
3. To use the simplest program command, write H'0010 800C to the COMMAND register (PROGRAM PAGE command, FIFO selected, slave interface selected).
4. Write data to the FIFO using the FIFO_DATA register. Data is sent to the NAND Flash memory device.

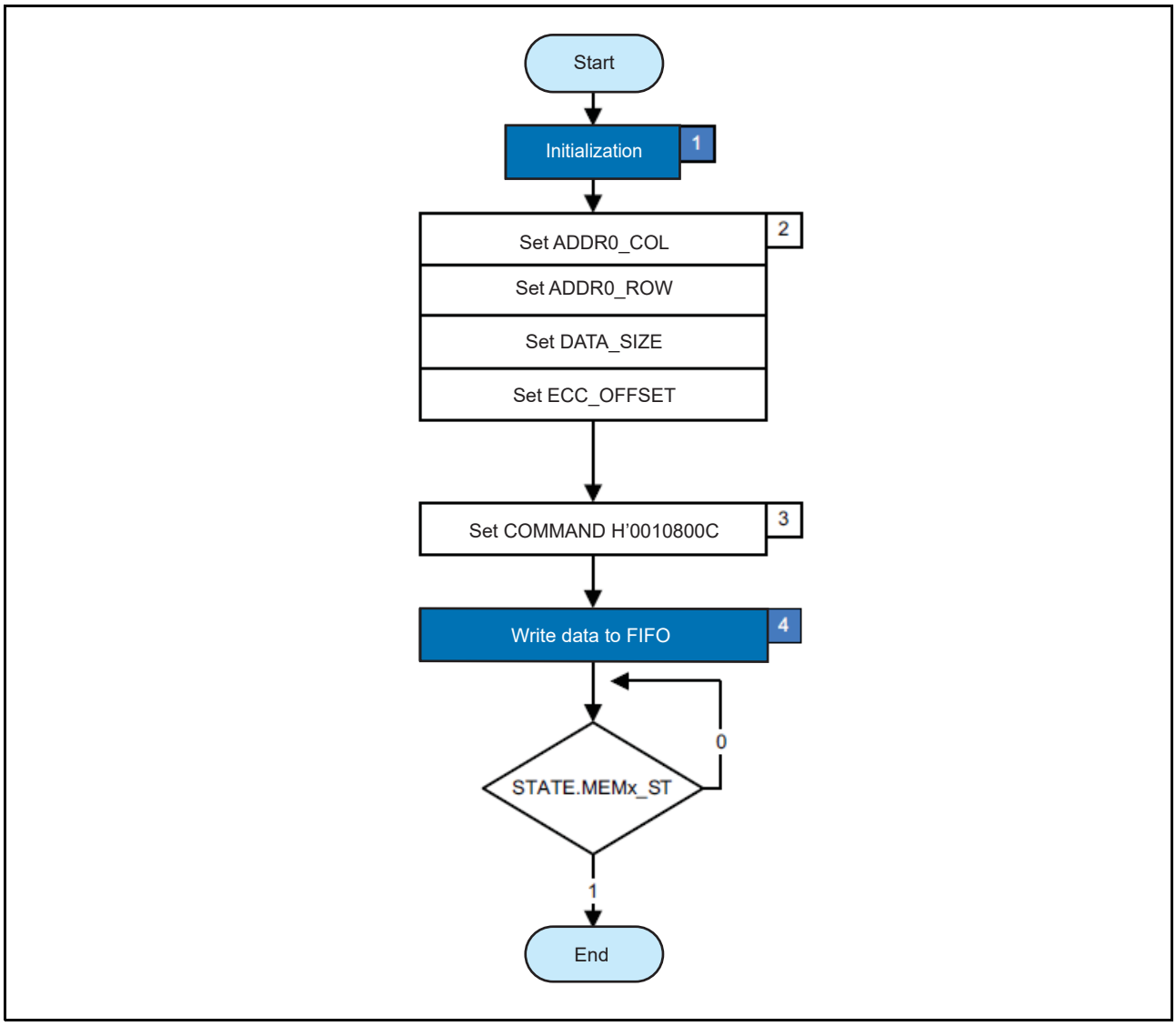


Figure 31.37 Writing to Partial Pages

(6) Reading Partial Pages

1. The NAND Flash controller must be correctly configured before sending data to the NAND Flash memory device. The setup process is described in detail in section 31.3.7, Setup and Configuration and section 31.2, Register Description.
2. Write the address of the data in NAND Flash memory into the address registers 0 (ADDR0_COL and ADDR0_ROW). You need to write the partial page offset into the ADDR0_COL register. Write the number of data that you want to read (DATA_SIZE register) – in this case you need to set the partial page size.
3. To use the simplest read command, write H'3000 002A to the COMMAND register (READ PAGE command, FIFO selected, slave interface selected).
4. The application program can read data immediately after sending the command but in this case reading will be preceded by a very few WAIT replies from the NAND flash controller. If this situation shall be avoided the application program must read FIFO_STATE register and wait when CF_EMPTY bit is set. After that wait for the DF_R_EMPTY bit in the FIFO_STATE register to be clear.
5. Read data from the FIFO using the FIFO_DATA register.

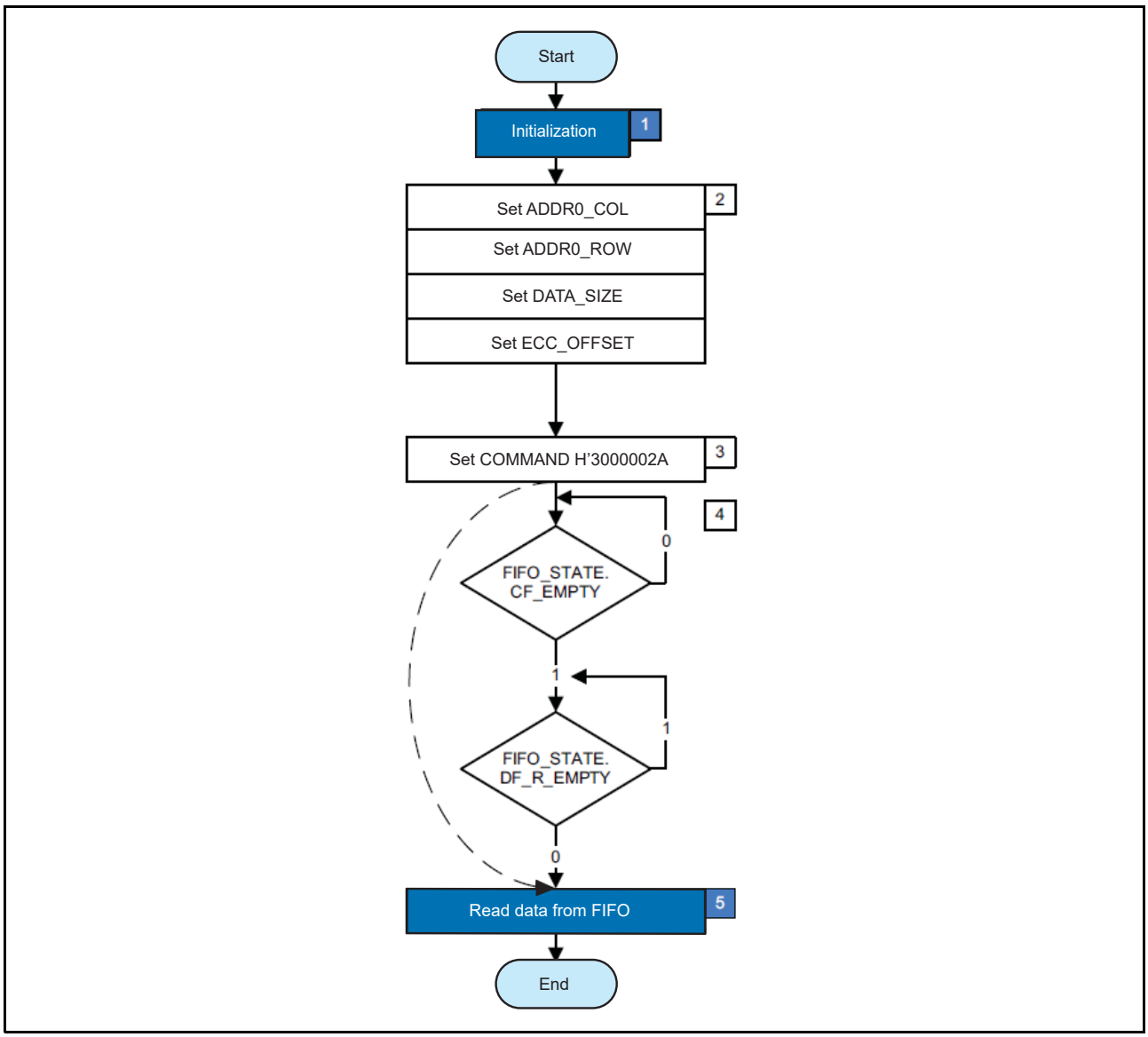


Figure 31.38 Reading Partial Pages

31.4 Functional Details

31.4.1 Block Diagram

The figure below shows the NAND Flash controller block diagram.

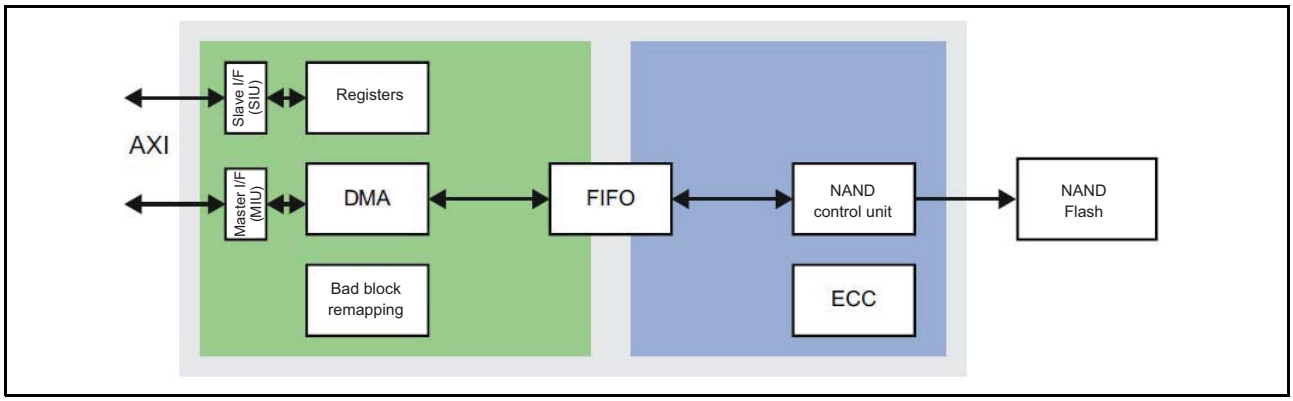


Figure 31.39 Block Diagram

The main blocks and their functions are as follows.

- FIFO provides the FIFO queue interface to other controller modules. Since the FIFO interface keeps control of the read and write pointers within the same module, one module can read and another write at the same time. One side of the queue is connected to the SIU or DMA module depending on the controller software configuration, and the other side of the queue is connected to the NAND control unit.
- DMA is responsible for the fast transfer of data between the external memory location and the controller. The unit can works in two modes: the Scatter-Gather mode and the internally triggered registers managed mode. DMA functions as the master.
- NAND control unit generates NAND Flash device access sequences.
- ECC is an error correction code calculator. A correction word is calculated for each 256B, 512B or 1024B ECC block of the NAND Flash memory page. During the read operation, this block can automatically correct bad bits without any interaction with the CPU. It has a status register, the bits of which signal errors occurring during a read, and then it informs when errors are corrected.
- Bad block remapping remaps blocks excluded from the poll of the usable blocks into the blocks that will be its substitute. Whole process is transparent for the software.

31.4.2 DMA

(1) Overview

The DMA in this module has the following properties:

- Two work modes:
  – The registers managed mode
  – The Scatter-Gather mode
- The DMA module preserves the configuration registers. These are copied into the DMA working registers.
- DMA trigger level enables DMA to send packet data.

(2) DMA Description

DMA supports two modes:

- The registers managed mode – the DMA does one transfer depending on the contents of DMA_ADDR, DMA_CNT, and DMA_CTRL registers.
- The Scatter-Gather mode – new DMA transfer algorithm.

The DMA mode for the current transfer is selected via the DMA_CTRL.DMA_MODE bit and cannot be changed during each consecutive data transfer. The application program can change the DMA mode when the transfer is completed and the DMA_CTRL.DMA_READY bit is set.

In order to begin the transfer, the DMA should be in an IDLE state (DMA_CTRL.DMA_READY = 1).

Setting the DMA_CTRL.DMA_START bit triggers the DMA after the commands are sent to the NAND Flash memory (writing CONTROL register). When the DMA_START bit is clear, sending commands to the NAND Flash memory will not trigger the DMA. This allows the DMA to be programmed to transfer more data than the page size of the NAND Flash memory. (example: the DMA is configured to 32 kB data transfer, the size of the page in NAND Flash memory is 4 kB. In order to send 32 kB of data into the NAND Flash memory, it is necessary to set the DMA_START bit in the register DMA_CTRL, then it is necessary to send the PROGRAM PAGE command to the NAND Flash memory, clear the DMA_START bit in the DMA_CTRL register, and send seven commands to the NAND Flash memory).
In this example, the DMA is set only once at the beginning.

The DMA trigger level enables the DMA to send packet data. It indicates when the DMA has to start transferring data. The application program must set trigger level depending on arbitration type, system and NAND Flash memory performance.
For example:

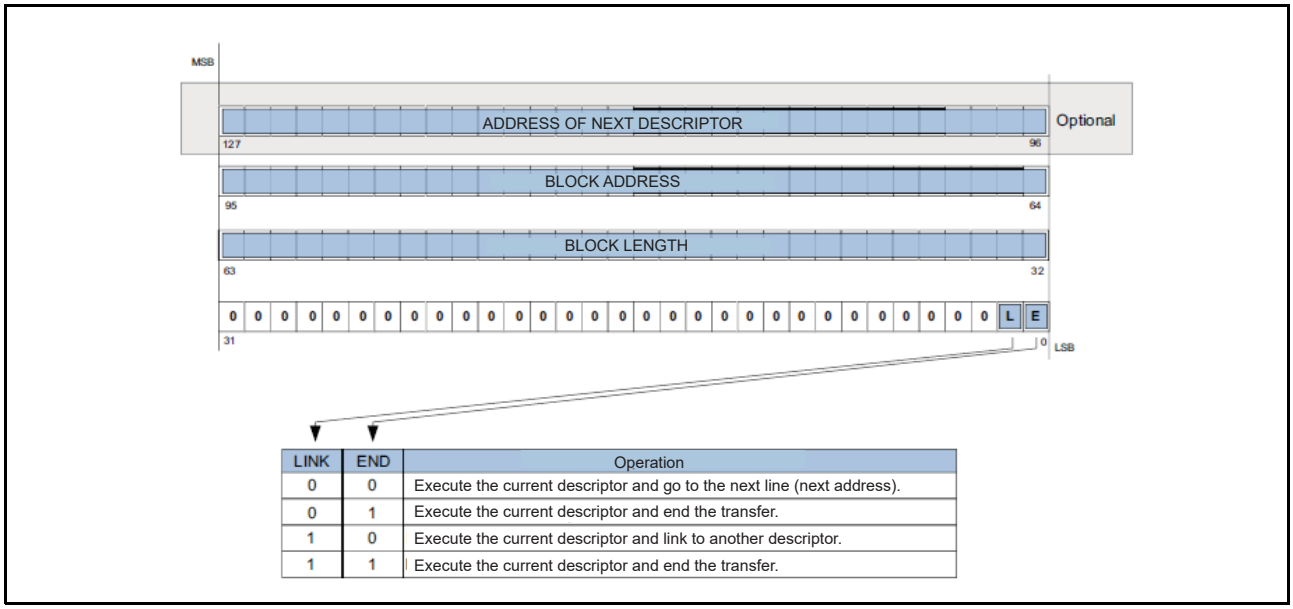
1. If the DMA trigger level equals 0, the DMA does not wait, but transfers data immediately because the trigger level is always exceeded.
2. For read from the NAND Flash memory: If the DMA trigger level equals H’40 (256 bytes), the DMA waits until data level reaches the previously programmed trigger level (256 bytes). Then the DMA will transfer data until FIFO gets empty status. Next the wait cycle for required data level is repeated as long as remain data size allow to reach programmed trigger level. The trigger level value should be chosen to ensure that only the continuous data block will be transferred and the FIFO will not be overflowed.
3. For write to NAND Flash memory: If the DMA trigger level equals H’40 (256 bytes), the DMA writes data to the FIFO until the FIFO is full. Then, the DMA waits until 256 bytes are written into the NAND Flash memory. Next, again the DMA transfers data until FIFO is full. The trigger level value should be chosen to ensure that only the continuous data block will be transferred.

(a) Registers Managed Mode

When the DMA works in this mode, it transfers only one continuous block (DMA_CNT is block length) to or from the system memory at address DMA_ADDR. Block can be multiple of DATA_SIZE value in the DATA_SIZE register. All the registers which modify this transfer are described in section 31.2.18, DMA base address register (DMA_ADDR_L), section 31.2.19, DMA counter initial value register (DMA_CNT) and section 31.2.20, DMA control register (DMA_CTRL).
The DMA_BURST bit in register DMA_CTRL defines the main transfer type used by the DMA to precede the requested transfer.

(b) Scatter-Gather Mode

The scatter-gather DMA controller implements high-speed DMA transfer from non-contiguous memory locations to a continuous address space, and vice versa.
The DMA in Scatter-Gather mode uses the Descriptors List to describe data transfers. The NAND Flash registers only points to the base address of the Descriptors List. The base address for this list is set in the DMA_ADDR register whether it is a read or write transfer. The base address, sizes of the data blocks, and flags are defined inside the descriptors.
The DMA_BURST bit in register DMA_CTRL defines the main transfer type used by the DMA.
When in Scatter-Gather mode, the DMA transfers data from data blocks. Each descriptor can define only one data block. The DMA adopts the scatter-gather DMA algorithm so that higher data transfer speed is available. The application program can program a list of data transfers between the system memory and NAND Flash to the Descriptor Table before executing Scatter-Gather mode.



LINK	END	Operation
0	0	Execute the current descriptor and go to the next line (next address).
0	1	Execute the current descriptor and end the transfer.
1	0	Execute the current descriptor and link to another descriptor.
1	1	Execute the current descriptor and end the transfer.

Figure 31.40 Descriptor Fields for Scatter-Gather DMA

The table below defines the Scatter-Gather DMA descriptor fields and their functions:

Table 31.47 Descriptor Fields for Scatter-Gather DMA

Bit	Symbol	Description		
[127:96]	ADDRESS	Next descriptor address when LINK = 1 and END = 0		
[95:64]	ADDRESS	Data block address		
[63:32]	LENGTH	Data page length (H'0000 0004 to H'FFFF FFFD) Specify a multiple of four as the number of bytes.		
[31:2]	—	Reserved		
[1]	LINK			
		LINK	END	Description
		0	0	Execute the current descriptor and go to the next descriptor on the list.
[0]	END	0	1	Execute the current descriptor and end the transfer.
		1	0	Execute the current descriptor and link to another descriptor.
		1	1	Execute the current descriptor and end the transfer.

Table 31.48 DMA Data Counter Length

LENGTH Value	Counter Length
H’0000 0000	Setting prohibited
H’0000 0001	Setting prohibited
H’0000 0002	Setting prohibited
H’0000 0003	Setting prohibited
H’0000 0004	4 bytes
H’0000 0005	Setting prohibited
H’0000 0006	Setting prohibited
H’0000 0007	Setting prohibited
H’0000 0008	8 bytes
….	….

The Descriptor Table is created in system memory by the application program. Each descriptor line (one executable unit) consists of the address, length and flags. The flags specify operation of the descriptor line. The size of each descriptor field is 16 bytes when LINK is 1 and END is 0, while the size is 12 bytes for other combinations.

The figure below shows the example of Scatter-Gather programming:

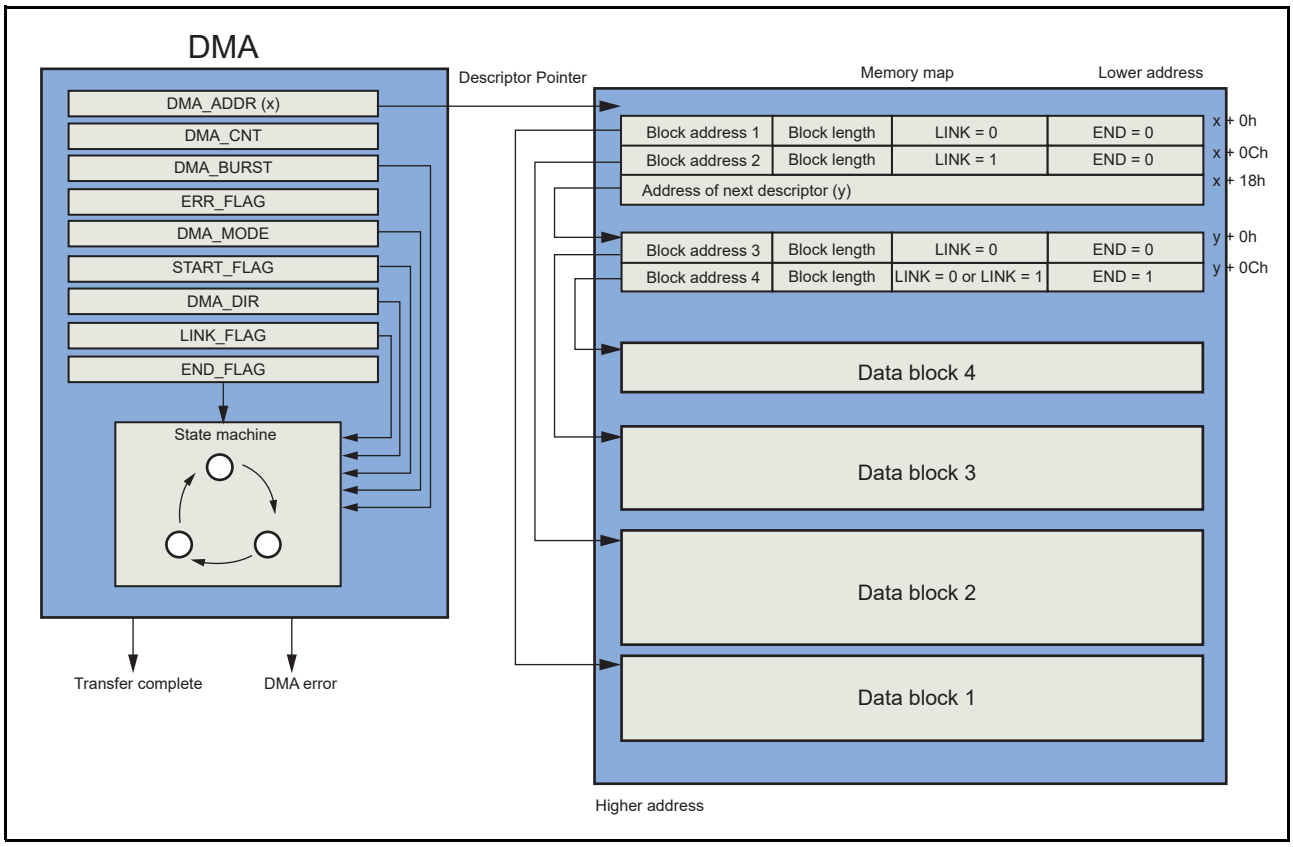


Figure 31.41 Example of Scatter-Gather DMA Data Transfer

31.4.3 ECC

(1) Overview

The built-in ECC of this module is based on one of the BCH algorithms and allows correction of multiple bit errors.
The ECC has the following features:

- The encoder and decoder work on the 256, 512, 1024 bytes data blocks.
- Programmable correction capability: 2, 4, 8, 16, 24 or 32 errors.
- The corrected data words are aligned to the 32 bits.
- The correction words are aligned to the 32 bits.
- Correction words are placed after the data.

(2) Block Diagram

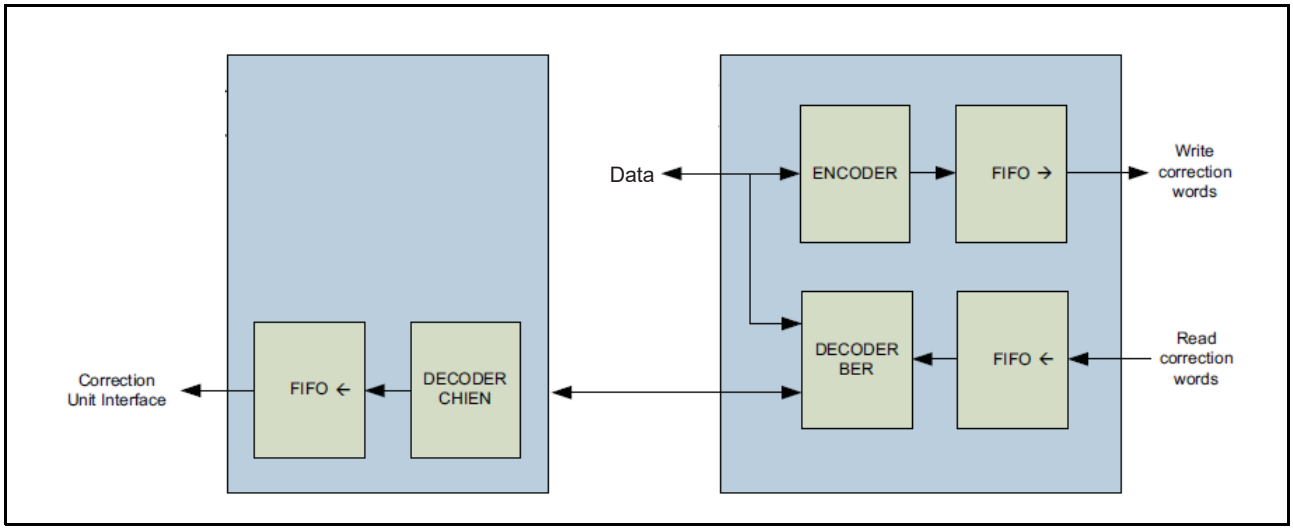


Figure 31.42 Block Diagram of ECC

31.4.4 BCH Algorithm Implementation

- Data block length: 256 (small block memory), 512, or 1024 bytes
- Programmable memory page length, multiple of the block length by any power of 2 (optional).
- Programmable correction capability: 2, 4, 8, 16, 24 or 32 errors.
- Separate encoder and decoder modules.
- Calculation of correction data performed during write to memory.
- Error detection performed during read data from memory.
- Internal pipeline allows the correction of errors in one data block simultaneously with detection of errors in the following data block.

Table 31.49 Size of Correction Bytes

Correction Capability	ECC Block Size	Size of Correction Bytes per one ECC Block
2 bits	256/512/1024 bytes	4 bytes
4 bits	256/512/1024 bytes	7 bytes
8 bits	256/512/1024 bytes	14 bytes
16 bits	256/512/1024 bytes	28 bytes
24 bits	256/512/1024 bytes	42 bytes
32 bits	256/512/1024 bytes	56 bytes

32. USB 2.0 Host Module

32.1 Overview

32.1.1 Overview

This LSI has two USB 2.0 host/function modules. For each module, you can switch between the host mode and the peripheral mode by specifying the UCOM register setting. This section describes the circuits that are common to both modes, and the host controller itself.

32.1.2 Features

This module has the following features:

Function	Description
Host function	• Supporting high-speed (480 Mbps), full-speed (12 Mbps), and low-speed (1.5 Mbps) transfers • Compliant with Open Host Controller Interface (OHCI) Specification for USB Revision 1.0a • Compliant with Enhanced Host Controller Interface (EHCI) Specification for USB Revision 1.1*1
Other functions	• Battery charging (compliant with Battery Charging Specification Revision 1.2)*2 • Dual-role-device function (static switching between the USB host and USB peripheral functions)

Note 1. Some functions (specifications) are not supported.
Note 2. The setting for battery charging is handled by the host controller even if the function controller is selected.

32.1.2.1 EHCI v1.1 functions

Conventional host controller modules comply with EHCI Specification Revision.1.0. Meanwhile, this module supports the additional functions listed below to comply with EHCI v1.1 Addendum.
Note, however, that this module supports only the functions marked with a circle "○" in the table below.

Function Name	Support
Per-Port Change Events	○
Shorter Periodic Frame List	○
Hardware Prefetching	—
Link Power Management (LPM)	○

Of the three features to support, the registers associated with “Per-Port Change Events” and “Shorter Periodic Frame List” are listed below. For more information on each register, refer to the appropriate register specification of the section 32.2, Register Descriptions.
With regard to “Link Power Management (LPM)”, it is described in section 32.1.2.2, Link Power Management (LPM) function.

-Per-Port Change Events

Relevant register	Relevant bit		Attribute	Function
HCCPARAMS (offset: 108h)	bit 18	Per-Port Change Event Capability	R	This bit indicates to support this function. This bit is fixed to 1b.
USBCMD (offset: 120h)	bit 15	Per-Port Change Events Enable	R/W	Per-Port event notification setting. Writing 1b to this bit enables the Per-Port event notification function.
USBSTS (offset: 124h)	bit 16	Port-1 Change Detect	R/W(1)	If a port-1 change event is detected, 1b is written to this bit.
USBINTR (offset: 128h)	bit 16	Port-1 Change Event Enable	R/W	Enable/Disable setting of the above “port-1 Change Detect” field. To reflect the port event on the above field for the port, set corresponding bit to 1b.

-Shorter Periodic Frame List

Relevant register	Relevant bit		Attribute	Function
HCCPARAMS (offset: 108h)	bit 19	32-Frame Periodic List Capability	R	This bit indicates to support this function. This bit is fixed to 1b.
USBCMD (offset: 120h)	bit[3:2]	Frame List Size	R/W	This bit determines the Frame List Size. To determine 32 frames, set this field 11b.

32.1.2.2 Link Power Management (LPM) function

This module supports the power management function conforming to “USB 2.0 Link Power Management Addendum to the Universal Serial Bus v2.0 Specification” (abbreviated as LPM).
When the peripheral device which supports LPM function is connected to this module, the device can be moved to suspend state faster than conventional by using LPM function.
*In using LPM function: 8 to 10 µs

The relevant registers are listed in the following. Refer to the appropriate register specification of each register detail in section 32.2, Register Descriptions.

Relevant register	Relevant bit		Attribute	Function
HCCPARAMS (offset: 108h)	bit 17	Link Power Management Capability	R	This bit indicates to support this function. This bit is fixed to 1b.
USBCMD (offset: 120h)	bit[27:24]	Host-Initiated Resume Duration	R/W	The minimum duration time of K-state drive in resume from LPM state.
PORTSC1 (offset: 164h)	bit[31:25]	Device Address [7:0]	R/W	The device address of the destination of LPM Token. In using LPM function, this setting is needed. Set the address of the device connected to the corresponding port before sending LPM Token.
	bit[24:23]	Suspend Status [1:0]	R	The device response to LPM Token.
	bit 9	Suspend using L1	R/W	In using LPM function, set this bit 1b.
	bit 7	Suspend	R/W	When this bit is set to 1b this module starts LPM Token transaction, in condition of “Suspend using L1 bit = 1b” and “Device address field = 000h”.
PORT_LPM_CTR1 (offset: 320h)	bit[7:4]	NYET_RETRY_CNT_P1[3:0]	R/W	The number of retry for NYET response of device in LPM transaction.
	bit 3	REMOTEWAKE_EN_P1	R/W	Setting of LPM RemoteWakeup permission. 0b: permitted (default) 1b: not permitted
	bit 2	SLEEP_INT_EN_P1	R/W	In LPM transaction, setting to generate an interrupt or not when received a response other than ACK. 0b: not to generate an interrupt (default) 1b: to generate an interrupt
	bit 1	RETRY_ENABLE_NYET_P1	R/W	In LPM transaction, setting to the host behavior when received NYET response from the device. 0b: not to Retry (default) 1b: to Retry
	bit 0	HIRD_SEL_P1	R/W	Setting the duration time of the K-state drive, in resume of LPM state.

32.1.2.3 Dual Role Device function

You can switch the role between the host function and the peripheral function via this module register (UCOM).
For details about the UCOM register, see section 32.2.4.5, UCOM Register.

32.1.2.4 Battery-charging function

The Control and monitoring of the Battery Charging I/F of the USBPHY is set in the UCOM register of this module.
The detail of the UCOM register, refer to section 32.2.4.5, UCOM Register.

32.1.2.5 Suspend extension function

This module implements the following two suspend extension functions to reduce power consumption by stopping PLL of USBPHY.
The detail of the register, refer to section 32.2.4.4, (6), Suspend Control Register (offset: 308h).

[Relevant register]
Suspend Control Register (offset: 308h)

[Functional specification]

Function		Relevant bit	
[1]	Function to assert USBPHY SUSPENDM by asserting the Suspend bit in the OHCI/EHCI Operational Register	bit 31	SUSPENDM_ENABLE
[2]	Function to forcibly assert USBPHY SUSPENDM	bit 0	GLOBAL_SUSPENDM_P1

32.1.2.6 Usage Notes

(1) In the case of disconnection immediately before a frame boundary from a device connected at full speed or low speed, subsequently recognizing the connection to the given device may not be possible.
(2) When an instruction to suspend the OHCI is executed after disconnection from a device connected at full speed or low speed, subsequently recognizing the connection to the given device may not be possible.

When a device is disconnected or an instruction to suspend the OHCI is executed after device disconnection, follow the procedure described in section 32.11.1, Actions after Device Disconnection.

32.1.3 Support of USB-Related Specifications

o: support
×: no support

USB-Related Specification or Function			Support
Host function	High Speed	Bulk IN/OUT transfer	○
		Control IN/OUT transfer	○
		Isochronous IN/OUT transfer	○
		Isochronous Highband transfer	○
		Interrupt IN/OUT transfer	○
	Full Speed	Bulk IN/OUT transfer	○
		Control IN/OUT transfer	○
		Isochronous IN/OUT transfer	○
		Interrupt IN/OUT transfer	○
	Low Speed	Control IN/OUT transfer	○
		Interrupt IN/OUTtransfer	○
	No. of hub connection stages	HS: 5 stages	○
		FS: 5 stages	○
	Support of EHCI V1.1	Hardware Prefetching	×
		Link Power Management	○
		Per-Port Change Events	○
		Shorter Periodic Frame List	○
Battery-charging function (hereafter called the "BC function")			○
Dual role device * Function to statically switch between the host and peripheral modes			○

32.1.4 Input/Output Pins

The following table describes the pin alignment.

Channel	Name	Pin	I/O	Function
0	USB D+ Data	DP0	I/O	USB built-in transceiver D+ I/O Connect this pin to the USB bus D+ pin.
	USB D- Data	DM0	I/O	USB built-in transceiver D- I/O Connect this pin to the USB bus D- pin.
	VBUS Input	VBUSIN0	Input	USB cable connection monitor pin Before connecting this pin, lower the voltage of VBUS of the USB bus to 3.3 V. This pin can be used to detect connection and disconnection of VBUS.
	VBUS Output	VBUSEN0	Output	VBUS power enable pin
	Overcurrent Input	OVRCUR0	Input	Overcurrent pin
	OTG Power IC Control	OTG_EXICEN0	Output	OTG power IC control pin
	OTG Power IC ID	OTG_ID0	Input	OTG power IC ID pin
	CC Input	CC1_Rd0, CC1_Ra0, CC2_Rd0, CC2_Ra0	Input	CC pin monitor for the Type-C pin Before connecting this pin, detect the status of the resistance value of the CC pin.
	Reference Input	RREF0	Input	Pin for connecting the reference resistance Connect this pin to USBVss via the resistance of 2.2 kΩ ± 1%.
	Transceiver Section Analog Pin Power	USBAPVcc0	Input	Power for the pin
	Transceiver Section Digital Pin Power	USBDPVcc0	Input	Power for the pin
1	USB D+ Data	DP1	I/O	USB built-in transceiver D+ I/O Connect this pin to the USB bus D+ pin.
	USB D- Data	DM1	I/O	USB built-in transceiver D- I/O Connect this pin to the USB bus D- pin.
	VBUS Input	VBUSIN1	Input	USB cable connection monitor pin Before connecting this pin, lower the voltage of VBUS of the USB bus to 3.3 V. This pin can be used to detect connection and disconnection of VBUS.
	VBUS Output	VBUSEN1	Output	VBUS power enable pin
	Overcurrent Input	OVRCUR1	Input	Overcurrent pin
	OTG Power IC Control	OTG_EXICEN1	Output	OTG power IC control pin
	OTG Power IC ID	OTG_ID1	Input	OTG power IC ID pin
	CC Input	CC1_Rd1, CC1_Ra1, CC2_Rd1, CC2_Ra1	Input	CC pin monitor for the Type-C pin. Before connecting this pin, detect the status of the resistance value of the CC pin.
	Reference Input	RREF1	Input	Pin for connecting the reference resistance. Connect this pin to USBVss via the resistance of 2.2 kΩ ± 1%.
	Transceiver Section Analog Pin Power	USBAPVcc1	Input	Power for the pin
	Transceiver Section Digital Pin Power	USBDPVcc1	Input	Power for the pin
Common	Quartz Oscillator for USB/External Clock	USB_X1	Input	Connect this pin to the quartz oscillator for the USB 2.0 host/function module. USB_X1 pin can be also used for inputting the external clock.
		USB_X2	Output	
	Transceiver Section pin Grand	USBVss	Input	Grand for pins

32.2 Register Descriptions

32.2.1 Register Attributes

Table 32.1 Register Attributes

Register Attribute	Description
R/W	Register bits can be read and written.
R/W(1)	Register bits can be read. A clear bit may be set by writing “1”; writing 0 to R/W(1) bits has no effect.
R/W(0)	Register bits can be read. A clear bit may be set by writing “0”; writing 1 to R/W(0) bits has no effect.
R	Register bits can only be read.
W	Register bits can only be written.
Reserved	Reserved bits are Read Only field.

32.2.2 Base Address

Table 32.2 Base Addresses for Each Channel of the USB Host Module

Channel	Base Address
0	E821 8000h
1	E821 A000h

32.2.3 Register Overview

Address Offset	Register Name	Abbreviated Name	Remarks
000h	HcRevision	HcRevision	OHCI Operation Registers
004h	HcControl	HcControl	
008h	HcCommandStatus	HcCommandStatus	
00Ch	HcInterruptStatus	HcInterruptStatus	
010h	HcInterruptEnable	HcInterruptEnable	
014h	HcInterruptDisable	HcInterruptDisable	
018h	HcHCCA	HcHCCA	
01Ch	HcPeriodCurrentED	HcPeriodCurrentED	
020h	HcControlHeadED	HcControlHeadED	
024h	HcControlCurrentED	HcControlCurrentED	
028h	HcBulkHeadED	HcBulkHeadED	
02Ch	HcBulkCurrentED	HcBulkCurrentED	
030h	HcDoneHead	HcDoneHead	
034h	HcFmInterval	HcFmInterval	
038h	HcFmRemaining	HcFmRemaining	
03Ch	HcFmNumber	HcFmNumber	
040h	HcPeriodicStart	HcPeriodicStart	
044h	HcLSThreshold	HcLSThreshold	
048h	HcRhDescriptorA	HcRhDescriptorA	
04Ch	HcRhDescriptorB	HcRhDescriptorB	
050h	HcRhStatus	HcRhStatus	
054h	HcRhPortStatus1	HcRhPortStatus1	
100h	HCIVERSION / CAPLENGTH	CAPL_VERSION	EHCI Capability Registers
104h	HCSPARAMS	HCSPARAMS	
108h	HCCPARAMS	HCCPARAMS	
10Ch	HCSP_PORTROUTE	HCSP_PORTROUTE	
120h	USBCMD	USBCMD	EHCI Operation Registers
124h	USBSTS	USBSTS	
128h	USBINTR	USBINTR	
12Ch	FRINDEX	FRINDEX	
130h	CTRLDSSEGMENT	CTRLDSSEGMENT	
134h	PERIODICLISTBASE	PERIODICLISTBASE	
138h	ASYNCLISTADDR	ASYNCLISTADDR	
160h	CONFIGFLAG	CONFIGFLAG	
164h	PORTSC1	PORTSC1	
200h	INT_ENABLE	INT_ENABLE	AHB Registers
204h	INT_STATUS	INT_STATUS	
208h	AHB_BUS_CTR	AHB_BUS_CTR	
20Ch	USBCTR	USBCTR	

Address Offset	Register Name	Abbreviated Name	Remarks
304h	Register Enable/Clock Gating Control	REGEN_CG_CTRL	Core Defined Registers
308h	Suspend Control	SPD_CTRL	
30Ch	Suspend/Resume Timer Setting	SPD_RSM_TIMSET	
310h	Overcurrent Detection/Sleep Timer Setting	OC_SLP_TIMSET	
314h	SBRN/FLADJ/PORTWAKECAP	SBRN_FLADJ_PW	
320h	PORT_LPM_CTRL1	PORT_LPM_CTRL1	
360h	USB 2 Host Controller Extended Function 2	U2HC_EXT2	
800h	Common Control	COMMCTRL	OTG/BC Module Control Register
804h	OTG-BC Interrupt Status	OBINTSTA	OTG/BC Interrupt Status Register
808h	OTG-BC Interrupt Enable	OBINTEN	OTG/BC Interrupt Enable Register
80Ch	VBUS Control	VBCTRL	OTG VBUS Control Register
810h	Line Control Port 1	LINECTRL1	OTG USB Bus Control Register (Port 1)
820h	BC Control Port 1	BCCTRL1	Battery Charging Control Register (Port 1)
840h	CC STATUS	CC_STATUS	CC Terminal Control Register
844h	PHYCLK CTRL	PHYCLK_CTRL	USBPHY Supply Clock Control Register
848h	PHYIF CTRL	PHYIF_CTRL	Deep Standby Recovery Control Register

32.2.4 Description of Registers

32.2.4.1 OHCI Operational Register

(1) HcRevision Register (offset: 000h)

Abbreviated name of register: HcRevision

Address:	003h								002h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	001h								000h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Revision							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Table 32.3 HcRevision Register

Bit	Symbol	Description
31 to 8		Reserved The write value should always be 0.
7 to 0	Revision	This field indicates the version of HCI specifications implemented in this host controller module. Because this module conforms to OHCI standard 1.0a, 10h is indicated.

(2) HcControl Register (offset: 004h)

Abbreviated name of register: HcControl

Address:	007h								006h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	005h								004h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved						RWC	Reserved	HCFS		BLE	CLE	IE	PLE	CBSR	
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.4 HcControl Register

Bit	Symbol	Description
31 to 10		Reserved The write value should always be 0.
9	RWC (Remote Wakeup Connected)	This bit indicates whether the host controller supports remote wakeup signaling. Software should set this bit to 1b in the initialization sequence if it is required to support remote wakeup. Note that this bit can be initialized at hardware reset only. 0b: The remote wakeup is not supported. 1b: The remote wakeup is supported
8		Reserved The write value should always be 0.

Table 32.4 HcControl Register

Bit	Symbol	Description
7, 6	HCFS (Host Controller Functional State)	This field indicates the operating state of the host controller. 00b: USB Reset 01b: USB Resume 10b: USB Operational 11b: USB Suspend When the state is changed to USB Operational, the host controller module starts SOF transmission at 1 ms boundary. This field is basically controlled by software, but it can be controlled by the host controller in the state of USB Suspend only. If the host controller (in the state of USB Suspend) detects a Remote Wakeup signal from the connecting device, the state in this field is changed to USB Resume. Note that the reset value of this field differs between hardware reset and software reset. Hardware reset: 00b (USB Reset) Software reset: 11b (USB Suspend)
5	BLE (BulkList Enable)	This bit sets whether the bulk list processing is performed for the next frame. The setting value of this bit is enabled from the next frame. When you correct the bulk list, this bit must be 0b. 0b: The processing of the Bulk list is disabled. 1b: The processing of the Bulk list is enabled
4	CLE (ControlList Enable)	This bit sets whether the control list processing is performed for the next frame. The setting value of this bit is enabled from the next frame. When you correct the control list, this bit must be 0b. 0b: The processing of the Control list is disabled. 1b: The processing of the Control list is enabled.
3	IE (Isochronous Enable)	This bit sets whether the isochronous ED processing is performed. The setting value of this bit is enabled from the next frame. If the host controller module detects isochronous ED (F = 1) during the periodic list processing, it checks the bit and determines whether to perform isochronous ED processing. 1b: The processing of the isochronous ED is continued. 0b: The periodic list processing is stopped, and the bulk/control list processing is started. 0b: The processing of the isochronous ED is disabled. 1b: The processing of the isochronous ED is enabled.
2	PLE (Periodic List Enable)	This bit indicates whether the periodic list processing is performed for the next frame. The setting value of this bit is enabled from the next frame. The host controller module checks this bit before starting the periodic list processing. 0b: The processing of the periodic list is disabled. 1b: The processing of the periodic list is enabled.
1, 0	CBSR (Control Bulk Service Ratio)	This field defines the service ratio of the control transfer and bulk transfer. When the periodic list is processed, the service ratio defined in this field is used for transfer.

CBSR	No. of Control EDs Over Bulk EDs Served
00b	1 : 1
01b	2 : 1
10b	3 : 1
11b	4 : 1

(3) HcCommandStatus Register (offset: 008h)

Abbreviated name of register: HcCommandStatus

Address:	00Bh								00Ah							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved						SOC	
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	009h								008h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	W
Symbol:	Reserved								Reserved					BLF	CLF	HCR
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.5 HcCommandStatus Register

Bit	Symbol	Description
31 to 18		Reserved The write value should always be 0.
17, 16	SOC (Scheduling Overrun Count)	This field counts the number of scheduling overrun. This field is initialized to 00b, and is counted up every time scheduling overrun is detected. After the field is incremented to 11b, it returns to 00b. Even if the SO (Scheduling Overrun) bit in the HcInterrupt Status register is set, this field is counted up when scheduling overrun is detected.
15 to 3		Reserved The write value should always be 0.
2	BLF (BulkList Filled)	This bit indicates whether TD exists in the bulk list. To add TD to ED in the bulk list, set this bit to 1b. The host controller checks this bit when it starts processing of the head ED in the bulk list. If this bit is set to 0b, the host controller does not start the list processing. If this bit is set to 1b, the host controller starts processing of the bulk list, and sets this bit to 0b. When the host controller finds TD in the bulk list, it sets this bit to 1b again, and continues processing of the bulk list. When the host controller finishes the list processing, this bit is set to 0b. However, if TD is not found in the bulk list, or if this bit is not set to 1b, this bit remains to be 0b, and the list processing stops. To rebuild the list and start the list processing, before you set the BLE bit of the HcControl register and start the list processing, you need to set this bit. 0b: TD does not exist in the bulk list. 1b: TD exists in the bulk list.
1	CLF (ControlList Filled)	This bit indicates whether TD exists in the control list. To add TD to ED in the control list, set this bit to 1b. The host controller checks this bit when it starts processing of the head ED in the control list. If this bit is set to 0b, the host controller does not start the processing of the control list. If this bit is set to 1b, the host controller starts processing of the control list, and set this bit to 0b. When the host controller finds TD in the control list, it sets this be to 1b again, and continues the list processing. When the host controller finishes the list processing, this bit is set to 0b. However, if TD is not found in the control list, or if this bit is not set to 1b, this bit remains to be 0b, and the list processing stops. To rebuild the list and start the list processing, before you set the CLE bit of the HcControl register and start the list processing, you need to set this bit. 0b: TD does not exist in the control list. 1b: TD exists in the control list.
0	HCR (Host Controller Reset)	This bit is used to start OHCI software reset for the host controller. When this bit is set to 1b, the operating status of the host controller is changed to USB Suspend regardless of the functional state of the host controller. Also, the most OHCI Operational registers and OHCI control circuits are initialized. When the software reset finishes, the host controller clears this bit to 0b.

(4) HcInterruptStatus Register (offset: 00Ch)

Abbreviated name of register: HcInterruptStatus

Address:	00Fh								00Eh							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	00Dh								00Ch							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved								Reserved	RHSC	FNO	UE	RD	SF	WDH	SO
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.6 HcInterruptStatus Register

Bit	Symbol	Description
31 to 7		Reserved The write value should always be 0.
6	RHSC (Root Hub Status Change)	Interrupt bit that indicates that the state of the HcRhStatus register or HcRhPortStatus1 register has changed. When the Root Hub status is changed, the host controller sets this bit to 1b. Writing 1b to this bit clears the interrupt. 0b: The status of root hub has not changed. 1b: The status of root hub has changed.
5	FNO (Frame Number Overflow)	Interrupt bit that indicates that the MSB in the FrameNumber (FN) field of theHcFmNumber register has changed. When the MSB in the Frame Number field is changed from 0 to 1, or from 1 to 0, this bit is set after HccaFrameNumber is updated. Writing this bit to 1b clears the interrupt. 0b: The overflow of frame number has not occurred. 1b: The overflow of frame number has occurred.
4	UE (Unrecoverable Error)	Interrupt bit that indicates that a system error that is not related to the USB (for example, an error on the system bus) has been detected. Writing this bit to 1b clears the interrupt. 0b: The unrecoverable error has not occurred. 1b: The unrecoverable error has occurred.
3	RD (Resume Detected)	Interrupt bit that indicates that Resume has been detected. When the host controller detects the Resume signal (RemoteWakeup) from an USB device, it sets this bit to 1b. This bit is not set when the Resume signal is sent by setting the HCFS field to USB Resume. Writing this bit to 1b clears the interrupt. 0b: The host controller has not detected resume signaling (RemoteWakeup). 1b: The host controller has detected resume signaling (RemoteWakeup).
2	SF (Start of Frame)	Interrupt bit that indicates that Hcca Frame Number was updated when each frame started. The host controller sends an SOF packet and updates HccaFrameNumber at the same time, and sets this bit to 1b. Writing this bit to 1b clears the interrupt. 0b: The host controller has not started new frame. 1b: The host controller has started new frame.
1	WDH (Writeback DoneHead)	Interrupt bit that indicates that the host controller has written the contents of HcDoneHead to HccDoneHead. The host controller sets this bit to 1b immediately after it updates HccaDoneHead. Note that HccaDoneHead is not updated until this bit is cleared. Writing this bit to 1b clears the interrupt. This bit must be cleared only after the contents of HccaDoneHead are saved. 0b: The write back to HccaDoneHead has not occurred. 1b: The write back to HccaDoneHead has occurred.

Table 32.6 HcInterruptStatus Register

Bit	Symbol	Description
0	SO (Scheduling Overrun)	Interrupt bit that indicates that overrun of the USB schedule occurred. When USB scheduling overrun occurs, the host controller updates HccaFrameNumberl, and sets this bit to 1b. When this bit is set, the SchedulingOverrunCount field of the HcCommandStatus register is also incremented. Writing this bit to 1b clears the interrupt. 0b: The scheduling overrun has not occurred. 1b: The scheduling overrun has occurred.

(5) HcInterruptEnable Register (offset: 010h)

Abbreviated name of register: HcInterruptEnable

Address:	013h								012h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	MIE	OCE	Reserved						Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	011h								010h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved								Reserved	RHSCE	FNOE	UEE	RDE	SFE	WDHE	SOE
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.7 HcInterruptEnable Register

Bit	Symbol	Description
31	MIE (Master Interrupt Enable)	This bit sets whether each Interrupt Enable setting (which is set in HcInterruptEnable [30:0]) is enabled. If this bit is set to 0b, all OHCI interrupts are masked. To clear this bit, set the MID bit (bit 31) of the HcInterruptDisable register to 1b. 0b: All interrupts are disabled. 1b: Interrupts that are set to 1b are enabled.
30	OCE	OC (Ownership Change) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes OC (OwnershipChange). To clear this bit, set the OCD bit (bit 30) of the HcInterruptDisable register to 1b. 0b: OC (OwnershipChange) interrupt is disabled. 1b: OC (OwnershipChange) interrupt is enabled.
29 to 7		Reserved The write value should always be 0.
6	RHSCE	RHSC (RootHub Status Change) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes RHSC (RootHub Status Change). To clear this bit, set the RHSCD bit (bit 6) of the HcInterruptDisable register to 1b. 0b: RHSC (RootHub Status Change) interrupt is disabled. 1b: RHSC (RootHub Status Change) interrupt is enabled.
5	FNOE	FNO (Frame Number Overflow) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes FNO (Frame Number Overflow). To clear this bit, set the FNOD bit (bit 5) of the HcInterruptDisable register to 1b. 0b: FNO (Frame Number Overflow) interrupt is disabled. 1b: FNO (Frame Number Overflow) interrupt is enabled.
4	UEE	UE (Unrecoverable Error) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes UE (Unrecoverable Error). To clear this bit, set the UED bit (bit 4) of the HcInterruptDisable register to 1b. 0b: UE (Unrecoverable Error) interrupt is disabled. 1b: UE (Unrecoverable Error) interrupt is enabled.
3	RDE	RD (Resume Detect) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes RD (Resume Detect). To clear this bit, set the RDD bit (bit 3) of the HcInterruptDisable register to 1b. 0b: RD (Resume Detect) interrupt is disabled. 1b: RD (Resume Detect) interrupt is enabled.
2	SFE	SF (Start of Frame) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes SF (Start of Frame). To clear this bit, set the SFD bit (bit 2) of the HcInterruptDisable register to 1b. 0b: SF (Start of Frame) interrupt is disabled. 1b: SF (Start of Frame) interrupt is enabled.
1	WDHE	WDH (Writeback DoneHead) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes WDH (Writeback DoneHead). To clear this bit, set the WDHDD bit (bit 1) of the HcInterruptDisable register to 1b. 0b: WDH (Writeback DoneHead) interrupt is disabled. 1b: WDH (Writeback DoneHead) interrupt is enabled.

Table 32.7 HcInterruptEnable Register

Bit	Symbol	Description
0	SOE	SO (Scheduling Overrun) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes SO (Scheduling Overrun). To clear this bit, set the SOD bit (bit 0) of the HcInterruptDisable register to 1b. 0b: SO (Scheduling Overrun) interrupt is disabled. 1b: SO (Scheduling Overrun) interrupt is enabled.

(6) HcInterruptDisable Register (offset: 014h)

Abbreviated name of register: HcInterruptDisable

Address:	017h								016h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W(1)	R/W(1)	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	MID	OCD	Reserved						Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	015h								014h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)
Symbol:	Reserved								Reserved	RHSCD	FNOD	UED	RDD	SFD	WDHD	SOD
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.8 HcInterruptDisable Register

Bit	Symbol	Description
31	MID (Master Interrupt Disable)	This bit sets whether the Enable setting (for each interrupt) that is set by HcInterruptEnable[30:0] is disabled. If this bit is set to 1b, the MIE bit (bit 31) of the HcInterruptEnable register is cleared to 0b, and all OHCI interrupts are masked. Writing 0b to this bit is ignored.
30	OCD	OC (Ownership Change) Interrupt Disable bit. If this bit is set to 1b, the OCE bit (bit 30) of the HcInterruptEnable register is cleared to 0b, and OC (Ownership Change) is excluded from the interrupt source. Writing 0b to this bit is ignored.
29 to 7		Reserved The write value should always be 0.
6	RHSCD	RHSC (RootHub Status Change) Interrupt Disable bit. If this bit is set to 1b, the RHSCE bit (bit 6) of the HcInterruptEnable register is cleared to 0b, and RHSC (RootHub Status Change) is excluded from the interrupt source. Writing 0b to this bit is ignored.
5	FNOD	FNO (Frame Number Overflow) Interrupt Disable bit. If this bit is set to 1b, the FNOE bit (bit 5) of the HcInterruptEnable register is cleared to 0b, and FNO (Frame Number Overflow) is excluded from the interrupt source. Writing 0b to this bit is ignored.
4	UED	UE (Unrecoverable Error) Interrupt Disable bit. If this bit is set to 1b, the UEE bit (bit 4) of the HcInterruptEnable register is cleared to 0b, and UE (Unrecoverable Error) is excluded from the interrupt source. Writing 0b to this bit is ignored.
3	RDD	RD (Resume Detected) Interrupt Disable bit. If this bit is set to 1b, the RDE bit (bit 3) of the HcInterruptEnable register is cleared to 0b, and RD (Resume Detected) is excluded from the interrupt source. Writing 0b to this bit is ignored.
2	SFD	SF (Start of Frame) Interrupt Disable bit. If this bit is set to 1b, the SFE bit (bit 2) of the HcInterruptEnable register is cleared to 0b, and SF (Start of Frame) is excluded from the interrupt source. Writing 0b to this bit is ignored.
1	WDHD	WDH (Writeback DoneHead) Interrupt Disable bit. If this bit is set to 1b, the WDHE bit (bit 1) of the HcInterruptEnable register is cleared to 0b, and WDH (Writeback DoneHead) is excluded form the interrupt source. Writing 0b to this bit is ignored.
0	SOD	SO (Scheduling Overrun) Interrupt Disable bit. If this bit is set to 1b, the SOE bit (bit 1) of the HcInterruptEnable register is cleared to 0b, and SO (Scheduling Overrun) is excluded from the interrupt source. Writing 0b to this bit is ignored.

(7) HcHCCA Register (offset: 018h)

Abbreviated name of register: HcHCCA

Address:	01Bh								01Ah							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	HcHCCA[31:24]								HcHCCA[23:16]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	019h								018h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R
Symbol:	HcHCCA[15:8]								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.9 HcHCCA Register

Bit	Symbol	Description
31 to 8	HcHCCA	This field sets the base address (of the RAM) that is assigned as the Host Controller Communication Area. This field must be set at initialization. The host controller requests (as HCCA) 256-byte area from the base address specified in this field.
7 to 0		Reserved The write value should always be 0.

(8) HcPeriodicCurrentED Register (offset: 01Ch)

Abbreviated name of register: HcPeriodCurrentED

Address:	01Fh								01Eh							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	PECD[31:24]								PECD[23:16]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	01Dh								01Ch							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	PECD[15:8]								PECD[7:4]				Reserved			
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.10 HcPeriodCurrentED Register

Bit	Symbol	Description
31 to 4	PECD (Period Current ED)	This pointer indicates the physical address of ED in the periodic list that is currently processed. The host controller updates this pointer when the list processing of the periodic ED finishes.
3 to 0	Reserved	Reserved The write value should always be 0.

(9) HcControlHeadED Register (offset: 020h)

Abbreviated name of register: HcControlHeadED

Address:	023h								022h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	CHED[31:24]								CHED[23:16]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	021h								020h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R
Symbol:	CHED[15:8]								CHED[7:4]				Reserved			
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.11 HcControlHeadED Register

Bit	Symbol	Description
31 to 4	CHED (Control Head ED)	This field specifies the physical address of the head ED of the control list. These bits must be set for control transfer before the CLE bit of the HcControl register is set. The host controller starts processing of the control list from the HcBulkHeadED pointer.
3 to 0		Reserved The write value should always be 0.

(10) HcControlCurrentED Register (offset: 024h)

Abbreviated name of register: HcControlCurrentED

Address:	027h								026h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	CCED[31:24]								CCED[23:16]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	025h								024h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R
Symbol:	CCED[15:8]								CCED[7:4]				Reserved			
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.12 HcControlCurrentED Register

Bit	Symbol	Description
31 to 4	CCED (Control Current ED)	This pointer indicates the physical address of the ED that is currently processed in the control list. After the current ED processing finishes, this pointer proceeds to the next ED. The host controller continues the list processing until the end of the frame. When the end of the control list is reached, the host controller checks the ControlListFilled bit of the HcCommandStatus register. If the corresponding bit is set to 1b, the contents of the HcControlHeadED field are copied to the HcControlCurrentED field, and the ControlListFilled bit is cleared. If the corresponding bit is set to 0b, nothing is performed. Update of this register is allowed only when the ControlListEnable bit of the HcControl register is cleared. If the ControlListEnable bit is set to 1b, the value of this register is only read. This register is initially set to 0h to indicate the end of the control list.
3 to 0		Reserved The write value should always be 0.

(11) HcBulkHeadED Register (offset: 028h)

Abbreviated name of register: HcBulkHeadED

Address:	02Bh								02Ah							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	BHED[31:24]								BHED[23:16]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	029h								028h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R
Symbol:	BHED[15:8]								BHED[7:4]				Reserved			
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.13 HcBulkHeadED Register

Bit	Symbol	Description
31 to 4	BHED (Bulk Head ED)	This field specifies the physical address of the head ED of the bulk list. This field must be set for bulk transfer before the BLE bit of the HcControl register is set. The host controller starts the processing of the control list from the HcBulkHeadED pointer.
3 to 0		Reserved The write value should always be 0.

(12) HcBulkCurrentED Register (offset: 02Ch)

Abbreviated name of register: HcBulkCurrentED

Address:	02Fh								02Eh							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	BCED[31:24]								BCED[23:16]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	02Dh								02Ch							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R
Symbol:	BCED[15:8]								BCED[7:4]				Reserved			
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.14 HcBulkCurrentED Register

Bit	Symbol	Description
31 to 4	BCED (Bulk Current ED)	This pointer indicates the physical address of the ED that is currently processed in the bulk list. After the current ED processing finishes, this pointer proceeds to the next ED. The host controller continues the list processing until the end of the frame. When the end of the bulk list is reached, the host controller checks the ControlListFilled bit of the HcCommandStatus register. If the corresponding bit is set to 1b, the contents of the HcBulkHeadED field are copied to the HcBulklCurrentED field, and the ControlListFilled bit is cleared. If the corresponding bit is set to 0b, nothing is performed. Update of this register is allowed only when the ControlListEnable bit of the HcControl register is cleared. If the ControlListEnable bit is set to 1b, the value of this register is only read. This register is initially set to 0h to indicate the end of the bulk list.
3 to 0		Reserved The write value should always be 0.

(13) HcDoneHead Register (offset: 030h)

Abbreviated name of register: HcDoneHead

Address:	033h								032h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	DH[31:24]								DH[23:16]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	031h								030h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	DH[15:8]								DH[7:4]				Reserved			
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.15 HcDoneHead Register

Bit	Symbol	Description
31 to 4	DH (Done Head)	This field indicates the physical address of the HcDoneHead of the host controller. The physical address of the TD that lately finished and is added to the Done queue. After the TD processing finishes, the host controller writes the contents of the HcDoneHead to the NextTD field of the TD. At the same time, the host controller overwrites the contents of HcDoneHead with the TD address. After the host controller writes the contents of this register into HCCA, it sets 0b to this register. Then, the WritebackDoneHead bit of the HcInterruptStatus register is set to 1b.
3 to 0		Reserved The write value should always be 0.

(14) HcFmInterval Register (offset: 034h)

Register abbreviation: HcFmInterval

Address:	037h								036h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	FIT	FSMPS[30:24]							FSMPS[23:16]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	035h								034h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved		FI[13:8]						FI[7:0]							
Reset value:	0	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1

Table 32.16 HcFmInterval Register

Bit	Symbol	Description
31	FIT (Frame Interval Toggle)	This bit is used to synchronize the frame setting value with the hardware (host controller). When the FI field is updated, the toggle value is written to this bit. When the FI field value is applied to the FR field of the HcFmRemaining register, the value of this bit is also applied to the FRT bit of the HcFmRemaining register. By checking the toggle value in the FRT bit of the cFmRemaining register, the software can check whether the value in the FI field has been applied to the FR field of the HcFmRemaining register.
30 to 16	FSMPS (FS Largest Data Packet)	This field sets the maximum amount of data (bits) that the host controller can send and receive without the fear of schedule overrun. The host controller compares the current frame position and the setting value in this field to determine the length (of the frame) that is ready to be transferred. This value differs depending on the capacity of the system bus and other reasons, estimate the value and set it to this field. Note: The maximum setting value for this field is 2778h. Do not set any value that is larger than 2778h.
15, 14		Reserved The write value should always be 0.
13 to 0	FI (Frame Interval)	This field is used to set the length of the frame (bit time) used for Full Speed. Set the value of this field to ”2EDFh” so that 1 frame (= 1 ms) of USB standard is satisfied.

(15) HcFmRemaining Register (offset: 038h)

Register abbreviation: HcFmRemaining

Address:	041h								040h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	FRT	Reserved							Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	039h								038h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved		FR[13:8]						FR[7:0]							
Reset value:	0	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1

Table 32.17 HcFmRemaining Register

Bit	Symbol	Description
31	FRT (Frame Remaining Toggle)	This bit is used to synchronize the frame setting value with the hardware (host controller). When the FR (Frame Remaining) field is set to 0000h, the host controller copies the FI (Frame Interval) field value to the FR field, and copies the FIT (Frame Interval Toggle) bit value to this bit. This bit can be used to check that the FI field of the HcFmInterval register has been correctly copied to the FR field.
30 to 14		Reserved The write value should always be 0.
13 to 0	FR (Frame Remaining)	This field indicates the current frame value for 14-bit down counter. The value in this field counts down as time passes. When the value becomes 0000h, the value of FI (Frame Interval) of the HcFmInterval register is loaded. When the state of the host controller is changed to the USB Operational state, the host controller reloads the value in the FI (Frame Interval) field of the HcFmInterval register, and the new value is used from the next SOF.

(16) HcFmNumber Register (offset: 03Ch)

Register abbreviation: HcFmNumber

Address:	03Fh								03Eh							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	03Dh								03Ch							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	FN[15:8]								FN[7:0]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.18 HcFmNumber Register

Bit	Symbol	Description
31 to 16		Reserved The write value should always be 0.
15 to 0	FN (Frame Number)	This field indicates the number of passed frames. When the HcFmRemaining register is reloaded, this field is incremented. If this field is reached to FFFFh, the value is rolled over to 0000h. When the state of the host controller is changed to the USB Operational state, this field is automatically incremented. After the host controller increments the frame number at the frame boundary and sends SOF, the contents of this field are written to HCCA. This is performed before the host controller reads the first ED of the frame. After writing to HCCA, the host controller sets the SF bit of the HcInterruptStatus register.

(17) HcPeriodicStart Register (offset: 040h)

Register abbreviation: HcPeriodicStart

Address:	043h								042h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	041h								040h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved		PS[13:8]						PS[7:0]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.19 HcPeriodicStart Register

Bit	Symbol	Description
31 to 14		Reserved The write value should always be 0.
13 to 0	PS (Periodic Start)	This field indicates the time the host controller starts the periodic list processing in the frame. Estimate an appropriate value, and set the value to this field at the initial setting of the host controller. OHCI standard recommends that you set this setting value to about 90% of the FI field value of the HcFmInterval register. The recommended value is 2A2Fh. When the value in the FR field of the HcFmRemaining register reaches the value set to this field, the periodic list processing is given priority over the control/bulk list processing. Therefore, after the currently running control or bulk transfer finishes, the host controller starts the Interrupt list processing.

(18) HcLSThreshold Register (offset: 044h)

Register abbreviation: HcLSThreshold

Address:	047h								046h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	045h								044h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved				LST[11:8]				LST[7:0]							
Reset value:	0	0	0	0	0	1	1	0	0	0	1	0	1	0	0	0

Table 32.20 HcLSThreshold Register

Bit	Symbol	Description
31 to 12		Reserved The write value should always be 0.
11 to 0	LST (LS Threshold)	This field indicates the threshold value of whether transfer is available for the remaining time of the LS transfer frame. If the value of the FR field of the FmRemaining register is larger than the value set to this field, the host controller can start LS transfer.

(19) HcRhDescriptorA Register (offset: 048h)

Register abbreviation: HcRhDescriptorA

Address:	04Bh								04Ah							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R
Symbol:	POTPGT								Reserved							
Reset value:	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
Address:	049h								048h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R/W	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R
Symbol:	Reserved			NOCP	OCPM	DT	NPS	PSM	NDP							
Reset value:	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1

Table 32.21 HcRhDescriptorA Register

Bit	Symbol	Description
31 to 24	POTPGT (PowerOn To PowerGood Time)	This field indicates the time for waiting until software can access the root hub port after the power is supplied to the port. The unit of time is 2 ms. Therefore, the wait time is POTPGT × 2 ms.
23 to 13		Reserved The write value should always be 0.
12	NOCP (No OverCurrent Protection)	This bit sets whether the overcurrent function of the root hub is supported. If this bit is set to 0b, the overcurrent state is reported depending on the setting of the OCPM bit. 0b: The overcurrent state is supported. 1b: The overcurrent state is not supported. If you do not require the overcurrent function, set this bit to 1b once the module is released from module standby. For details, see section 32.9.1, Host/Peripheral Common Setting Sequence.
11	OCPM (OverCurrent Protection Mode)	This bit sets how to report the overcurrent state of the root hub. If this bit is reset, this bit must indicate the same mode as the PSM (Power Switching Mode) bit. This bit is valid only when the NOCP (NoOverCurrent Protection) bit is cleared (0b). 0b: The overcurrent state is collectively reported for all ports. 1b: The overcurrent state is reported for each port.
10	DT (Device Type)	This bit indicates that the root hub is not a composite device. This bit is always 0b because the root hub is not allowed to be a composite device.
9	NPS (No Power Switching)	This bit sets how to control the port power. If this bit is set to 0b, the PSM bit is used to set whether the power control is collectively performed for all ports or is performed for each port. 0b: The port power can be switched between on and off. 1b: The power is always on while the host controller is running.
8	PSM (Power Switching Mode)	This bit sets how to control the power switch for each port of the root hub. This bit is valid only when the NPS bit is 0b. 0b: The power of all ports is collectively controlled. 1b: The power of ports is controlled for each port. If the PPCM (PortPower Control Mask) bit of the HcRhDescriptorB register is set, each port responds only to the Set/ClearPortPower command. If the PPCM bit is cleared, each port is controlled by the Set/ClearGlobalPower command.
7 to 0	NDP (Number Down stream Ports)	This field indicates the number of downstream ports supported by the root hub.

(20) HcRhDescriptorB Register (offset: 04Ch)

Register abbreviation: HcRhDescriptorB

Address:	04Fh								04Eh							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R
Symbol:	Reserved								Reserved						PPCM	Reserved
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Address:	04Dh								04Ch							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R
Symbol:	Reserved								Reserved						DR	Reserved
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.22 HcRhDescriptorB Register

Bit	Symbol	Description
31 to 18		Reserved The write value should always be 0.
17	PPCM (PortPower Control Mask)	This bit sets the port power control command when the PSM (Power Switching Mode) bit of the HcRhDescriptorA register is set. If this bit is 0b, the Global Power Control command (Set/ClearGlobalPower) is used for control. If this bit is 1b, the Port Power Control command (Set/ClearPortPower) is used for control. If the PSM bit is 0b, this bit is ignored.
16 to 2		Reserved The write value should always be 0.
1	DR (Device Removable)	This bit indicates whether each port of the root hub is removable. If this bit is 0b, the connected device is removable. If this bit is 1b, the connected device is not removable.
0		Reserved The write value should always be 0.

(21) HcRhStatus Register (offset: 050h)

Register abbreviation: HcRhStatus

Address:	053h								052h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W(1)	R/W
Symbol:	CRWE	Reserved							Reserved						OCIC	LPSC
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	051H								050h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Symbol:	DRWE	Reserved							Reserved						OCI	LPS
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.23 HcRhStatus Register

Bit	Symbol	Description
31	CRWE (Clear RemoteWakeup Enable)	This bit is used to clear the DRWE bit. If this bit is set to 1b, DRWE (Device RemoteWakeup Enable) bit can be cleared. Writing 0b to this bit has no effect.
30 to 18		Reserved The write value should always be 0.
17	OCIC (OverCurrent Indicator Change)	This bit is used to report the change in the OCI bit. If there is any change in the OCI bit, the host controller sets this bit to 1b. If 1b is written to this bit while this bit is set to 1b, this bit can be cleared. Writing 0b to this bit has no effect. 0b: There is no change in the Overcurrent state. 1b: There is a change in the Overcurrent state.
16	LPSC	The meaning of this bit differs depending on whether the operation is read or write. [Read] Local Power Status Change This bit is always read as 0b because the Local Power Status is not supported. [Write] Set Global Power If this bit is set to 1b, the power to the ports is turned on. The ports whose power is turned on are determined by the settings of the PSM (Power Switching Mode) bit and the PPCM (Port Power Control Mask) bit of the HcRhDescriptorA register.

written to this bit	PSM	PPCM[N]	Description
0	—	—	Setting ignored
1	0	—	1b is set to the PPS bit.
	1	0	1b is set to the PPS bit.
		1	Setting ignored

Writing 0b to this bit has no effect.

15	DRWE	[Read] Device RemoteWakeup Enable This bit sets whether the RemoteWakeup event includes the CSC (Connect Status Change) bit. If this bit is set to 1b, the CSC bit of the HcRhPortStatus1 register becomes valid as the Resume event. If the CSC bit is changed to 1b, the state is changed from USB Suspend to USB Resume, and the Resume detection interrupt occurs. 0b: Connect Status Change is not the source of RemoteWakeup. 1b: Connect Status Change is the source of RemoteWakeup. [Write] Set RemoteWakeup Enable This bit is used to set the DRWE bit. Writing 0b to this bit has no effect.
14 to 2		Reserved The write value should always be 0.

Table 32.23 HcRhStatus Register

<table>
<tr><th>Bit</th><th>Symbol</th><th>Description</th></tr>
<tr><td>1</td><td>OCI
(Over Current Indicator)</td><td>This bit is used to report the overcurrent state in the global overcurrent detection mode (OCPM bit = 0b).
This bit always indicates 0b when overcurrent for each port is reported (when OPCM bit = 1b).
0b: The port state is normal.
1b: The port is in the overcurrent state.</td></tr>
<tr><td>0</td><td>LPS</td><td>The meaning of this bit differs depending on whether the operation is read or write.

[Read] Local Power Status
This bit is always read as 0b because the Local Power Status is not supported.

[Write] Clear Global Power
If this bit is set to 1b, the power to the ports is turned off.
The ports whose power is turned off are determined by the settings of the PSM (Power Switching Mode) bit and PPCM (Port Power Control Mask) bit of the HcRhDescriptorA register.
<table>
<tr><th>written to this bit</th><th>PSM</th><th>PPCM[N]</th><th>Description</th></tr>
<tr><td>0</td><td>—</td><td>—</td><td>Setting ignored.</td></tr>
<tr><td>1</td><td>0</td><td>—</td><td>The PPS bit is cleared to 0b.</td></tr>
<tr><td></td><td>1</td><td>0</td><td>The PPS bit is cleared to 0b.</td></tr>
<tr><td></td><td></td><td>1</td><td>Setting ignored.</td></tr>
</table>
Writing 0b to this bit has no effect.</td></tr>
</table>

(22) HcRhPortStatus[1:NDP] Register (offset: 054h)

Register abbreviation: HcRhPortStatus1

Address:	057h								056h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)
Symbol:	Reserved								Reserved			PRSC	OCIC	PSSC	PESC	CSC
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	055h								054h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved						LSDA	PPS	Reserved			PRS	POCI	PSS	PES	CCS
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.24 HcRhPortStatus1 Register

Bit	Symbol	Description
31 to 21		Reserved The write value should always be 0.
20	PRSC (Port Reset Status Change)	This bit indicates that port reset (bus reset) has finished. The host controller sets this bit when 10 ms port reset (bus reset) finishes. 0b: Port reset has not finished, or the PRS (Port Reset Status) bit is not changed. 1b: Port reset has finished.
19	OCIC (Port OverCurrent Indicator Change)	This bit is set when the overcurrent state of the port is detected. This bit is set when the host controller changed the POCI bit value. This bit is valid only under the setting that the overcurrent state is reported for each port (OCPM bit = 1b). If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. 0b: The overcurrent state has not changed. 1b: The overcurrent state has changed (POCI bit changed).
18	PSSC (Port Suspend Status Change)	This bit indicates that the Resume sequence finished. This sequence includes 20 ms of the Resume signal, LS EOP, and 3 ms of resynchronization delay. If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. If the PRSC (Port Reset Status Change) bit is set, this bit is cleared. 0b: The Resume sequence has not finished. 1b: The Resume sequence has finished.
17	PESC (Port Enable Status Change)	This bit indicates that the PES (Port Enable Status) bit was changed. If a hardware event clears the PES bit, this bit is set to 1b. If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. 0b: The PES (Port Enable Status) bit has not changed. 1b: The PES (Port Enable Status) bit has changed.
16	CSC (Connect Status Change)	This bit indicates that the CCS (Current Connect Status) bit was changed. If the Connect/Disconnect event occurs, this bit is set to 1b. If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. If a request (Port Reset/Port Enable/Port Suspend) is received during the Disconnect status, this bit is set for reevaluation of device connection confirmation. 0b: The CCS (Current Connect Status) bit has not changed. 1b: The CCS (Current Connect Status) bit has changed.
15 to 10		Reserved The write value should always be 0.

Table 32.24 HcRhPortStatus1 Register

Bit	Symbol	Description
9	LSDA	The meaning of this bit differs depending on whether the operation is read or write. [Read] Low Speed Device Attached This bit indicates the speed of the device connected to the port. This bit is valid only when the CCS (Current Connect Status) bit is set. 0b: Full Speed device is connected. 1b: Low Speed device is connected. [Write] Clear Port Power This bit is used to turn off the power of the port when the port is power controlled. Writing 1b to this bit turns off the port power. Writing 0b to this bit has no effect.
8	PPS	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Power Status This bit indicates the power status of the port. This bit is cleared when the overcurrent is detected. 0b: Port power is off. 1b: Port power is on. [Write] Set Port Power This bit is used to turn on the power of the port when the port is power controlled. Writing 1b to this bit turns on the port power. Writing 0b has no effect.
7 to 5		Reserved The write value should always be 0.
4	PRS	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Reset Status This bit indicates the port reset (bus reset) status. When 10 ms of port reset finishes, the PRSC (Port Reset Status Change) bit is set and this bit is cleared. When the CCS bit is cleared (when no device is connected), this bit cannot be set. 0b: Not during port reset 1b: During port reset [Write] Set Port Reset This bit is used to issue a port reset (bus reset) to the downstream port. Writing 1b to this bit starts 10 ms of port reset. Writing 0b to this bit has no effect. If CCS is cleared, this bit cannot be set. Instead, CSC is set to 1b. This is performed to report to ports to which no device is connected, that port reset was performed.
3	POCI	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port OverCurrent Indicator This bit indicates that the downstream port is in the overcurrent status. This bit is valid only when the setting is specified that the overcurrent status is reported for each port (OCPM bit = 1b). On the other hand, if the setting is specified that the overcurrent status is reported for all ports, this bit is always read as 0b. 0b: The port is in normal status. 1b: The port is in overcurrent status. [Write] Clear Suspend Status This bit is used to finish the Suspend status and start the Resume sequence. Writing 1b to this bit starts the Resume sequence. Writing 0b has no effect. The Resume sequence starts only when PSS (Port Suspend Status) is set.

Table 32.24 HcRhPortStatus1 Register

Bit	Symbol	Description
2	PSS	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Suspend Status This bit indicates that the port status is in Suspend or Resume sequence. 0b: The port is not in Suspend status. 1b: The port is in Suspend status. When the CCS bit is cleared (when no device is connected), this bit cannot be set. This bit is cleared under the following conditions: - When the Resume sequence finished, and the PSSC (Port Suspend Status Change) bit is set - When the port reset finished, and the PRSC (Port Reset Status Change) bit is set - When the host controller is in the USB Resume state [Write] Set Port Suspend This bit is used to change the port status to Suspend. Writing 1b to this bit changes the status to Suspend. Writing 0b to this bit has no effect. If CCS is cleared, this bit cannot be set. Instead, CSC is set to 1b. This is performed to report to the ports to which no device is connected that the Suspend command was issued.
1	PES	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Enable Status This bit indicates whether the port status is Enable or Disable. If the host controller detects a bus error (for example, overcurrent state, disconnect, port power off, and babble error), it clears this bit. Then, the PESC (PortEnableStatusChange) bit is set. When the CCS bit is cleared (when no device is connected), this bit cannot be set. This bit is set "when port reset finished and the PRSC bit is set" or "when the port status becomes Suspend and the PRSC bit is set". 0b: The port status is Disable. 1b: The port status is Enable. [Write] Set Port Enable This bit is used to set the PES bit. Writing 1b to this bit changes the port status to Enable. Writing 0b to this bit has no effect. If CCS is cleared, this bit cannot be set. Instead, CSC is set to 1b. This is performed to report that the status of the ports to which no device is connected was tried to be changed to Enable.
0	CCS	The meaning of this bit differs depending on whether the operation is read or write. [Read] Current Connect Status The current connection status of the downstream port is applied to this bit. 0b: No device is connected. 1b: A device is connected. [Write] Clear Port Enable This bit is used to clear the PES (Port Enable Status) bit. Writing 1b to this bit changes the port status to Disable. Writing 0b to this bit has no effect.

32.2.4.2 EHCI Controller Capability Register

(1) HCIVERSION/CAPLENGTH Register (offset: 102h/100h)

Register abbreviation: CAPL_VERSION

Address:	103h								102h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Interface Version Number[15:8]								Interface Version Number[7:0]							
Reset value:	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0
Address:	101h								100h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Capability Registers Length[7:0]							
Reset value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Table 32.25 CAPL_VERSION Register

Bit	Symbol	Description
31 to 16	Interface Version Number	This field indicates the EHCI version supported by the host controller. 0110h is indicated because this host controller supports EHCI Rev1.1.
15 to 8		Reserved The write value should always be 0.
7 to 0	Capability Registers Length	This field is used as an offset that is added to the base address to find the start address of the EHCI Operational register. 20h is indicated because the EHCI Operation register of this module starts from 20h.

(2) HCSPARAMS Register (offset: 104h)

Register abbreviation: HCSPARAMS

Address:	107h								106h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Debug Port Number [3:0]				Reserved			P_INDICATOR
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	105h								104h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	N_CC				N_PCC				Port Routing Rules	Reserved		PPC	N_PORTS[3:0]			
Reset value:	0	0	0	1	0	0	0	1	1	0	0	1	0	0	0	1

Table 32.26 HCSPARAMS Register

Bit	Symbol	Description
31 to 24		Reserved The write value should always be 0.
23 to 20	Debug Port Number	This field indicates that the host controller port is a debug port. 0000b is indicated because this module does not have a debug port.
19 to 17		Reserved The write value should always be 0.
16	P_INDICATOR (Port Indicators)	This bit indicates whether the host controller supports port indicator control. 0b is indicated because this module does not support port indicator control.
15 to 12	N_CC (Number of Companion Controller)	This field indicates the number of OHCI host controllers implemented in this module. 1h is indicated because this module has one OHCI host controller.
11 to 8	N_PCC (Number of Ports per Companion Controller)	This field indicates the number of ports supported by an OHCI host controller. The setting value of the Port_no field of the PCI Configuration EXT1 register is applied to this field.
7	Port Routing Rules	This bit indicates how individual ports are mapped to the OHCI host controller. This bit indicates 1b because, in this module, the contents of the HCSP_PORTROUTE register show the mapping method.
6, 5		Reserved The write value should always be 0.
4	PPC (Port Power Control)	This bit indicates how the port power of this module is controlled. 0b: The port power is always on. 1b: The PP bit of the PORTSC register controls the port power.
3 to 0	N_PORTS	This field indicates the number of physical downstream ports used by this module. 0001b: 1 Port

(3) HCCPARAMS Register (offset: 108h)

Register abbreviation: HCCPARAMS

Address:	10Bh								10Ah							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved				32-Frame Periodic List Capability	Per-Port Change Event Capability	Link Power Management Capability	Hardware Prefetch Capability
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
Address:	109h								108h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	EECP								Isochronous Scheduling Threshold				Reserved	Asynchronous Schedule Park Capability	Programmable Frame List Flag	64-bit Addressing Capability
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0

Table 32.27 HCCPARAMS Register

Bit	Symbol	Description
31 to 20		Reserved The write value should always be 0.
19	32-Frame Periodic List Capability	This bit indicates 1b because this module supports 32 Frame Periodic List defined in EHCI V1.1. This means that the Frame List Size field of the USBCMD register is set to 11b, so this module supports 32 Frame Periodic List.
18	Per-Port Change Event Capability	This bit indicates 1b because this module supports event detection function for ports that are defined in EHCI V1.1. This means that this module supports event detection function for ports, and is associated with the Pre-Port Change Event Enable field of the USBCMD register, Port-1 Change Detect field of the USBSTS register, and Port-1 Change Interrupt Enable field of the USBINT register. If this bit is 0b, the corresponding fields of the above registers are treated as Reserved
17	Link Power Management Capability	This bit indicates 1b because this module supports LPM (Link Power Management) defined in EHCI V1.1. This means that this module supports LPM L1 state, and is controlled by the Suspend using L1 bit, Suspend Status bit, and Device Address field of the PORTSC register. If this bit is 0b, the corresponding bits and field of the above PORTC register are treated as Reserved.
16	Hardware Prefetch Capability	This bit indicates 0b because this module does not support the hardware prefetch function defined in EHCI V1.1.
15 to 8	EECP	This field indicates the offset address of the EHCI Extend Capabilities Registers. This field indicates 00h because this module does not use EHCI Extend Capabilities Registers.
7 to 4	Isochronous Scheduling Threshold	This field indicates 0h because this module does not support caches with isochronous data structure for the entire frame.
3		Reserved The write value should always be 0.
2	Asynchronous Schedule Park Capability	This bit indicates whether the Park mode is supported for High Speed QH (Queue Head) in an asynchronous schedule. This bit indicates 1b because this module supports the above function.
1	Programmable Frame List Flag	This bit indicates the setting for the available frame list size. This bit indicates 1b for this module. If this bit is set to 1b, bit [3:2] (Frame List Size) of the USBCMD register can be used to set the available frame list size, and the frame list size smaller than 4 Kbyte is configurable.
0	64-bit addressing Capability	This bit indicates which type of memory pointer the data structure uses (32 bit address memory pointer or 64 bit address memory pointer). This bit indicates 0b for this module because this module has the data structure that uses 32 bit address memory pointer. 64 bit address is not supported.

(4) HCSP-PORTROUTE Register (offset: 10Ch)

Register abbreviation: HCSP_PORTROUTE

Address:	10Fh								10Eh							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Companion Port Route[31:24]								Companion Port Route[23:16]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	010Dh								010Ch							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Companion Port Route[15:8]								Companion Port Route[7:0]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.28 HCSP-PORTROUTE Register

Bit	Symbol	Description
31 to 0	Companion Port Route[31:0]	This field indicates the ports controlled by the OHCI host controller. This field indicates 0b for this module because this module has one OHCI host controller.

32.2.4.3 EHCI Operational Register

(1) USBCMD Register (offset: 120h)

Register abbreviation: USBCMD

Address:	123h								122h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved				Host-Initiated Resume Duration				Interrupt Threshold Control							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Address:	121h								120h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R	R	R	R/W	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	W	R/W
Symbol:	Per-Port Change Events Enable	Reserved			Asynchronous Schedule Park Mode Enable	Reserved	Asynchronous Schedule Park Mode Count		Reserved	Interrupt on Async Advance Doorbell	Asynchronous Schedule Enable	Periodic Schedule Enable	Frame List Size		HCRESET	RS
Reset value:	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0

Table 32.29 USBCMD Register

Bit	Symbol	Description
31 to 28		Reserved The write value should always be 0.
27 to 24	Host-Initiated Resume Duration	This field indicates the minimum time of the K-state drive while the host controller resumes from the LPM (L1) state. The value in this field is sent to the connected device that has the LPM function via the HIDR field in the bmAttributes field of the LPM token. The encoded value of this field is defined as the name of the HIRD field in the LPM Token. Specifically, 0h means 50 µs, and if the value is incremented by 1, 75 µs is incremented. For example, 1h means 125 µs, and 4h means 1175 µs.
23 to 16	Interrupt Threshold Control	This field indicates the maximum rate until the host controller issues an interrupt. Note that the setting values other than the following values are not guaranteed. Value / Maximum Interrupt Interval 00h: Reserved 01h: 1 micro-frame 02h: 2 micro-frames 04h: 4 micro-frames 08h: 8 micro-frames (default, equals to 1 ms) 10h: 16 micro-frames (2 ms) 20h: 32 micro-frames (4 ms) 40h: 64 micro-frames (8 ms) Do not set 00h to this field while the Halted bit is 0b.
15	Per-Port Change Events Enable	This field is used to enable the event report function of the ports defined by the Port-1 Change Detect field of the USBSTS register and the Port-1 Change Detect Enable field of the USBINTR register. 0b: The event report function of the ports is disabled. 1b: The event report function of the ports is enabled.
14 to 12		Reserved The write value should always be 0.
11	Asynchronous Schedule Park Mode Enable	This bit enables or disables the Asynchronous Schedule Park mode. 0b: The Park mode is disabled. 1b: The Park mode is enabled.
10		Reserved The write value should always be 0.

Table 32.29 USBCMD Register

Bit	Symbol	Description
9, 8	Asynchronous Schedule Park Mode Count	This field sets the number of transactions that the host controller can serially execute for one QH (Queue Head) fetch in an asynchronous schedule. The valid value range is 1h to 3h. This field is valid when bit 11 (Asynchronous Schedule Park Mode Enable) is 1b. Do not set 0h to this field.
7		Reserved The write value should always be 0.
6	Interrupt on Async Advance Doorbell	This bit is used as the doorbell. In an asynchronous schedule processing, if you want an interrupt to occur before proceeding to the next QH (Queue), set this bit to 1b. After a QH processing normally finishes, the host controller clears this bit to 0b, and sets bit 5 (Interrupt on Async Advance bit) of the USBSTS register to 1b. If bit 5 (Interrupt on Async Advance Enable bit) of the UBINTR register is set to 1b, an interrupt occurs at the next interrupt timing. If the asynchronous schedule is disabled, do not write 1b to this bit.
5	Asynchronous Schedule Enable	This bit sets whether the host controller proceeds to the asynchronous list processing or skip the processing. 0b: The asynchronous list processing is skipped. 1b: Use the ASYNCLISTADDR register to proceed the asynchronous list processing.
4	Periodic Schedule Enable	This bit sets whether the host controller proceeds or skips the periodic list processing. 0b: The periodic list processing is skipped. 1b: Use the PERIODICLISTBASE register to proceed the periodic list processing.
3, 2	Frame List Size	This field specifies the frame list size. The setting value of this field determines the size of the Frame List Current index of the FRINDEX register. Value — the number of frames in Frame List 00b — 1024 frames (default) 01b — 512 frames 10b — 256 frames 11b — 32 frames
1	HCRESET (Host Controller Reset)	This bit is used to initialize the EHCI circuit of the host controller. If this bit is set to 1b, the host controller initializes the internal pipelines, counters, and state machines, and communications on the USB immediately stop. At this time, port reset is not issued to the downstream ports. Reset by this bit has no effect on the registers other than the EHCI Operational register. The EHCI Operational register is initialized, and the port owner returns to OHCI. Software must be reset to return the host controller to the operational state. When this reset processing finishes, the host controller sets this bit to 0b. Writing 0b to this bit cannot stop the reset processing. If the HCHalted bit of the USBSTS register is set to 0b, do not set 1b to this bit.
0	RS (Run/Stop)	This bit is used to run or stop the EHCI host controller. If this bit is set to 1b, the host controller starts operation. As long as this bit is set to 1b, the host controller continues running. If this bit is set to 0b, the host controller finishes the currently executing transaction and some other transactions, and then is changed to the Halt status. The HCHalted bit of the USBSTS register indicates that the host controller finished the transaction processing and entered into the stop status. If the host controller is in a status other than Halt (the HCHalted bit of the USBSTS register is 1b), do not write 1b to this bit.

(2) USBSTS Register (offset: 124h)

Register abbreviation: USBSTS

Address:	127h								126h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W(1)
Symbol:	Reserved								Reserved							Port-1 Change Detect
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	125h								124h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)
Symbol:	Asynchronous Schedule Status	Periodic Schedule Status	Reclamation	HCHalted	Reserved				Reserved		Interrupt on Async Advance	Host System Error	Frame List Rollover	Port Change Detect	USBERRINT	USBINT
Reset value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.30 USBSTS Register

Bit	Symbol	Description
31 to 17		Reserved The write value should always be 0.
16	Port-1Change Detect	If this bit is set to 1b, it indicates that a change in the port status was detected. This bit is used only when the Per-Port Change Events Enable bit of the USBCMD register is set to 1b.
15	Asynchronous Schedule Status	This bit indicates the current status of the asynchronous schedule. 0b: The asynchronous schedule is disabled. 1b: The asynchronous schedule is enabled. If this bit and bit 5 (Asynchronous Schedule Enable) of the USBCMD register have the same value, the asynchronous schedule is enabled (1b) or disabled (0b).
14	Periodic Schedule Status	This bit indicates the current status of the periodic schedule. 0b: The periodic schedule is disabled. 1b: The periodic schedule is enabled. If this bit and bit 4 (Periodic Schedule Enable) of the USBCMD register have the same value, the periodic schedule is enabled (1b) or disabled (0b).
13	Reclamation	This bit is used to detect an empty asynchronous schedule. If this bit is 1b, the asynchronous schedule is empty. After reset or when a QH (H = 1) is fetched, the host controller clears this bit to 0b. Also, when the host controller executes an asynchronous transaction or detects a start event, it sets this bit to 1b. If this bit is 0b and a QH (H = 1) is fetched, the host controller is entered into the Async Sched Sleeping mode.
12	HCHalted	If the Run/Stop bit of the USBCMD register is 1b, this bit indicates 0b. If the software or host controller sets the Run/Stop bit to 0b, the host controller stops operation, and sets 1b to this bit. 0b: The EHCI host controller is running. 1b: The EHCI host controller is stopped.
11 to 6		Reserved The write value should always be 0.
5	Interrupt on Async Advance	This bit indicates the Async Advance Interrupt status. After the host controller fetches the QH, it checks bit 6 (Interrupt on Async Advance Doorbell [IAAD] bit) of the USBCMD Register. If the IAAD bit is set to 1b, the host controller clears the IAAD bit after the QH processing normally finishes, and sets this bit. If bit 5 (Interrupt on Async Advance Enable bit) of the UBINTR register is set to 1b, an interrupt due to this source will occur at the next interrupt timing after 1b is set to this bit. If HCD writes 1b to this bit, this bit can be cleared. Writing 0b to this bit has no effect. 0b: Async Advance Interrupt not occurred. 1b: Async Advance Interrupt status is detected.

Table 32.30 USBSTS Register

Bit	Symbol	Description
4	Host System Error	This bit is set to 1b if a serious error occurs in the host controller. If this error occurs, the host controller clears the Run/Stop bit of the USBCMD register to 0b so that the subsequent schedules are not executed. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: No system error occurred. 1b: A system error occurred.
3	Frame List Rollover	If a frame list rollover occurs, the host controller sets this bit to 1b. Specifically, when the Frame Index field of the FRINDEX register is returned to 000h from the maximum value (rollover), the host controller sets this bit to 1b. The maximum value (the value at which rollover occurs) depends on the Frame List Size field of the USBCMD register. For example, if the Frame List Size is 1024 frame, rollover occurs every time FRIMDEX [13] toggles. If the Frame List Size is 512 frame, rollover occurs every time FRINDEX [12] toggles. If the Frame List Size is 256 frame, rollover occurs every time FRINDEX [11] toggles. Writing 1b to this bit can clears this bit. Writing 0b to this bit has no effect. 0b: The frame list is not returned to 000h. 1b: The frame list is returned to 000h (rollover occurred).
2	Port Change Detect	This bit indicates that the port status has changed. Among the ports for which the Port Owner bit of the PORTSC1 register is set to 0b, if any port satisfies one of the following conditions, the host controller sets this bit to 1b: - Connect or Disconnect status of a device is detected, and the Connect Status Change bit of the PORTSC1 register is changed from 0 to 1. - A change of the Enable status of the port is detected, and the Port Enable/Disable Change bit of the PORTSC1 register is changed from 0 to 1. - The overcurrent state is detected, and the Over-current Change bit of the PORTSC1 register is changed from 0 to 1. - J-K transition is detected on a port in the Suspend status, and the Force Port Resume bit of the PORTSC1 register is changed from 0 to 1. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect.
1	USBERRINT (USB Error Interrupt)	This bit indicates that a USB transaction finished with an error. When a USB transaction finishes with an error, the host controller sets this bit to 1b. If 1b is set to the IOC bit of qTD at which an error interrupt occurred, 1b is set to both of this bit and USBINT bit. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: USB transaction is normal. 1b: USB transaction finished with an error.
0	USBINT (USB Interrupt)	This bit indicates that a USB transfer has finished. The host controller sets this bit to 1b if one of the following conditions is satisfied: - A USB transfer has finished. - A short packet is received. Even if a USB transfer finished with an error, if IOC (Interrupt On Complete) of the TD is set to 1b, this bit is set to 1b. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: A USB transfer has not finished. 1b: A USB transfer has finished.

(3) USBINTR Register (offset: 128h)

Register abbreviation: USBINTR

Address:	12Bh								12Ah							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Symbol:	Reserved								Reserved							Port-1 Change Event Enable
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	129h								128h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved								Reserved		Interrupt on Async Advance Enable	Host System Error Enable	Frame List Rollover Enable	Port Change Detect Enable	USB Error Interrupt Enable	USB Interrupt Enable
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.31 USBINTR Register

Bit	Symbol	Description
31 to 17		Reserved The write value should always be 0.
16	Port-1 Change EventEnable	If this bit is 1b and the Port Change Detect bit of the USBSTS register is set to 1b, the host controller issues an interrupt.
15 to 6		Reserved The write value should always be 0.
5	Interrupt on Async Advance Enable	This bit sets whether bit 5 (Interrupt on Async Advance [IAA] bit) of the USBSTS register is enabled or disabled. If the IAA bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the IAA bit).
4	Host System Error Enable	This bit sets whether bit 4 (Host System Error [HSE] bit) of the USBSTS register is enabled or disabled. If the HSE bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the HSE bit).
3	Frame List Rollover Enable	This bit sets whether the bit 3 (Frame List Rollover [FLR] bit) of the USBSTS register is enabled or disabled. If the FLR bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the FLR bit).
2	Port Change Detect Enable	This bit sets whether bit 2 (Port Change Detect [PCD] bit) of the USBSTS register is enabled or disabled. If the USBERRINT bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the PCD bit).
1	USB Error Interrupt Enable	This bit sets whether bit 1 (USBERRINT bit) of the USBSTS register is enabled or disabled. If the USBERRINT bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the USBERRINT bit).
0	USB Interrupt Enable	This bit sets whether bit 0 (IUSBINT bit) of the USBSTS register is enabled or disabled. If the USBINT bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the USBINT bit).

(4) FRINDEX Register (offset: 12Ch)

Register abbreviation: FRINDEX

Address:	12Fh								12Eh							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	12Dh								12Ch							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved		Frame Index[13:8]						Frame Index[7:0]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.32 FRINDEX Register

Bit	Symbol	Description
31 to 14		Reserved The write value should always be 0.
13 to 0	Frame Index [13:0]	This field is used by the host controller to add an index to the periodic frame list. The value in this field is incremented at the end of a micro frame. Bit [N:3] of this field is used as the Frame List Current index. This means that, before the next index arrives, the current frame list is accessed 8 times. The value for N is determined, as follows, by the setting value of bit [3:2] (Frame List Size field) of the USBCMD register.

Frame List Size	Number of Frames	N
00b	1024	12
01b	512	11
10b	256	10
11b	32	12

Access this register only when the host controller is in stop status (bit 12 [HCHalted] = 1b).
The setting value of this field is applied to the SOF frame number of the SOF token.

(5) CTRLDSSEGMENT Register (offset: 130h)

Register abbreviation: CTRLDSSEGMENT

Address:	133h								132h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	CTRLDSSEGMENT[31:24]								CTRLDSSEGMENT[23:16]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	131h								130h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	CTRLDSSEGMENT[15:8]								CTRLDSSEGMENT[7:0]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.33 CTRLDSSEGMENT – Control Data Structure Segment Register

Bit	Symbol	Description
31 to 0	CTRLDSSEGMENT	This register is not used because this module does not support 64-bit address method. Therefore, HCD must not access this register.

(6) PERIODICLISTBASE Register (offset: 134h)

Register abbreviation: PERIODICLISTBASE

Address:	137h								136h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Base Address[31:24]								Base Address[23:16]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	135h								134h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Base Address[15:12]				Reserved				Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.34 PERIODICLISTBASE Register

Bit	Symbol	Description
31 to 12	Base Address	This field indicates the head address of the periodic frame list on the system memory. The host controller loads the contents of this register before starting the list processing. The host controller determines the frame list to be processed by using this field and Frame Index of the FRINDEX register. Align the address of the periodic frame list by 4 Kbyte. Normal operation is not guaranteed if any of these bits are changed during operation.
11 to 0		Reserved The write value should always be 0.

(7) ASYNCLISTADDR Register (offset: 138h)

Register abbreviation: ASYNCLISTADDR

Address:	13Bh								13Ah							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	LPL[31:23]								LPL[23:16]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	139h								138h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R
Symbol:	LPL[15:8]								LPL[7:5]			Reserved				
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.35 ASYNCLISTADDR Register

Bit	Symbol	Description
31 to 5	LPL (Link Pointer Low)	This field indicates the address (on the system memory) of the Asynchronous Queue Head to be processed next time. Align the address of Asynchronous Queue Head by 32 byte.
4 to 0		Reserved The write value should always be 0.

(8) CONFIGFLAG Register (offset: 160h)

Register abbreviation: CONFIGFLAG

Address:	163h								162h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	161h								160h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Symbol:	Reserved								Reserved							CF
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.36 CONFIGFLAG Register

Bit	Symbol	Description
31 to 1		Reserved The write value should always be 0.
0	CF (Configuration Flag)	This bit controls which of OHCI or EHCI is routed by the port routing control circuit by default. At the end of the host controller configuration, this bit is set to 1b. 0b: The port routing control circuit routes each port to the OHCI host controller by default. 1b: The port routing control circuit routes each port to the EHCI host controller by default.

(9) PORTSC1Register (offset: 164h)

Register abbreviation: PORTSC1

Address:	167h								166h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Device Address							Suspend Status[1]	Suspend Status[0]	WKOC_E	WKDSCNNT_E	WKCNNT_E	Port Test Control			
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	165h								164h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W(1)	R	R/W(1)	R/W	R/W(1)	R
Symbol:	Reserved		Port Owner	PP	Line Status		Suspend using L1	Port Reset	Suspend	Force Port Resume	Over current Active Change	Over-current Active	Port Enabled/ Disabled Change	Port Enabled/ Disabled	Connect Status Change	Current Connect Status
Reset value:	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.37 PORTSC1 Register

Bit	Symbol	Description
31 to 25	Device Address	This field indicates the USB device address (7 bits) of the device connected to the down port. This address is used when the LPM Token is sent. If the value of this filed is 00h, it means that no device that needs to use this field has been connected.
24, 23	Suspend Status	This field indicates the response from the connected device to the LPM Token (L1 transition request). 00b: The device succeeded transition to the L1 state (ACK received from the device). 01b: The device has not changed to the L1 state (NYET received from the device). 10b: The device does not support the L1 state transition (STALL received from the device). 11b: Other response (for example, Timeout error) Change this field only when the Suspend bit is 0b.
22	WKOC_E (Wake on Overcurrent Enable)	By writing 1b to this bit, the overcurrent state can be detected as an EHCI Wakeup event. If bit 12 (PP [Port Power] bit) is 0b, this bit becomes 0b.
21	WKDSCNNT_E (Wake on Disconnect Enable)	By writing 1b to this bit, device disconnection can be detected as an EHCI Wakeup event. If bit 12 (PP [Port Power] bit) is 0b, this bit becomes 0b.
20	WKCNNT_E (Wake on Connect Enable)	By writing 1b to this bit, device connection can be detected as an EHCI Wakeup event. If bit 12 (PP [Port Power] bit) is 0b, this bit becomes 0b.
19 to 16	Port Test Control	This field is controlled by the test mode. If the value of this field is other than 0000b, it indicates that this module is running in the test mode.

Value	Test mode
0000b	Test mode not enabled
0001b	Test J_STATE
0010b	Test K_STATE
0011b	Test SE0_NAK
0100b	Test Packet
0101b	Test FORCE_ENABLE
Other	Reserved

15, 14		Reserved The write value should always be 0.

Table 32.37 PORTSC1 Register

<table>
<tr><th>Bit</th><th>Symbol</th><th>Description</th></tr>
<tr><td>13</td><td>Port Owner</td><td>This bit indicates which of OHCI or EHCI has the port ownership.
0b: EHCI has the port ownership.
1b: OHCI has the port ownership.
When bit 0 (Configure Flag bit) of the CONFIGFLA register is changed from 0b to 1b, this bit becomes 0b. If bit 0 (Configure Flag bit) of the CONFIGFLA register is 0b, this bit becomes 1b.
If the connected device is not a High Speed device, this bit is set to 1b to transfer the port ownership to OHCI.</td></tr>
<tr><td>12</td><td>PP
(Port Power)</td><td>This bit controls power supply to the port.
If this bit is 0b, power is not supplied to the port. Therefore, the port does not function, and does not recognize connection and disconnection.
If overcurrent is detected while this bit is set to 1b, the host controller clears this bit to 0b, and the power supplied to the port is stopped.
Note: As described later, if the PPC bit is 0b, this bit is fixed to 1b, so the power supplied to the port is not stopped.

The function of this bit differs depending on the value of bit 4 (PPC [Port Power Control] bit) of the HCSPARAMS register.
<table>
<tr><th>PPC</th><th>PP</th><th>Operation</th><th></th></tr>
<tr><td>0b</td><td>1b</td><td>R</td><td>This bit is fixed to 1b, and the power is always supplied to the port.</td></tr>
<tr><td>1b</td><td>0b/1b</td><td>R/W</td><td>Whether or not power is supplied to the port depends on the setting of this bit.</td></tr>
<tr><td></td><td></td><td></td><td>0b Power is not supplied to the port.</td></tr>
<tr><td></td><td></td><td></td><td>1b Power is supplied to the port.</td></tr>
</table></td></tr>
<tr><td>11, 10</td><td>Line Status</td><td>This field indicates the logical level of D+/D- lines of the current USB bus. (bit 11: DP / bit 10: DM)
This field is used to detect an LS device before starting a sequence for port reset and port enable.
Therefore, this bit is valid only when bit 3 (Port Enable/Disable bit) is 0b and bit 0 (Current Connect Status bit) is 1b.
<table>
<tr><th>bit 11</th><th>bit 10</th><th>USB bus status</th><th>Description</th></tr>
<tr><td>0b</td><td>0b</td><td>SE0</td><td>The device is not an LS device.</td></tr>
<tr><td></td><td></td><td></td><td>EHCI port reset is executing.</td></tr>
<tr><td>1b</td><td>0b</td><td>J-state</td><td>The device is not an LS device.</td></tr>
<tr><td></td><td></td><td></td><td>EHCI port reset is executing.</td></tr>
<tr><td>0b</td><td>1b</td><td>K-state</td><td>An LS device was connected.</td></tr>
<tr><td></td><td></td><td></td><td>The port ownership is transferred from EHCI to OHCI.</td></tr>
<tr><td>1b</td><td>1b</td><td>Undefined</td><td>The device is not an LS device.</td></tr>
<tr><td></td><td></td><td></td><td>EHCI port reset is executing.</td></tr>
</table></td></tr>
<tr><td>9</td><td>Suspend using L1</td><td>Suspend using L1 (LPM) control bit.
If 1b is written to the Suspend bit (bit 7) while this bit is 1b, the state of this module is changed to the LPM state.
Writing to this bit is possible only when the Suspend bit (bit 7) is 0b.
0b: Suspend using L2
1b: Suspend using L1 (LPM)
If this bit is 1b and the Device Address field is other than 0000h, when the Suspend bit is set to 1b, the host controller generates an LPM Token to change the state to the L1 state.
If this bit is 0b, the host controller operates as L2 Suspend.</td></tr>
</table>

Table 32.37 PORTSC1 Register

<table>
<tr><th>Bit</th><th>Symbol</th><th>Description</th></tr>
<tr><td>8</td><td>Port Reset</td><td>This bit indicates the reset status of the port.
0b: The port is not being reset.
1b: The port is being reset.
If 1b is written to this bit while this bit is 0b, the bus reset sequence defined in USB 2.0 standard starts. To finish the bus reset sequence, 0b must be written to this bit. Note that this bit must remain 1b long time enough to guarantee that the bus reset sequence defined in USB 2.0 standard will be complete.
If bit 12 (HCHalted) of the USBSTS register is 1b, do not set this bit to 1b.
If any of the PP (Port Power) bit, Port Owner bit, and Current Connect Status bit is in the following status, this bit becomes 0b.
Note: Even if 1b is written to this bit, the bus reset sequence does not start.
- PP (Port Power) bit = 0b
- Port Owner bit = 1b
- Current Connect Status bit = 0b</td></tr>
<tr><td>7</td><td>Suspend</td><td>This bit indicates the Suspend control and status of the port.
This bit and bit 2 (Port Enabled/Disabled bit) indicate the following port status.
<table>
<tr><th>Port Enabled</th><th>Suspend</th><th>Port Status</th></tr>
<tr><td>0b</td><td>—</td><td>Disable</td></tr>
<tr><td>1b</td><td>0b</td><td>Enable</td></tr>
<tr><td></td><td>1b</td><td>Suspend</td></tr>
</table>
To change the port state to L1 or L2 Suspend, set this bit to 1b.
Whether the host controller supports L1 Suspend state or L2 Suspend state depends on the value in the Suspend Using L1 bit.
In the Suspend state, data transfer to the downstream port is blocked by this port (except for port reset). If this bit is set to 1b during data transfer, blocking of transfer data does not occur until the current data transfer finishes.
Writing 0b to this bit has no effect.
This bit can be set to 1b only when all the following conditions are satisfied: ”PP (Port Power) bit = 1b”, “Port Owner bit = 0b”, and “Current Connect Status bit = 1b”. If any of the following conditions is satisfied, the host controller clears this bit:
- "Resume is complete" is detected.
- The PR bit is set to 1b when PR (Port Reset) bit is 0b.
- The port owner is 1b (OHCI).
- The PP (Port Power) bit is set to 0b.
- The Port Enabled/Disabled bit is set to 0b.</td></tr>
<tr><td>6</td><td>Force Port Resume</td><td>This bit indicates that the Resume state of the port is detected.
0b: Resume signal is not detected or output.
1b: Resume (K-state) is detected or output.
When the port is in Suspend state, if the host controller detects transition of the state from J to K (if RemoteWakeup is detected from the connected device), it sets this bit to 1b. The host controller also sets the Port Change Detect bit or Port-1 Changes Detect bit of the USBSTS register to 1b.
If this bit is set to 1b, the host controller does not set 1b to the Port Change Detect bit and Port-1 Changes Detect bit of the USBSTS register. While this bit is 1b, the Resume signal (FS K) is driven onto the USB bus.
For L2 transition, this bit must be cleared to 0b after an appropriate time has passed. By writing 0b to this bit while this bit is 1b, the port status is recovered to be the HS Idle status. This bit remains 1b until the port is recovered. The host controller must finish transition to the HS Idle state within 2msec since this bit is cleared to 0b.
On the other hand, for L1 transition, the host controller sends a Resume signal at necessary timing, and this bit is cleared to 0b at Resume recovery. Note that the software sets the length of the Resume signal driven by the host controller, by using the Host-Initiated Resume Duration field of the USBCMD register.
If the Port Power (PP) bit is 0b, this bit becomes 0b.</td></tr>
<tr><td>5</td><td>Over-current Change</td><td>This bit indicates that bit 4 (Over-current Active bit) has changed.
Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect.
0b: Over-current Active bit has not changed.
1b: Over-current Active bit has changed.</td></tr>
</table>

Table 32.37 PORTSC1 Register

Bit	Symbol	Description
4	Over-current Active	This bit indicates the overcurrent status of the port. If the host controller detects overcurrent, it disables the port, and set this bit to 1b. After the overcurrent state is released, the host controller automatically clears this bit from 1b to 0b. 0b: The port is not in overcurrent state. 1b: The port is in overcurrent state.
3	Port Enable/Disable Change	This bit indicates that the host controller detected frame babble. If the Host Controller detects frame babble, it disables the port and sets this bit to 1b. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. If the Port Power (PP) bit is 0b, this bit becomes 0b. 0b: Frame babble has not occurred. 1b: Frame babble is detected.
2	Port Enabled/ Disabled	This bit indicates the Enable/Disable status of the port. The host controller resets the port, and enables the port if the connected device is recognized as an HS device, and sets this bit to 1b. The software cannot set this bit to 1b. If the host controller detects disconnection of a device or other errors, it disables the port, and clears this bit to 0b. The port also becomes disabled when 0b is written to this bit. If the port is disabled, data transfer to the downstream port is blocked except for port reset. If Port Test Control [3:0] = 0101b (Test FORCE_ENABLE), the port becomes enabled, and this bit is set to 1b. If the Port Power (PP) bit is 0b, this bit becomes 0b. 0b: The port is disabled. 1b: The port is enabled.
1	Connect Status Change	This bit indicates that bit 0 (Current Connect Status bit) has changed. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. If the Port Power (PP) bit is 0b, this bit becomes 0b. 0b: The Current Connect Status bit has no change. 1b: The Current Connect Status bit has changed.
0	Current Connect Status	This bit indicates the connection status of the port. If the host controller detects connection of a device, it sets this bit to 1b. Also, if Port Test Control [3:0] = 0101b (Test FORCE_ENABLE), the Host Controller sets this bit to 1b even if no device is connected. On the other hand, if the host controller detects disconnection of a device, it sets this bit to 0b. If the Port Power (PP) bit is 0b, or the Port Owner (PO) bit is 0b, this bit becomes 0b. 0b: No device is connected to the port. 1b: A device is connected to the port.

32.2.4.4 AHB Bridge Register

(1) INT_ENABLE Register (offset: 200h)

Abbreviated name of register: INT_ENABLE

Address:	203h								202h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	201h								200h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved								Reserved			WAKEON_INTEN	UCOM_INTEN	USBH_INTBEN	USBH_INTAEN	AHB_INTEN
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.38 INT_ENABLE Register

Bit	Symbol	Description
31 to 5		Reserved The write value should always be 0.
4	WAKEON_INTEN	This bit enables or disables bit 4 (WAKEON_INT) of the INT_STATUS register. 0b: disable 1b: enable
3	UCOM_INTEN	This bit enables or disables bit 3 (UCOM_INT) of the INT_STATUS register. 0b: disable 1b: enable
2	USBH_INTBEN	This bit enables or disables bit 2 (USBH_INTB) of the INT_STATUS register. 0b: disable 1b: enable
1	USBH_INTAEN	This bit enables or disables bit 1 (USBH_INTA) of the INT_STATUS register. 0b: disable 1b: enable
0	AHB_ INTEN	This bit enables or disables bit 0 (AHB_ INT) of the INT_STATUS register. 0b: disable 1b: enable

(2) INT_STATUS Register (offset: 204h)

Abbreviated name of register: INT_STATUS

Address:	207h								206h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	205h								204h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R/W(1)	R	R	R	R/W(1)
Symbol:	Reserved								Reserved			WAKE ON_ INT	UCOM _INT	USBH _INTB	USBH _INTA	AHB _INT
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.39 INT_STATUS Register

Bit	Symbol	Description
31 to 5		Reserved The write value should always be 0.
4	WAKEON_INT	This bit indicates the state of WAKEON interrupt from the HOST module. Writing 1b to this bit can clear this bit. 0b: No WAKEON interrupt 1b: WAKEON interrupt
3	UCOM_INT	This bit indicates the state of interrupt from the UCOM register. To clear the interrupt, use the UCOM2 Register. 0b: No UCOM register interrupt 1b: UCOM register interrupt
2	USBH_INTB	This bit indicates the state of EHCI interrupt. To clear the interrupt, use the USBSTS Register (of the EHCI Operational Register). 0b: No INTB interrupt 1b: INTB interrupt
1	USBH_INTA	This bit indicates the state of OHCI interrupt. To clear the interrupt, use the HcInterruptStatus Register (of the OHCI Operational Register). 0b: No INTA interrupt 1b: INTA interrupt
0	AHB_INT	This bit indicates that a BUS Master error occurred. Writing 1b to this bit can clear this bit. 0b: No bus error occurred. 1b: A bus error occurred.

(3) AHB_BUS_CTR Register (offset: 208h)

Abbreviated name of register: AHB_BUS_CTR

Address:	20Bh								20Ah							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	209h								208h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R/W	R/W
Symbol:	PROT_TYPE				Reserved			PROT_ MODE	Reserved		ALIGN_ ADDRESS		Reserved		MAX_BURST _LEN	
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.40 AHB_BUS_CTR Register

Bit	Symbol	Description
31 to 16		Reserved The write value should always be 0.
15 to 12	PROT_TYPE	This field sets MHPROT [3:0] used when the BUS Master interface issues a transfer request. bit 15 0b: Cache disabled. 1b: Cache enabled. bit 14 0b: Buffer disabled. 1b: Buffer enabled. bit 13 0b: User access 1b: Privileged access bit 12 0b: Operation code 1b: Data
11 to 9		Reserved The write value should always be 0.
8	PROT_MODE	This bit selects the mode of MHPROT [3:0] used when the Master interface issues a transfer request. 0b: The value of PROT_TYPE is output as MHPROT [3:0]. 1b: When a DMA transfer is performed, MHPROT [3:0] is set to 0000b if the final burst is performed, or MHPROT [3:0] is set to the PROT_TYPE value if another burst transfer is performed.
7, 6		Reserved The write value should always be 0.
5, 4	ALIGN_ ADDRESS	This field sets the address boundary used when the BUS Master interface issues a burst transfer. 00b: A burst transfer is issued so that not to exceed 1-Kbyte boundary. 01b: A burst transfer is issued so that not to exceed 64-byte boundary. 10b: A burst transfer is issued so that not to exceed 32-byte boundary. (The maximum burst length is INCR8. This is because, if INCR16 is used, 32-byte boundary is exceeded.) 11b: A burst transfer is issued so that not to exceed16 byte boundary. (The maximum burst length is INCR4. This is because, if the length is at least INCR8, 16-byte boundary is exceeded.)
3, 2		Reserved The write value should always be 0.
1 to 0	MAX_BURST _LEN	This field selects the maximum burst length used when the BUS Master interface issues a transfer request. 00b: INCR16 01b: INCR8 10b: INCR4 11b: SINGLE

(4) USBCTR Register (offset: 20Ch)

Abbreviated name of register: USBCTR

Address:	20Fh								20Eh							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	20Dh								20Ch							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	W
Symbol:	Reserved								Reserved					DIRPD	PLL_RST	USBH_RST
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Table 32.41 USBCTR Register

Bit	Symbol	Description
31 to 3		Reserved The write value should always be 0.
2	DIRPD	USBPHY Standby Mode Control 0b: USBPHY normal operating mode 1b: USBPHY standby mode [Note] When the USB module is not in use, setting this bit to 1 reduces power consumption by the USBPHY module. Only set it to 1 when the USB module is not in use. In transitions from USBPHY standby mode to USBPHY normal operating mode, assert the reset signal for the USBPHY module for at least 1 µs before the transition.
1	PLL_RST	This bit controls resetting of the USBPHY module. 0b: The USBPHY reset is released. 1b: The USBPHY module is reset.
0	USBH_RST	Software reset to this module. Setting this bit to 1b resets this module entirely. This bit is always read as 0b. [Note] Set this bit only when the BUS Master interface of this module is not running. Access to this module becomes valid 10 CLK (internal bus clock [Bφ]) after this bit is written. 0b: Nothing occurs. 1b: Reset is issued to this module.

(5) Register Enable/Clock Gating Control Register (offset: 304h)

Abbreviated name of register: REGEN_CG_CTRL

Address:	307h								306h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	NONUSE_CLK_MSK	Reserved	HOST_CLK_MSK	PERI_CLK_MSK	Reserved			RPB_WEN	Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	305h								304h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.42 REGEN_CG_CTRL Register

Bit	Symbol	Description
31	NONUSE_CLK_MSK	This bit is used to mask the clock for the unused host or peripheral controller, depending on the setting value of the OTG_PERI bit of the COMMCTRL register. 0b: Do not mask the clock. 1b: Mask the clock. For details, see section 32.3.1.2, Specifications of NONUSE_CLK_MSK operation.
30		Reserved The write value should always be 0.
29	HOST_CLK_MSK	This bit is used to forcibly mask clock supply to the host controller. 0b: Do not mask the clock supply. 1b: Mask the clock supply. For details, see section 32.3.1.3, Specifications of PERI_CLK_MSK and HOST_CLK_MSK operations.
28	PERI_CLK_MSK	This bit is used to forcibly mask the clock supply to the peripheral controller. 0b: Do not mask the clock supply. 1b: Mask the clock supply. For details, see section 32.3.1.3, Specifications of PERI_CLK_MSK and HOST_CLK_MSK operations.
27 to 25		Reserved The write value should always be 0.
24	RPB_WEN	This is used to enable writing to the U2HC_EXT2 register (offset 360h). 0b: Writing to U2HC_EXT2 is disabled. 1b: Writing to U2HC_EXT2 is enabled.
23 to 0		Reserved The write value should always be 0.

(6) Suspend Control Register (offset: 308h)

Abbreviated name of register: SPD_CTRL

Address:	30Bh								30Ah							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R
Symbol:	SUSPENDM_ENABLE	SLEEPM_ENABLE	Reserved						WKCNNT_ENABLE	Reserved						
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	309h								308h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Symbol:	Reserved								Reserved							GLOBAL_SUSPENDM_P1
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.43 SPD_CTRL Register

Bit	Symbol	Description
31	SUSPENDM_ ENABLE	The setting of this bit is valid only in the Host mode (when the OTG_PERI bit of the COMMCTRL register is 0b). This bit is used to place USBPHY into the suspend status (the state in which PHY built-in PLL is stopped) when this module is changed to the USB Suspend mode. If this bit is set to 1b, the Suspend related bits of the OHCI/EHCI Operational registers below are set. If this module is changed to the USB Suspend mode, USBPHY is placed into the suspend status. [This function is valid for the following OHCI/EHCI Operational registers] EHCI: bit [7] (Suspend bit) of the PORTSC1 register OHCI: bit [2] (PSS bit) of the HcRhPortStatus1 register OHCI: bit [7:6] (HCFS field) of the HcControl register
30	SLEEPM_ ENABLE	The setting of this bit is valid only in the Host mode (when the OTG_PERI bit of the COMMCTRL register is 0b). This bit is used to place USBPHY into the sleep status when the LPM function is used to change to the L1 Suspend mode. (In the sleep status, the PHY built-in PLL is running, but the 60MHz clock from USBPHY is gated.) This bit is valid when "Suspend using L1" of the POTSC register is 1b. If this bit is set to 1b, an L1 transition request is issued to the device. If the request is accepted, USBPHY is placed into the sleep status after the following EHCI register bit is set to 1b. [This function is valid for the following OHCI/EHCI Operational register] EHCI: bit [7] (Suspend bit) of the PORTSC1 register
29 to 24		Reserved The write value should always be 0.
23	WKCNNT_ ENABLE	The setting of this bit is valid only in the Host mode (the OTG_PERI bit of the COMMCTRL register is 0b). If this bit is set to 1b, when a device disconnect occurs while USBPHY is in the suspend or sleep status, the suspend or sleep status is released. This bit is valid only when the SUSPENDM_ENABLE bit (bit 31) or SLEEP_ENABLE bit (bit 30) is 1b. [Note] To set 1b to the SUSPENDM_ENABLE bit (bit 31) or SLEEPM_ENABLE bit (bit 30), as the general rule, set this bit to 1b. If, in the above case, this bit is not set to 1b, the suspend or sleep status cannot be released even if a device disconnect occurs.
22 to 1		Reserved The write value should always be 0.

Table 32.43 SPD_CTRL Register

Bit	Symbol	Description
0	GLOBAL_ SUSPENDM_P1	The setting of this bit is valid regardless of the value of the OTG_PERI bit of the COMMCTRL register. This bit is used to forcibly place USBPHY into the suspend status (in which the PHY built-in PLL is stopped). If this bit is set to 1b, USBPHY is placed into the suspend status, regardless of the operating status and port status of the host controller. Note: • Do not set this bit to 1b during data transfer. We recommend that you set this bit to 1b after "stopping the EHCI/OHCI list processing" and "placing the port in the Disable status". • If the SUSPENDM_ENABLE bit (bit 31) is 1b, do not set this bit to 1b.

(7) Suspend/Resume Timer Setting Register (offset: 30Ch)

Abbreviated name of register: SPD_RSM_TIMSET

Address:	30Fh								30Eh							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	TIMER_CONNECT[15:8]								TIMER_CONNECT[7:0]							
Reset value:	0	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0
Address:	30Dh								30Ch							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	TIMER_RESUME[15:8]								TIMER_RESUME[7:0]							
Reset value:	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0

Table 32.44 SPD_RSM_TIMSET Register

Bit	Symbol	Description
31 to 16	TIMER_CONNECT [15:0]	This field indicates the timer value used by USBPHY to detect Device Connect/Disconnect of the device in the suspend status (in which the PHY built-in PLL is stopped) when the SUSPENDM_ENABLE bit of the SPD_CTRL register is set to 1b. When USBPHY is in suspend status, whether Connect/Disconnect occurs is judged by the internal bus clock (Bφ). According to the internal bus clock (Bφ) frequency, specify the setting so that this timer value becomes at least 2.5 μs. 1 bit = 1 cycle (μs) (Setting guideline) For 100 MHz: At least FAh
15 to 0	TIMER_RESUME [15:0]	This field indicates the timer value used by USBPHY to detect RemoteWakeup signal from the device in suspend status (in which the PHY built-in PLL is stopped) when the SUSPENDM_ENABLE bit of the SPD_CTRL register is set to 1b. When USBPHY is in suspend status, whether the RemoteWakeup signal or not is judged by the internal bus clock (Bφ). According to the internal bus clock (Bφ) frequency, specify the setting so that this timer value becomes at least 5 μs. 1 bit = 1 cycle (μs) (Setting guideline) For 100 MHz: At least 1F4h

(8) Overcurrent Detection/Sleep Timer Setting Register (offset: 310h)

Abbreviated name of register: OC_SLP_TIMSET

Address:	313h								312h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved			TIMER_SLEEP[8:4]					TIMER_SLEEP[3:0]				TIMER_OC[19:16]			
Reset value:	0	0	0	0	1	1	0	0	1	0	0	0	0	0	1	1
Address:	311h								310h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	TIMER_OC[15:8]								TIMER_OC[7:0]							
Reset value:	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0

Table 32.45 OC_SLP_TIMSET Register

Bit	Symbol	Description
31 to 29		Reserved The write value should always be 0.
28 to 20	TIMER_SLEEP [8:0]	This field indicates the timer value used by USBPHY when the SLEEPM_ENABLE bit of the SPD_CTRL register is set to 1b. This timer value is used to measure the time for detecting RemoteWakeup reception and Resume-K drive time during sleep status (in which the PHY built-in PLL is running, but 60MHz clock from the USBPHY is gated). When USBPHY is in sleep status, whether the RemoteWakeup signal or not is judged by the internal bus clock (Bφ). According to the internal bus clock (Bφ) frequency, specify the setting so that this timer value becomes 1 µs. 1 bit = 1 cycle (µs) (Setting guideline) For 100 MHz: 064h
19 to 0	TIMER_OC [19:0]	This field indicates the timer value used for overcurrent detection. If the overcurrent input (OVRCUR) set in this field is continuously asserted (0b) for the duration set in this register, this module determines that overcurrent occurred. According to the internal bus clock (Bφ) frequency, specify the setting so that this timer value becomes at least 1 ms. 1 bit is 1 cycle (µs). (Setting guideline) For 100 MHz and 1 ms: At least 1_86A0h

(9) SBRN_FLADJ_PORTWAKECAP Register (offset: 314h)

Abbreviated name of register: SBRN_FLADJ_PW

Address:	317h								316h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	PORTWAKECAP[15:8]								PORTWAKECAP[7:0]							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
Address:	315h								314h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R
Symbol:	FLADJ[7:0]								SBRN[7:0]							
Reset value:	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0

Table 32.46 SBRN_FLADJ_PW Register

Bit	Symbol	Description
31 to 16	PORTWAKECAP [15:0]	This field is used to mask the ports (of the connected device) that are used for Wakeup event. Operation on this field has no effect on the operation of the HOST module.
15 to 8	FLADJ[7:0]	This field adjusts the length of one micro frame by 16HS bit time unit. The initial value indicates 20h (60000d HS bit time).
7 to 0	SBRN[7:0]	This field indicates the Serial Bus Release Number. The fixed value “20h” is indicated.

(10) PORT_LPM_CTR1 Register (offset: 320h)

Abbreviated name of register: PORT_LPM_CTRL1

Address:	323h								322h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	321h								320h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved								NYET_RETRY_CNT_P1[3:0]				REMOTE WAKE_ EN_P1	SLEEP_ INT_EN_ P1	RETRY_ ENABLE_ NYET_P1	HIRD_ SEL_P1
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.47 PORT_LPM_CTRL1 Register

Bit	Symbol	Description
31 to 8		Reserved The write value should always be 0.
7 to 4	NYET_RETRY_ CNT_P1[3:0]	This field sets the number of retries that are allowed when the response from the device in the LPM transaction was NYET. The setting value of this bit is valid if the RETRY_ENABLE_NYET_P1 bit (bit 1) is 1b. 0000b: No retry. 0001b to 1111b: Retries the set number of times. (MAX: 15 retries)
3	REMOTEWAKE_ EN_P1	This bit is used to indicate the value of the RemoteWakeup bit of the LPM Token. 0b: RemoteWakeup is supported. 1b: RemoteWakeup is not supported.
2	SLEEP_INT_ EN_P1	This bit is used to enable an interrupt to occur when a status other than ACK is received in the LPM transaction. The Per-Port Change interrupt can occur. 0b: No interrupt occurs. 1b: An interrupt occurs.
1	RETRY_ENABLE_ NYET_P1	This bit is used to set the behavior of the host controller when a response from the device is NYET in an LPM transaction. 0b: No retry. 1b: Retries are made.

Table 32.47 PORT_LPM_CTRL1 Register

Bit	Symbol	Description
0	HIRD_SEL_P1	This bit sets the time for K drive for when recovered from the Sleep state. Based on the setting values of this bit and EHCI USBCMD Register bit [27:24], the K drive time is determined as follows.

USBCMD Register bit [27:24]	HIRD_SEL_P1 (Setting value of this bit) 0b	1
0000b	75 µs	50 µs
0001b	100 µs	125 µs
0010b	150 µs	200 µs
0011b	250 µs	275 µs
0100b	350 µs	350 µs
0101b	450 µs	425 µs
0110b	950 µs	500 µs
0111b	1950 µs	575 µs
1000b	2950 µs	650 µs
1001b	3950 µs	725 µs
1010b	4950 µs	800 µs
1011b	5950 µs	875 µs
1100b	6950 µs	950 µs
1101b	7950 µs	1025 µs
1110b	8950 µs	1100 µs
1111b	9950 µs	1175 µs

(11) USB 2 Host Controller Extended Function 2 Register (offset:360h)

Abbreviated name of register: U2HC_EXT2

Note: • Writing to this register is only possible while the setting of the RPB_WEN bit is 1b.
• Set bit 9 to 1b when making the initial settings for this module.

Address:	363h								362h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	361h								360h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved						DUR_CTRL	Reserved	Reserved							
Reset value:	0	0	1	1	0	0	0	1	1	0	0	1	0	0	0	1

Table 32.48 U2HC_EXT2 Register

Bit	Symbol	Description
31 to 14		Reserved The write value should always be 0.
13, 12		Reserved The write value should always be 1.
11, 10		Reserved The write value should always be 0.
9	DUR_CTRL	Set this bit to 1b when making the initial settings for this module. For the procedure for the setting, see section 32.9.2, Initialization Sequence.
8, 7		Reserved The write value should always be 1.
6, 5		Reserved The write value should always be 0.
4		Reserved The write value should always be 1.
3 to 1		Reserved The write value should always be 0.
0		Reserved The write value should always be 1.

32.2.4.5 UCOM Register

(1) Common Control Register (offset: 800h)

Abbreviated name of register: COMMCTRL

Address:	803h								802h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	OTG_PERI	Reserved							Reserved							
Reset value:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	801h								800h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R/W	R/W
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.49 COMMCTRL Register

Bit	Symbol	Description
31	OTG_PERI	This bit specifies whether this module is set to the host mode or peripheral mode. 0b: Host mode 1b: Peripheral mode
30 to 0		Reserved The write value should always be 0.

(2) OTG-BC Interrupt Status Register (offset: 804h)

Abbreviated name of register: OBINTSTA

Address:	807h								806h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W(1)	R/W(1)
Symbol:	Reserved								Reserved						DPMON CHG_STA	DMMON CHG_STA
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	805h								804h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R/W(1)	R	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)
Symbol:	Reserved									CHGDET CHG1_STA	Reserved	PDDET CHG1_STA	VBSTAINT_STA	VBSTACHG_STA	OCINT_STA	IDCHG_STA
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Table 32.50 OBINTSTA Register

Bit	Symbol	Description
31 to 18		Reserved The write value should always be 0.
17	DPMONCHG_STA	This bit is set if the DPMON bit of the LINECTRL1 register has changed. 0b: The DPMON bit of the LINECTRL1 register has not changed. 1b: The DPMON bit of the LINECTRL1 register has changed.
16	DMMONCHG_STA	This bit is set if the DMMON bit of the LINECTRL1 register has changed. 0b: The DMMON bit of the LINECTRL1 register has not changed. 1b: The DMMON bit of the LINECTRL1 register has changed.
15 to 7		Reserved The write value should always be 0.
6	CHGDETCHG1_STA	This bit is set if the setting of the CHGDETSTS bit in the BCCTRL1 register has been changed. 0b: The setting of the CHGDETSTS bit in the BCCTRL1 register has not been changed. 1b: The setting of the CHGDETSTS bit in the BCCTRL1 register has been changed.
5		Reserved The write value should always be 0.
4	PDDETCHG1_STA	This bit is set if the setting of the PDDETSTS bit in the BCCTRL1 register has been changed. 0b: The setting of the PDDETSTS bit in the BCCTRL1 register has not been changed. 1b: The setting of the PDDETSTS bit in the BCCTRL1 register has been changed.
3	VBSTAINT_STA	This bit is set if the value of the VBSTA bit of the VBCTRL register is equal to the value set to the VBLVL bit of the VBCTRL register. However, after this bit is set when the VBSTA bit and the VBLVL bit become equal, this bit might be cleared afterward. In such a case, if the value of the VBSTA bit has not changed, this bit is not be set, even if the value of the VBSTA bit is equal to the value of the VBLVL bit. Then, if the value of the VBSTA bit changes and becomes equal to the VBLVL bit again, this bit is set again. 0b: The VBSTA bit of the VBCTRL register is not equal to the value set to the VBLVL bit of the VBCTRL register. 1b: The VBSTA bit of the VBCTRL register becomes equal to the value set to the VBLVL bit of the VBCTRL register.
2	VBSTACHG_STA	This bit is set if the state of the VBSTA bit of the VBCTRL register has changed. 0b: The VBSTA bit of the VBCTRL register has not changed. 1b: The VBSTA bit of the VBCTRL register has changed.
1	OCINT_STA	This bit is set if the OVRCUR pin is asserted. 0b: The OVRCUR pin is not asserted (and remains 1b). 1b: The OVRCUR pin is asserted (and became 0b).
0	IDCHG_STA	This bit is set if the input value from the OTG_ID pin has changed. 0b: There is no change in the OTG_ID pin. 1b: There is a change in the OTG_ID pin. Note: The initial value is 1. Before using this bit, clear the status.

(3) OTG-BC Interrupt Enable Register (offset: 808h)

Abbreviated name of register: OBINTEN

Address:	80Bh								80Ah							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W
Symbol:	Reserved								Reserved						DPMONCHG_EN	DMMONCHG_EN
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	809h								808h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved								Reserved	CHGDETCHG1_EN	Reserved	PDDETCHG1_EN	VBSTAINT_EN	VBSTACHG_EN	OCINT_EN	IDCHG_EN
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.51 OBINTEN Register

Bit	Symbol	Description
31 to 18		Reserved The write value should always be 0.
17	DPMONCHG_EN	DPMONCHG_STA bit interrupt enable 0b: Interrupt via the DPMONCHG_STA bit is disabled. 1b: Interrupt via the DPMONCHG_STA bit is enabled.
16	DMMONCHG_EN	DMMONCHG_STA bit interrupt enable 0b: Interrupt via the DMMONCHG_STA bit is disabled. 1b: Interrupt via the DMMONCHG_STA bit is enabled.
15 to 7		Reserved The write value should always be 0.
6	CHGDETCHG1_EN	CHGDETCHG1_STA bit interrupt enable 0b: Interrupt via the CHGDETCHG1_STA bit is disabled. 1b: Interrupt via the CHGDETCHG1_STA bit is enabled.
5		Reserved The write value should always be 0.
4	PDDETCHG1_EN	PDDETCHG1_STA bit interrupt enable 0b: Interrupt via the PDDETCHG1_STA bit is disabled. 1b: Interrupt via the PDDETCHG1_STA bit is enabled.
3	VBSTAINT_EN	VBSTAINT_STA bit interrupt enable 0b: Interrupt via the VBSTAINT_STA bit is disabled. 1b: Interrupt via the VBSTAINT_STA bit is enabled.
2	VBSTACHG_EN	VBSTACHG_STA bit interrupt enable 0b: Interrupt via the VBSTACHG_STA bit is disabled. 1b: Interrupt via the VBSTACHG_STA bit is enabled.
1	OCINT_EN	OCINT_STA bit interrupt enable 0b: Interrupt via the OCINT_STA bit is disabled. 1b: Interrupt via the OCINT_STA bit is enabled.
0	IDCHG_EN	IDCHG_STA bit interrupt enable 0b: Interrupt via the IDCHG_STA bit is disabled. 1b: Interrupt via the IDCHG_STA bit is enabled.

(4) VBUS Control Register (offset: 80Ch)

Abbreviated name of register: VBCTRL

Address:	80Fh								80Eh							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R
Symbol:	Reserved		VBSTA	Reserved					Reserved		VBLVL	Reserved				
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Address:	80Dh								80Ch							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved								Reserved			VGPUO	Reserved			VBOUT
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.52 VBCTRL Register

Bit	Symbol	Description
31, 30		Reserved The write value should always be 0.
29	VBSTA	This bit indicates the voltage level of VBUS. Input level from the VBUSIN pin is indicated as it is.
28 to 22		Reserved The write value should always be 0.
21	VBLVL	This bit is used to detect (by an interrupt) that VBUS reaches a specific level. When the value of the VBUSIN pin becomes equal to the value set to this field, the VBSTAINT_STA bit of the OBINTSTA register is set. If the bit is not masked, an interrupt (caused by the VBSTAINT_STA bit of the OBINTSTA register) occurs.
20 to 17		Reserved The write value should always be 0.
16		Reserved The write value should always be 1.
15 to 5		Reserved The write value should always be 0.
4	VGPUO	The level corresponding to the inverse of this bit (in terms of positive logic) is output from the OTG_EXICEN pin. This bit is used, for example, for control of the external power IC.
3 to 1		Reserved The write value should always be 0.
0	VBOUT	This bit is one of the VBUSEN control bits. This bit is used to assert VBUS by controlling the external power IC. 0b: VBUS output disable 1b: VBUS output enable If overcurrent occurs, this bit is automatically cleared to 0.

(5) Line Control Port 1 Register (offset: 810h)

Abbreviated name of register: LINECTRL1

Address:	813h								812h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved								Reserved		DSDP[1:0]		DPRPD_EN	DP_RPD	DMRPD_EN	DM_RPD
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	811h								810h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved				DP_MON	DM_MON	Reserved	ID_MON
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.53 LINECTRL1 Register

Bit	Symbol	Description
31 to 22		Reserved The write value should always be 0.
21, 20	DSDP[1:0]	These bits are set on transitions to and returning from the deep standby state in the DCP mode. Set the bits by following the procedure described in section 33.9.16.4, Flowchart of resumption from deep standby in response to reception of the resume signal or section 32.6.1.2, When the host controller is to be used in the DCP mode. If you do not so, operation is not guaranteed.
19	DPRPD_EN	This bit enables DP_RPD (bit 18) to control USB bus (DP) 15 kΩ Pulldown resister. 0b: Control of DP-side 15 kΩ Pulldown resister by DP_RPD (bit 18) is disabled. 1b: Control of DP-side 15 kΩ Pulldown resister by DP_RPD (bit 18) is enabled.
18	DP_RPD	This bit controls USB bus (DP) 15 kΩ Pulldown resister when DPRPD_EN (bit 19) = 1b. 0b: DP-side 15 kΩ Pulldown resister is OFF. 1b: DP-side 15 kΩ Pulldown resister is ON.
17	DMRPD_EN	This bit enables DM_RPD (bit 16) to control USB bus (DM) 15 kΩ Pulldown resister. 0b: Control of DM-side 15 kΩ Pulldown resister by DM_RPD (bit 16) is disabled. 1b: Control of DM-side 15 kΩ Pulldown resister by DM_RPD (bit 16) is enabled.
16	DM_RPD	This bit controls USB bus (DM) 15 kΩ Pulldown resister when DMRPD_EN (bit 17) = 1b. 0b: DM-side 15 kΩ Pulldown resister is OFF. 1b: DM-side 15 kΩ Pulldown register is ON.
15 to 4		Reserved The write value should always be 0.
3	DPMON	This bit indicates the value of USB bus DP.
2	DMMON	This bit indicates the value of USB bus DM.
1		Reserved The write value should always be 0.
0	IDMON	This bit indicates the value of the OTG_ID pin.

(6) BC Control Port 1 Register (offset: 820h)

Abbreviated name of register: BCCTRL1

Address:	823h								822h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
Address:	821h								820h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Symbol:	Reserved						PDDET STS	CHGD ET STS	Reserved		DCP MODE	VDM SRCE	IDP SINKE	VDP SRCE	IDM SINKE	IDP SRCE
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.54 BCCTRL1 Register

Bit	Symbol	Description
31 to 26		Reserved The write value should always be 0.
25, 24		Reserved The write value should always be 1.
23 to 10		Reserved The write value should always be 0.
9	PDDETSTS	This bit indicates the USBPHY Portable Device Detect signal state.
8	CHGDETSTS	This bit indicates the USBPHY Charging Downstream Port Detect signal state.
7, 6		Reserved The write value should always be 0.
5	DCPMODE	If USBPHY is used as DCP (Dedicated Charging Port), this bit is set to 1b.
4	VDMSRCE	This bit controls the USBPHY built-in VDM_SRC circuit. If this bit is set to 1b, VDM_SRC goes ON, and the DM pin is driven.
3	IDPSINKE	This bit controls the USBPHY built-in Portable Device Detect circuit. If this bit is set to 1b, Portable Device detection is enabled.
2	VDPSRCE	This bit controls the USBPHY built-in VDP_SRC circuit. If this bit is set to 1b, VDP_SRC goes ON, and the DP pin is driven.
1	IDMSINKE	This bit controls the USBPHY built-in Charging Downstream Port Detect circuit. If this bit is set to 1b, Charging Downstream Port detection is enabled.
0	IDPSRCE	This bit controls the USBPHY built-in IDP_SRC circuit. If this bit is set to 1b, IDP_SRC goes ON, and the DP pin is driven.

(7) CC STATUS Register (offset: 840h)

Abbreviated name of register: CC_STATUS

Address:	843h								842h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R
Symbol:	CC_INT_SEL	Reserved	CC_LVL_CLR	CC_LVL[3]	CC_LVL[2]	CC_LVL[1]	CC_LVL[0]	CC_LVL_EN	Reserved							
Reset value:	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0
Address:	841h								840h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R
Symbol:	Reserved								Reserved	Reserved	CC_PERI_STA	CC_LVL_STA	CC1_RD	CC1_RA	CC2_RD	CC2_RA
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.55 CC_STATUS Register

Bit	Symbol	Description
31	CC_INT_SEL	This bit specifies the condition for the CC_INT interrupt. 0: The CC_INT interrupt occurs whenever a peripheral device is connected.*1 1: Whether a CC_INT interrupt occurs and its level depend on the settings of the CC_LVL and CC_LVL_EN bits.
30		Reserved The write value should always be 0.
29	CC_LVL_CLR	CC_LVL_STA flag clear 0: Do not clear CC_LVL_STA. 1: Clear CC_LVL_STA.*2
28	CC_LVL[3]	If “CC_INT_SEL = 1” is set, this bit specifies the level of interrupt occurrence. Specify the CC1_Rd level.
27	CC_LVL[2]	If “CC_INT_SEL = 1” is set, this bit specifies the level of interrupt occurrence. Specify the CC1_Ra level.
26	CC_LVL[1]	If “CC_INT_SEL = 1” is set, this bit specifies the level of interrupt occurrence. Specify the CC2_Rd level.
25	CC_LVL[0]	If “CC_INT_SEL = 1” is set, this bit specifies the level of interrupt occurrence. Specify the CC2_Ra level.
24	CC_LVL_EN	CC_LVL enable setting 0: If “CC_INT_SEL = 1” is set, and input state of the CCn_Rd and CCn_Ra (n = 1, 2) pins is equal to the CC_LVL[3:0] value, an interrupt does not occur. 1: If “CC_INT_SEL = 1” is set, and input state of the CCn_Rd and CCn_Ra pins is equal to the CC_LVL[3:0] value, an interrupt occurs.
23 to 7		Reserved The write value should always be 0.
6		Reserved The read value is not stable. The write value should always be 0.
5	CC_PERI_STA	This bit indicates whether a peripheral device is connected.*3 0: A peripheral device is not connected. 1: A peripheral device is connected.*1
4	CC_LVL_STA	This bit indicates the relationship between the setting value of the CC_LVL bit and the pin status. 0: The status of the CCn_Rd and CCn_Ra pins is not equal to the setting value of CC_LVL[3:0]. 1: The status of the CCn_Rd and CCn_Ra pins is equal to the setting value of CC_LVL[3:0].*4
3	CC1_RD	This bit indicates the CC1_Rd pin status.
2	CC1_RA	This bit indicates the CC1_Ra pin status.
1	CC2_RD	This bit indicates the CC2_Rd pin status.
0	CC2_RA	This bit indicates the CC2_Ra pin status.

Note 1. When the settings of the CC1_Rd, CC1_Ra, CC2_Rd, and CC2_Ra bits are as in Table 32.56, the connection of peripheral devices is detected.
For details about the values to be input to the CCn_Rd and CCn_Ra pins, see section 32.10.1, Notes on Using the CCn_Ra and CCn_Rd Pins.

Note 2. This bit is not automatically cleared from 1 to 0, so write 0 to this bit after CC_LVL_STA clear.
Note 3. Regardless of the CC_INT_SEL setting, if the input of the CCn_Rd and CCn_Ra pins is equal to Table 32.56, 1 is set.
Note 4. Regardless of the CC_INT_SEL setting, if the value set for CC_LVL is equal to the status of the CCn_Rd and CCn_Ra pins, 1 is set.

Table 32.56 Conditions for Detecting the Connection of a Peripheral Device

CC1_Rd	CC1_Ra	CC2_Rd	CC2_Ra
0	1	1	1
1	1	0	1
0	1	0	0
0	0	0	1

(8) PHYCLK CTRL Register (offset: 844h)

Abbreviated name of register: PHYCLK_CTRL

Address:	847h								846h							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	845h								844h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Symbol:	Reserved								Reserved							UCLK SEL
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.57 PHYCLK_CTRL Register

Bit	Symbol	Description
31 to1		Reserved The write value should always be 0.
0	UCLKSEL	This bit selects the clock supplied to USBPHY. 0: EXTAL clock is selected. 1: USB_X1 clock is selected.

(9) PHYIF CTRL Register (offset: 848h)

Abbreviated name of register: PHYIF_CTRL

Address:	84Bh								84Ah							
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Symbol:	Reserved								Reserved							
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Address:	849h								848h							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Symbol:	Reserved								Reserved							FIXPHY
Reset value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 32.58 PHYIF_CTRL Register

Bit	Symbol	Description
31 to 1		Reserved The write value should always be 0.
0	FIXPHY	This bit is to be set when the USB module is not in use or on recovery from the deep standby state. Set the bit by following the descriptions in section 33.9.16.1, Setting at transition to deep standby and section 52.4.3, Usage Notes when USB 2.0 Host/Function Modules are Not to be Used. If you do not so, operation is not guaranteed.

32.3 Clock Signals

32.3.1 Clock Gating Specifications

32.3.1.1 Overview of clock gating

Because this module has a feature of switching the host controller and peripheral controller, clock supply to the unused controller might not be necessary.
Therefore, three clock gating control bits are allocated in the Register Enable/Clock Gating Control Register for the purpose of reducing power consumption by implementing clock gating for the "circuits that do not require clock supply temporarily."
The clock gating control bits can be controlled to gate the clock that is supplied to the host controller or the peripheral controller.

[Target Register]
Register Enable/Clock Gating Control Register (offset:304h)

[Functional specification]

bit	Symbol	Functional specification
31	NONUSE_CLK_MSK	Gating clocks for unused host controller or peripheral controller.
29	HOST_CLK_MSK	Gating clocks for host controller.
28	PERI_CLK_MSK	Gating clocks for peripheral controller.

32.3.1.2 Specifications of NONUSE_CLK_MSK operation

This function automatically gates the clock to the unused host controller or the unused peripheral controller.
This function is enabled if the NONUSE_CLK_MSK bit (bit 31) is set to 1b.

If it is no problem whether the clock supply to the unselected function is stopped, use this function.

The following table shows the operating specifications of this function, based on the setting of the OTG_PERI bit (bit 31) of the Common Control Register (offset: 800h).

			Host/Peripheral Switching Setting	Gating Target	
Clock Gating Register Setting			Register		
NONUSE_CLK_MSK	HOST_CLK_MSK	PERI_CLK_MSK	OTG_PERI	Host Controller	Function Controller
1	0	0	0		○
			1	○	

[Note on using the NONUSE_CLK_MSK bit]
If the NONUSE_CLK_MSK bit is used, as the general rule, set the HOST_CLK_MSK/PERI_CLK_MSK bit to 0b.
If the HOST_CLK_MSK/PERI_CLK_MSK bit is 1b, the effect of clock gating becomes logical OR of each bit.

32.3.1.3 Specifications of PERI_CLK_MSK and HOST_CLK_MSK operations

This function forcibly gates the clock to the host controller and function controller.
The following table shows the operating specifications of this function.

Clock Gating Register Setting			Host/Peripheral Switching Setting	Gating Target	
			Register		
NONUSE_CLK_MSK	HOST_CLK_MSK	PERI_CLK_MSK	OTG_PERI	Host Controller	Function Controller
0	1	0	—	○	
	0	1	—		○
	1	1	0	○	○

Note: If the HOST_CLK_MSK/PERI_CLK_MSK bit is used, as the general rule, set the NONUSE_CLK_MSK bit to 0b.
If the NONUSE_CLK_MSK bit is set t0 1b, the effect of clock gating becomes logical OR of each bit.

32.4 Interrupt Sources

32.4.1 Interrupt Signals

This module has the six interrupt signals listed below. The logical OR of these six interrupt signals is sent to the interrupt controller as USBHIn (n = 0, 1). For details about the occurrence source of USBHIn (n = 0, 1) interrupt, check the corresponding status register.
Use all these interrupt signals as level interrupt signals.

Interrupt Source Name	Interrupt Type	Pulse/ Level	Active Level
U2H_INT	BUS Master interrupt signal. This signal is asserted when a bus error occurs in the BUS Master. Interrupt control is performed by the AHB Bridge Register.	Level	H
U2H_OHCI_INT	OHCI interrupt signal This signal is asserted during FS/LS transfer, when data transfer finishes or when the change of the USB bus state is detected. Interrupt control is performed by the OHCI Operational Register.	Level	H
U2H_EHCI_INT	EHCI interrupt signal. This signal is asserted during HS transfer, when data transfer finishes or when the change of the USB bus state is detected. Interrupt control is performed by the EHCI Operational Register.	Level	H
U2H_WAKEON_INT	EHCI Wakeup interrupt signal. This signal is asserted by an EHCI Wakeup event. Interrupt control is performed by the EHCI Operational Register.	Level	H
U2H_OBINT	OTG/Battery Charging interrupt signal. This signal is asserted by OTG or Battery Charging related event. Interrupt control is performed by the UCOM2 Register.	Level	H
CC_INT	CC pin interrupt signal	Level	H

32.4.2 Interrupt Sources and Control

32.4.2.1 U2H_INT assertion source and control

[Interrupt enable control by register]
Assert the interrupt enable bit in the register below. The interrupt signal is asserted when an interrupt source occurs.
AHB Bridge Register INT_ENABLE Register (offset:200h) bit[0] (AHB_INTEN)

[Interrupt source]
A bus error (MHRESP = 1b) occurs in the AHB master.

[Clearing the interrupt]
To clear the interrupt, write 1b to the relevant bit in the register below.
AHB Bridge Register INT_STATUS Register (offset:204h) bit[0] (AHB_INT)

32.4.2.2 U2H_OHCI_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset:200h) bit[1] (USBH_INTAEN)

OHCI Operational Register HcInterruptEnable Register (offset:010h) bit[31], bit[6:0]*

Note: * Enable the bit required as the assertion source.

[Interrupt source]

		Registers That Require Interrupt Enable Setting							
		USBH_INTAEN	HcInterruptEnable						
			bit [31]	bit [6]	bit [5]	bit [3]	bit [2]	bit [1]	bit [0]
Interrupt Source			MIE	RHSCE	FNOE	ROE	SFE	WDHE	SOE
1	Device connection is detected.	○	○	○					
2	Device disconnection is detected.	○	○	○					
3	Port power is OFF (excluding overcurrent detection).	○	○	○					
4	Babble error is detected during a USB transfer.	○	○	○					
5	Resume is complete.	○	○	○					
6	Overcurrent is detected.	○	○	○					
7	Bus reset is complete.	○	○	○					
8	When the HcRhDescritorB Register DR bit is 1b, OHCI becomes “USB Operational” (HCFS[1:0] bit = 10b) or “USB Suspend” (HCFS[1:0] bit = 11b).	○	○	○					
9	When no device is connected (CCS bit = 0b), 1 is written to bit [0] (Clear Port Enable) of the OHCI HcRhPort Status register.	○	○	○					
10	When no device is connected (CCS bit = 0b), 1 is written to bit [1] (Set Port Enable) of the OHCI HcRhPort Status register.	○	○	○					
11	When no device is connected (CCS bit = 0b), 1 is written to bit [2] (Set Port Suspend) of the OHCI HcRhPort Status register.	○	○	○					
12	When no device is connected (CCS bit = 0b), 1 is written to bit [3] (Clear Suspend Status) of the OHCI HcRhPort Status register.	○	○	○					
13	When no device is connected (CCS bit = 0b), 1 is written to bit [4] (Set Port Reset) of the OHCI HcRhPort Status register.	○	○	○					
14	When no device is connected (CCS bit = 1b), the port power is turned off.	○	○	○					
15	The MSB of bit [15:0] (Frame Number) of the HcFmNumber register has changed.	○	○		○				
16	RemoteWakeup signal (Resume signal) is detected from a device.	○	○			○			
17	HccaFrameNumber is updated. (Almost the same meaning as SOF is sent.)	○	○				○		
18	A transfer finishes (including an error), and the host module updated HccaDoneHead.	○	○					○	
19	USB schedule overrun occurred for the frame.	○	○						○

[Clearing the interrupt]

To clear the interrupt, write 1b to the bit corresponding to the interrupt source in the register below to clear the interrupt.

OHCI Operational Register HcInterruptStatus Register (offset:00Ch) bit[6:0]

32.4.2.3 U2H_EHCI_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset:200h) bit[2] (USBH_INTBEN)

EHCI Operational Register USBINTR Register (offset:128h) bit[17:16], bit[5:0]*

* Enable the bit required as the assertion source.

Also, control the relevant bit(s) in the register below as required.

EHCI Operational Register USBCMD Register (offset:120h) bit[15], bit[6]

[Interrupt source]

		Registers That Require Interrupt Enable Setting								
			USBCMD		USBINTR					
			bit [15]	bit [6]	bit [17]	bit [16]	bit [5]	bit [3]	bit [2]	bit [1]	bit [0]
	Interrupt Source	USBH_INTBEN	Per-Port Change Event	Doorbell	Port-1 Change Event		Async Advance	Frame List Rollover	Port Change	USB ERRINT	USB INT
1	Device connection is detected.	○							○		
2	Device disconnection is detected.	○							○		
3	Overcurrent is detected.	○							○		
4	RemoteWakeup signal (Resume Signal) is detected from a device.	○							○		
5	Babble status of the USB bus is detected.	○							○		
6	USB transfer with "qTD IOC = 1b" normally finishes.	○									○
7	Short packet is received.	○									○
8	USB transfer finished with an error. (Retry transfer failed three times. A bubble error was detected. STALL was received.)	○								○	
9	The QH processing normally finished while the USBCMD Register bit [6] (Interrupt on Async Advance Doorbell) is 1b.	○		○			○				
10	The FRINDEX Register FrameIndex bit returned from the maximum value to 000h (rollover detected).	○						○			
11	The Port Change Detect event (interrupt source 1 to 5) was detected.	○	○			○					

[Clearing the interrupt]

To clear the interrupt, write 1b to the bit corresponding to the interrupt source in the register below to clear the interrupt.

EHCI Operational Register USBSTS Register (offset:124h) bit[17:16], bit[5:0]

32.4.2.4 U2H_WAKEON_INT assertion source and control

[Interrupt enable control by register]
Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.
AHB Bridge Register INT_ENABLE Register (offset:200h) bit[4] (WAKEON_INTEN)
EHCI Operational Register PORTSC[1:2] Register (offset:164h/168h) bit[22:20]*

Note: * Enable the bit required as the assertion source.

[Interrupt source]

		Registers That Require Interrupt Enable Setting			
			PORTSC		
Interrupt Source		WAKEON_INTEN	bit [22] WKOC_E	bit [21] WKDSCNNT_E	bit [20] WKCNNT_E
1	Device connection is detected.	○			○
2	Device disconnection is detected.	○		○	
3	Overcurrent is detected.	○	○		
4	RemoteWakeup signal (Resume Signal) is detected from a device.	○			

[Clearing the interrupt]
To clear the interrupt, write 1b to the relevant bit in the register below.
AHB Bridge Register INT_STATUS Register (offset:204h) bit[4] (WAKEON_INT)

32.4.2.5 U2H_OBINT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset:200h) bit[3] (UCOM_INTEN)

UCOM Register OTG-BC Interrupt Enable Register (offset:808h) bit[27:24], bit[18:16], bit[12:8], bit[3:0]*

Note: * Enable the bit required as the assertion source.

[Interrupt source]

		Registers That Require Interrupt Enable Setting		
			OTG-BC Interrupt Enable	
			bit [17]	bit [16]
	Interrupt Source	UCOM_INTEN	DPMONCHG_EN	DMMONCHG_EN
1	The DP pin has changed.	○	○	
2	The DM pin has changed.	○		○

		Registers That Require Interrupt Enable Setting		
			OTG-BC Interrupt Enable	
			bit [6]	bit [4]
	Interrupt Source	UCOM_INTEN	CHGDETCHG1_EN	PDDETCHG1_EN
1	The portable device detection signal has changed.	○		○
2	The charging port detection signal has changed.	○	○	

		Registers That Require Interrupt Enable Setting				
			OTG-BC Interrupt Enable			
			bit [3]	bit [2]	bit [1]	bit [0]
	Interrupt Source	UCOM_INTEN	VBSTAINT_EN	VBSTACHG_EN	OCINT_EN	IDCHG_EN
1	It is detected that the value of the VBUSIN pin becomes equal to the value set to bit [21] (VBLVL) of the VBUS Control Register.	○	○			
2	The VBUSIN pin has changed.	○		○		
3	Overcurrent is detected. (The change of the OVRCUR pin from 1 to 0b is detected.)	○			○	
4	The OTG_ID pin has changed.	○				○

[Clearing the interrupt]

To clear the interrupt, write 1b to the bit corresponding to the interrupt source in the register below to clear the interrupt.

UCOM Register OTG-BC Interrupt Status Register (offset:804h) bit[27:24], bit[18:16], bit[12:8], bit[3:0]

32.4.2.6 CC_INT assertion source and control

[Register interrupt enable control and interrupt source]

By the setting of bit [31] of the CC_STATUS register, you can change the interrupt source.

When bit [31] = 0: If the conditions in Table 32.56, Conditions for Detecting the Connection of a Peripheral Device are satisfied, an interrupt occurs.

When bit [31] = 1: According to the setting value of bit [28:24], an interrupt occurs.

Note: An unintended interrupt (CC_INT) might occur because of chattering of the CCn_Rd and CCn_Ra (n = 1, 2) pins. If an interrupt occurs, read the input status of the CCn_Rd and CCn_Ra pins, and confirm that they are preferable values. Then, start the interrupt processing.

[Interrupt clear]

When bit [31] = 0: If the input values of the CCn_Rd and CCn_Ra pins do not satisfy the conditions in Table 32.56, Conditions for Detecting the Connection of a Peripheral Device, an interrupt is cleared.

When bit [31] = 1: An interrupt is cleared by setting bit [29] to 1.

32.4.3 Timing of De-asserting Interrupt Signals

After the register access to clear an interrupt source, it may take time to begin clearing the interrupt triggered by the interrupt source. Therefore, take a measure to prevent the false recognition of interrupts during the period from the end of the register clear access until the next interrupt is recognized.

32.5 Power-Saving Function

This module controls power consumption by using the following two methods:

1. Controlling the SUSPENDM/SLEEPM pin of USBPHY
2. Using clock gating to stop the clock to the host controller or peripheral controller.

32.5.1 Controlling the SUSPENDM and SLEEPM Pins of the USBPHY

You can expect the following power-saving effects by asserting the SUSPENDM and SLEEPM pins of the USBPHY:

- Reducing the power consumption by the USBPHY
- Reducing the power consumption by the host core by stopping the clocks from the USBPHY

As described at section 32.1.2.5, Suspend extension function, by default, the SUSPENDM and SLEEPM pins of the USBPHY are not asserted even when EHCI and OHCI are put into the Suspended state.
See section 32.2.4.4, (6), Suspend Control Register (offset: 308h) and control the relevant registers appropriately.

32.5.2 Controlling the Clock-Gating Function

See section 32.2.4.4, (5), Register Enable/Clock Gating Control Register (offset: 304h) and section 32.3.1, Clock Gating Specifications", then control the relevant registers appropriately.

32.6 Battery Charging

32.6.1 Support of charging port

The charging port refers to a port that supplies power in compliance with the Battery Charging Specification. The charging port is usually installed on the host controller.
Charging ports can be generally classified by function as described below.

Table 32.59 Function of Charging Port

Type	Function
CDP (Charging Downstream Port)	Downstream port that can supply power in compliance with the Battery Charging Specification. This type of charging port detects a portable device, and after the handshake (for Battery Charging) finishes, it proceeds to the usual device connect sequence (and operates as the usual host).
DCP (Dedicated Charging Port)	Port that provides only power supply function in compliance with the Battery Charging Specification. This type of charging port does not operate as the usual host.
SDP (Standard Downstream Port)	Standard Downstream Port that is not compliant with the Battery Charging Specification. This type of charging port supplies power in the range conforming to the conventional USB 2.0 standard, and operates as the usual host.

Examples of control when the host controller is to be used as a charging port are described on the following pages.

32.6.1.1 When the host controller is to be used in the CDP mode

An example of the flow of control when the host controller is to be used in the CDP mode is shown below.

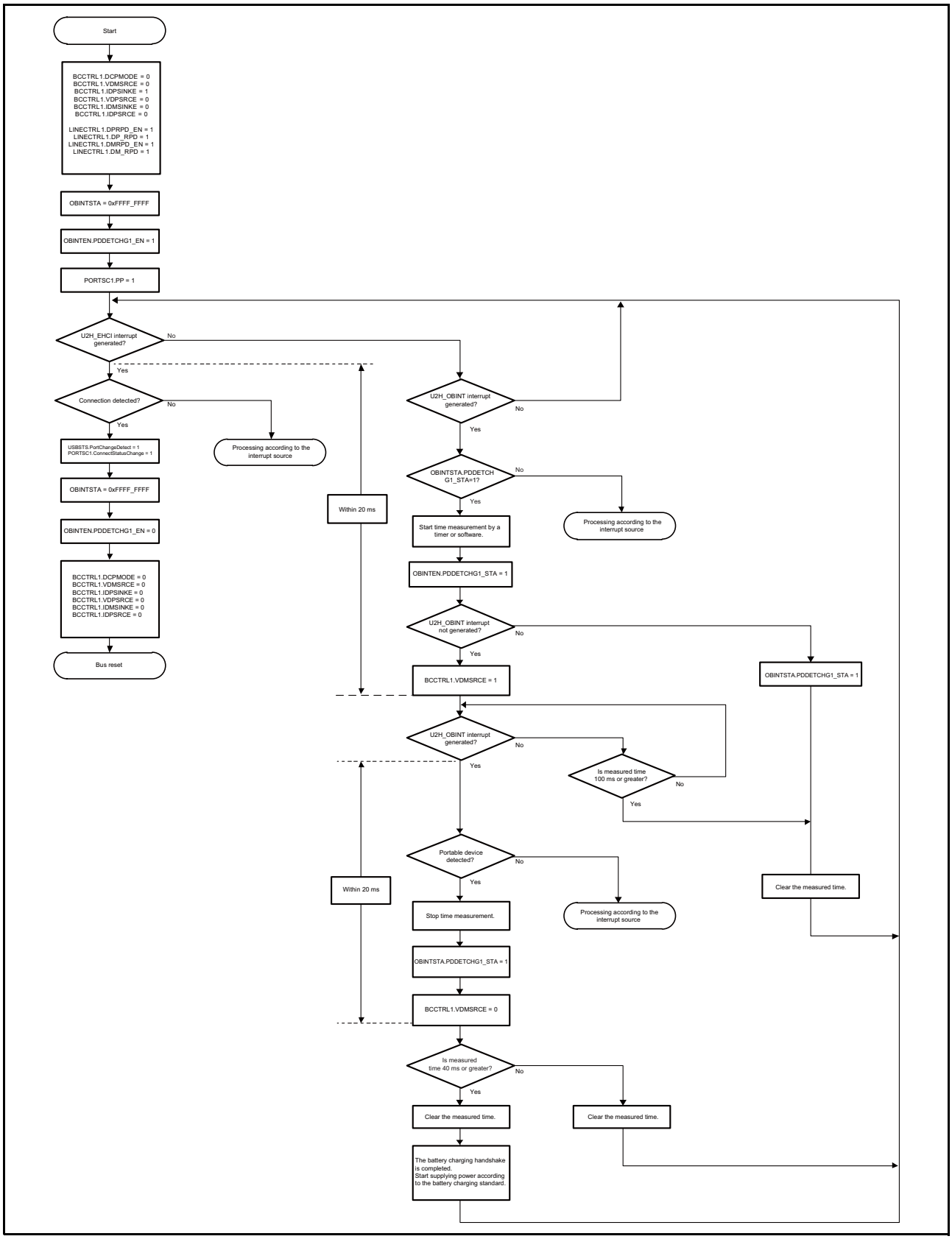


Figure 32.1 Example of the Control Flow in CDP Mode

32.6.1.2 When the host controller is to be used in the DCP mode

When the host controller is to be used in the DCP mode, make the following settings.
BCCTRL1 = 0x0300_0020
LINECTRL1 = 0x002A_0000

32.6.2 Support of portable device

A portable device refers to a device that is supplied power (or that requests power supply) in compliance with the Battery Charging Specification. The portable port is usually installed in the peripheral controller.
An example of the flow of control when the peripheral controller is in use as a portable device is shown on the next page.

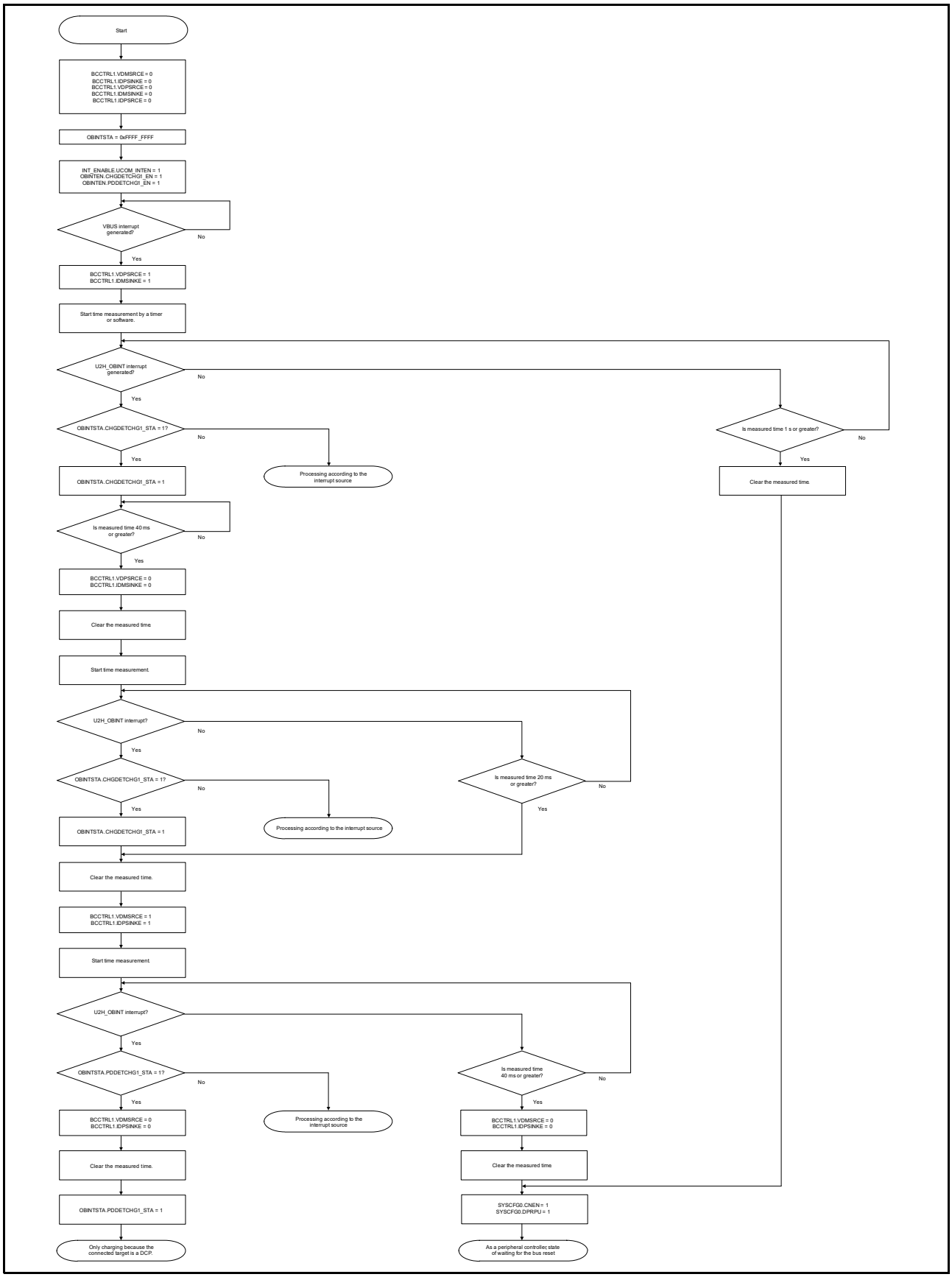


Figure 32.2 Example of the Control Flow for Portable Devices

32.7 Bus Master

32.7.1 Functional specifications of the bus master

32.7.1.1 Supported bus master functions

Function as the BUS MASTER	Status
Residual burst after an error response is received	The transfer is not stopped.
1 Kbyte boundary processing	The fixed length burst (INCRx) across 1 Kbyte boundary is not performed.

32.7.1.2 Issuing requests for different types of bus transfer

MHTRANS[1:0]	MHSIZE[2:0]	MHWRITE	MHBURST[2:0]	Reply	Remarks
IDLE (00b)	—	—	—	—	—
BUSY (01b)	—	—	—	—	Not issued.
NONSEQ (10b)	32-bit (010b)	WRITE	SINGLE INCR4 INCR8 INCR16	OKAY/ ERROR	32-bit transfer is issued. A response error is reported by an interrupt, and the transfer is not stopped.
		READ			
	8-bit (000b) 16-bit (001b)	WRITE	SINGLE	OKAY/ ERROR	8-bit/16-bit transfer is issued (SINGLE only). A response error is reported by an interrupt.
	other than the above	—	—	—	Not issued.
SEQ (11b)	32-bit (010b)	WRITE	INCR4 INCR8 INCR16	OKAY/ ERROR	32-bit transfer is issued. A response error is reported by an interrupt, and the transfer is not stopped.
		READ		OKAY/ ERROR	
	other than the above	—	—	—	Not issued.

32.7.1.3 Supported responses

Response type	Response	Remarks
OKAY	Enable	Supported.
ERROR	Enable	A response error is reported by an interrupt, and termination of transfer (Early Burst Termination) is not performed.

32.7.1.4 Protection control information

The value of MHPROT[3:0] can be set in the PROT_TYPE bits (bits [15:12]) in the AHB_BUS_CTR Register (offset: 208h).

Also, when the PROT_MODE bit (bit 8) in the AHB_BUS_CTR Register is set, only the last data transfer in an EHCI/OCHI DMA transfer can be handled as a non-buffered transfer, and other transfers can be handled as buffered transfers.

32.7.1.5 Maximum burst length

The maximum burst length can be selected from SINGLE, INCR4, INCR8, and INCR16 by using the MAX_BURST_LEN bits (bit[1:0]) in the AHB_BUS_CTR Register (offset: 208h). The maximum burst length is common to reading and writing.

32.7.1.6 Boundary of transfer data

The value of MHADDR[31:0] bits does not exceed 1 KB boundary during a burst transfer. Also, you can change the address boundary for burst transfer to 16, 32, or 64 bytes by writing a value to the ALIGN_ADDRESS bits (bit[5:4]) in the AHB_BUS_CTR Register (offset: 208h).

32.7.1.7 Start address of fixed-length INCR burst transfer

The following table lists the values of the lower bits of MHADDR to be applied when a fixed-length INCR burst transfer starts.

ALIGN_ADDRESS Setting	Fixed length Start Address of INCR Burst		
	INCR4	INCR8	INCR16
00b (Aligned at the 1-Kbyte boundary)	MHADDR[9:0] = 000h	MHADDR[9:0] = 000h	MHADDR[9:0] = 000h
	004h	004h	004h
	008h	008h	008h
	00Ch	00Ch	00Ch
	010h	010h	010h
	:	:	:
	3D0h	3C8h	3B8h
	3D4h	3CCh	3BCh
	3D8h	3D0h	3C0h
	3DCh	3D4h	
	3E0h	3D8h	
	3E4h	3DCh	
	3E8h	3E0h	
	3ECh		
	3F0h		
01b (Aligned at the 16-byte boundary)	MHADDR[3:0] = 0h	-- (Not issued)	-- (Not issued)
10b (Aligned at the 32-byte boundary)	MHADDR[4:0] = 00h	MHADDR[4:0] = 00h	-- (Not issued)
	04h		
	08h		
	0Ch		
	10h		
11b (Aligned at the 64-byte boundary)	MHADDR[5:0] = 00h	MHADDR[5:0] = 00h	MHADDR[5:0] = 00h
	04h	04h	
	08h	08h	
	0Ch	0Ch	
	10h	10h	
	14h	14h	
	18h	18h	
	1Ch	1Ch	
	20h	20h	
	24h		
	28h		
	2Ch		
	30h		

32.8 Overcurrent Control and VBUS Control

32.8.1 OVRCUR/VBUSEN pin

Overcurrent detection on the USB port and port power (VBUS) control are performed by the external power IC connected to this module.

This module pin	Input/Output	Level	Description
OVRCURI	Input	L	Overcurrent status was detected.
		H	Overcurrent status was not detected.
VBUSEN	Output	L	Port Power (VBUS) OFF
		H	Port Power (VBUS) ON

32.8.2 Overcurrent detection timer setting

This module detects overcurrent when the OVRCUR pin remains asserted (0b) for a set period.
The period over which assertion of the OVRCUR pin is required ("vercurrent detection time") can be set in the following register.

Register	Overcurrent detection/sleep timer setting register (offset: 310h)
Bits	TIMER_OC[19:0]
Initial value	3_0D40h

The overcurrent detection time can be converted from the setting value of the above register, taking that one bit equal to the internal bus clock (Bφ) cycle.
Therefore, set the value of the above register at initial configuration, considering "the internal bus clock (Bφ) frequency to be used" and "the overcurrent detection time you want to specify".

32.8.3 Port Power (VBUS) control specifications

VBUSEN can be controlled by the Port Power bit of the EHCI/OHCI Operational register or by the VBOUT bit of the VBUS Control Register. Which of the above bit controls VBUSEN is determined by the PMODEPFS register on the general input/output port.
The following table describes control examples by the Port Power bit of the EHCI/OHCI Operational register.

Situation		Register		bit
EHCI control		PORTSC1 (offset: 164h)		bit 12 (PP)
OHCI control	Global control*1	ON setting	HcRhStatus Register (offset: 050h)	bit 16 (Set Global Power)
		OFF setting		bit 0 (Clear Port Status)
	Selective control*2	ON setting	HcRhPortStatus1 Register (offset: 054h)	bit 8 (Set Port Power)
		OFF setting		bit 9 (Clear Port Power)

Note 1. Global control refers to the status in which the register settings are as follows:
HcRhDescriptorA Register (offset: 048h) bit 8 (PSM) = 0b
or
HcRhDescriptorA Register (offset: 048h) bit 8 (PSM) = 1b and HcRhDescriptorB Register (offset: 04Ch) bit 17 (PPCM[1]) = 0b

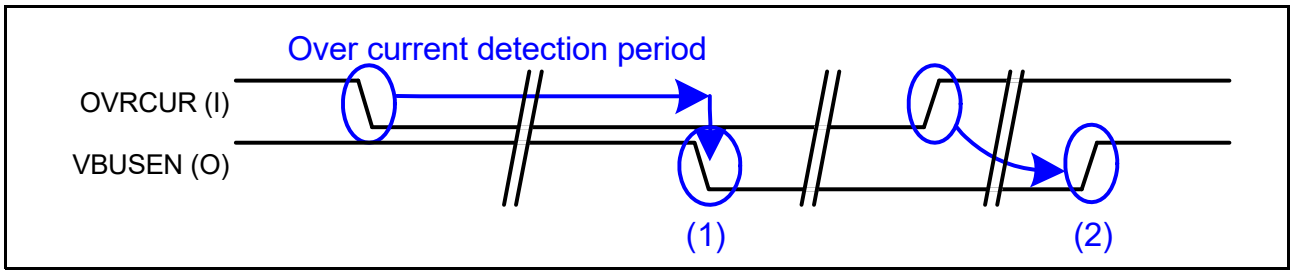
Note 2. Selective control refers to the status in which the register settings are as follows:
HcRhDescriptorA Register (offset: 048h) bit 8 (PSM) = 1b and HcRhDescriptorB Register (offset: 04Ch) bit 17 (PPCM[1]) = 1b

However, if the register settings are as follows, the VBUSEN pin is always asserted (1b), and the Port Power (VBUS) becomes ON, regardless of the OVRCUR pin's status.
Specify the settings as necessary (for example, your system needs that VBUS is always ON).

EHCI Operational	OHCI Operational				
HCSPARAMS (offset: 104h)	HcRhDescriptorA (offset: 048h)			HcRhDescriptorB (offset: 04Ch)	Pin Operation
PPC (bit 4)	NOCP (bit 12)	NPS (bit 9)	PSM (bit 8)	PPCM[1] (bit 17)	When the OVRCUR pin is asserted (0b)
0	—	—	—	—	Fixed to 1b
—	1	—	—	—	Fixed to 1b
—	—	1	—	—	Fixed to 1b
1	0	0	0	—	0b
			1	0	
				1	

32.8.4 Timing Chart for Overcurrent Detection and Recovery

The figure below shows the assertion/de-assertion timings of the OVRCUR and VBUSEN pins signals at overcurrent detection and recovery.
Note that this timing chart is on the assumption that the changes of register settings to fix the Port Power bit to asserted state (see section 32.8.3, Port Power (VBUS) control specifications) have not been made.



1. When the OVRCUR pin is kept asserted (0b) for the overcurrent detection time, this module determines the occurrence of overcurrent, and then de-asserts the VBUSEN pin (0b).

2. After the overcurrent status has been resolved, and de-assertion of the OVRCUR pin (1b) is confirmed, 1b is written to the Port Power bit described in section 32.8.3, Port Power (VBUS) control specifications to turn on the Port Power (Vbus).

Note: Before the Port Power bit is set by firmware, be sure to check that the OVRCUR pin has been deasserted.

32.9 Procedure for Setting this Module

32.9.1 Host/Peripheral Common Setting Sequence

The following shows the necessary sequence common to both host and peripheral modes.

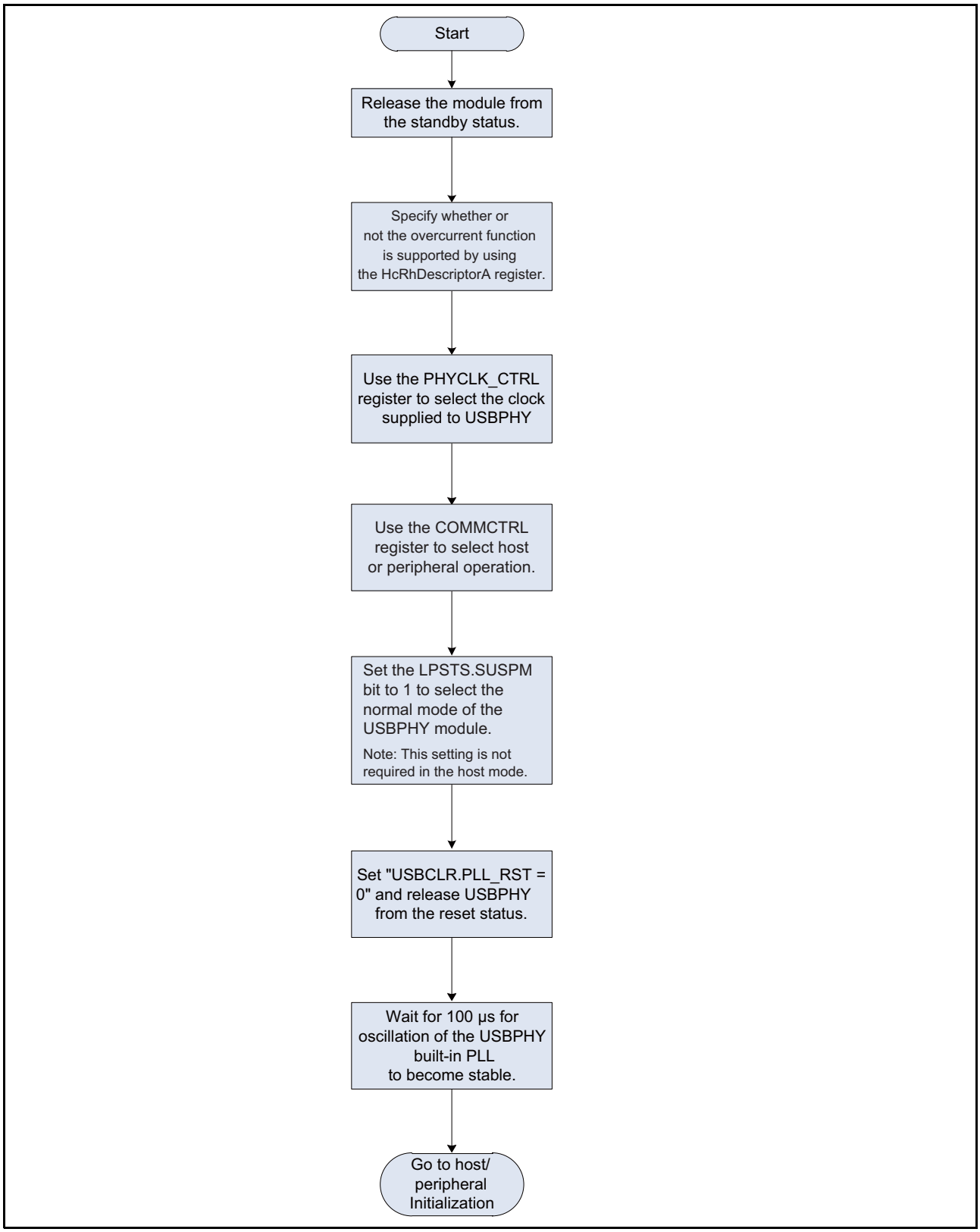


Figure 32.3 Sequence Common to Both Host and Peripheral Modes

32.9.2 Initialization Sequence

The following shows the initialization sequence in the Host mode

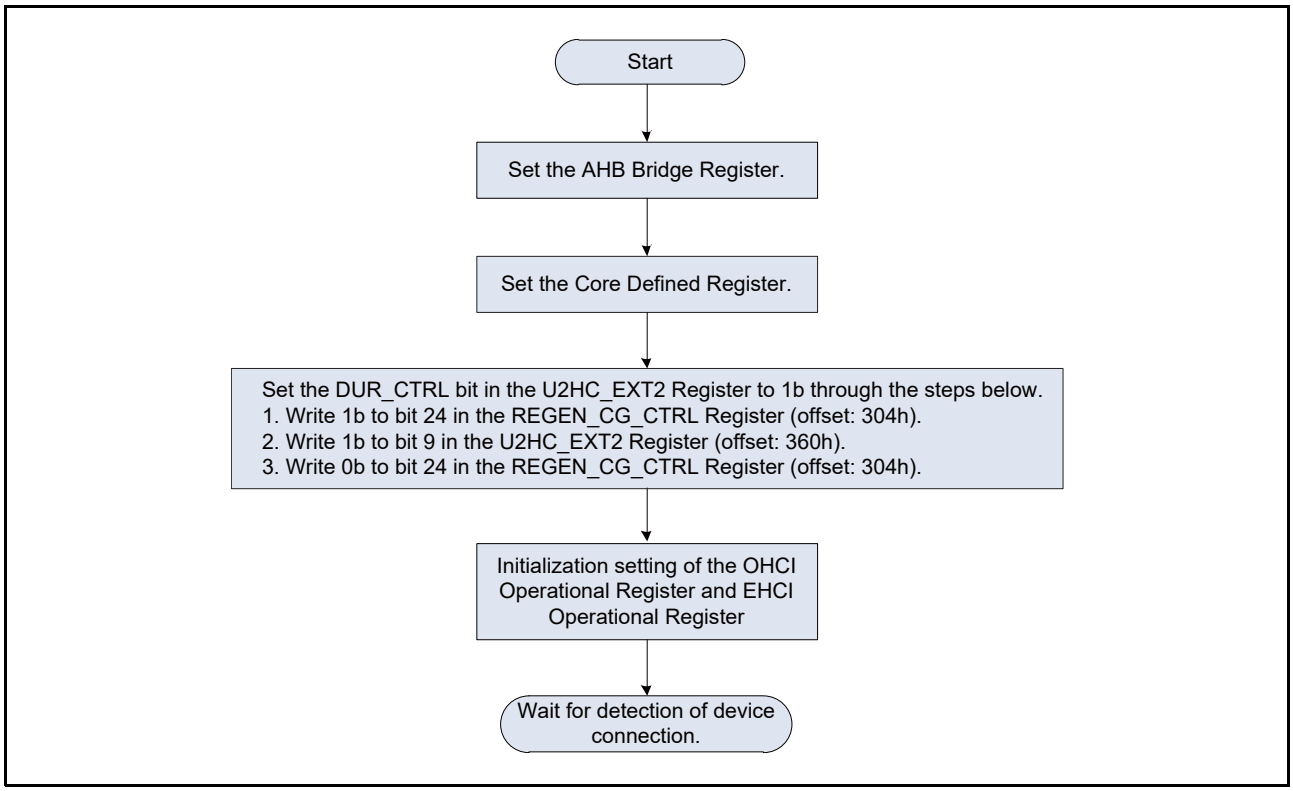


Figure 32.4 Initialization Sequence

32.9.3 Flow of Error Handling

While operating this module, if the operation falls into an abnormal state and recovery to the normal flow seems difficult, perform the following steps for reset.

[When an abnormality occurs while EHCI is running]

1. Write 1b to the HCRESET bit (bit 1) of the EHCI Operational Register USBCMD Register (offset: 120h) to execute EHCI software reset.
2. Re-initialize the EHCI Operational Register.

[When an abnormality occurs while OHCI is running]

1. Write 1b to the HCR bit (bit 0) of the OHCI Operational Register HcCommandStatus Register (offset: 008h) to execute OHCI software reset.
2. Re-initialize the OHCI Operational Register.

32.10 CCn_Rd and CCn_Ra Pins

32.10.1 Notes on Using the CCn_Ra and CCn_Rd Pins

When the CCn_Ra and CCn_Rd (n = 1 or 2) pins are to be used, include a pull-up resistor Rp and pull-down resistor Rd in conformance with the USB specification, and an external circuit for use in checking the state of the resistance (voltage state) on the CC pins of the Type-C connector. After that, input the result of checking through the CCn_Ra and CCn_Rd pins.
The threshold values for use in checking are given in Table 32.60.

Table 32.60 Threshold Values for Use in Checking for Inputs on CCn_Ra and CCn_Rd

Condition for Supply of USB Type-C VBUS	Threshold Value for CCn_Rd*	Threshold Value for CCn_Ra*
Default USB power (500 mA at 5 V)	1.6 V	0.2 V
1.5 A at 5 V	1.6 V	0.4 V
3.0 A at 5 V	2.6 V	0.8 V

Note: * When the voltage is no less than the threshold, apply a high-level signal to the CCn_Ra and CCn_Rd pins. When the voltage is below the threshold, apply a low-level signal to the CCn_Ra and CCn_Rd pins.

32.11 Points for Caution

32.11.1 Actions after Device Disconnection

When a connected device is disconnected or an instruction to suspend the OHCI is executed after device disconnection, confirm the settings of the NPS (bit 9) and PSM (bit 8) bits in the HcRhDescriptorA register and the setting of the PPCM[1] (bit 17) bit in the HcRhDescriptorB register, and follow the flow that corresponds to the register setting in the list below.

Table 32.61 Register Setting and Corresponding Flows

Register Setting			Corresponding Flows	
HcRhDescriptorA		HcRhDescriptorB	Within 4 ms of device disconnection or OHCI suspension	
NPS (bit 9)	PSM (bit 8)	PPCM[1] (bit 17)	When the flow can be completed	When the flow cannot be completed
0	1	1	Flow 1	Flow 5
0	1	0	Flow 2	Flow 6
0	0	—	Flow 2	Flow 6
1	1	1	Flow 3	Flow 7
1	1	0	Flow 4	Flow 8
1	0	—	Flow 4	Flow 8

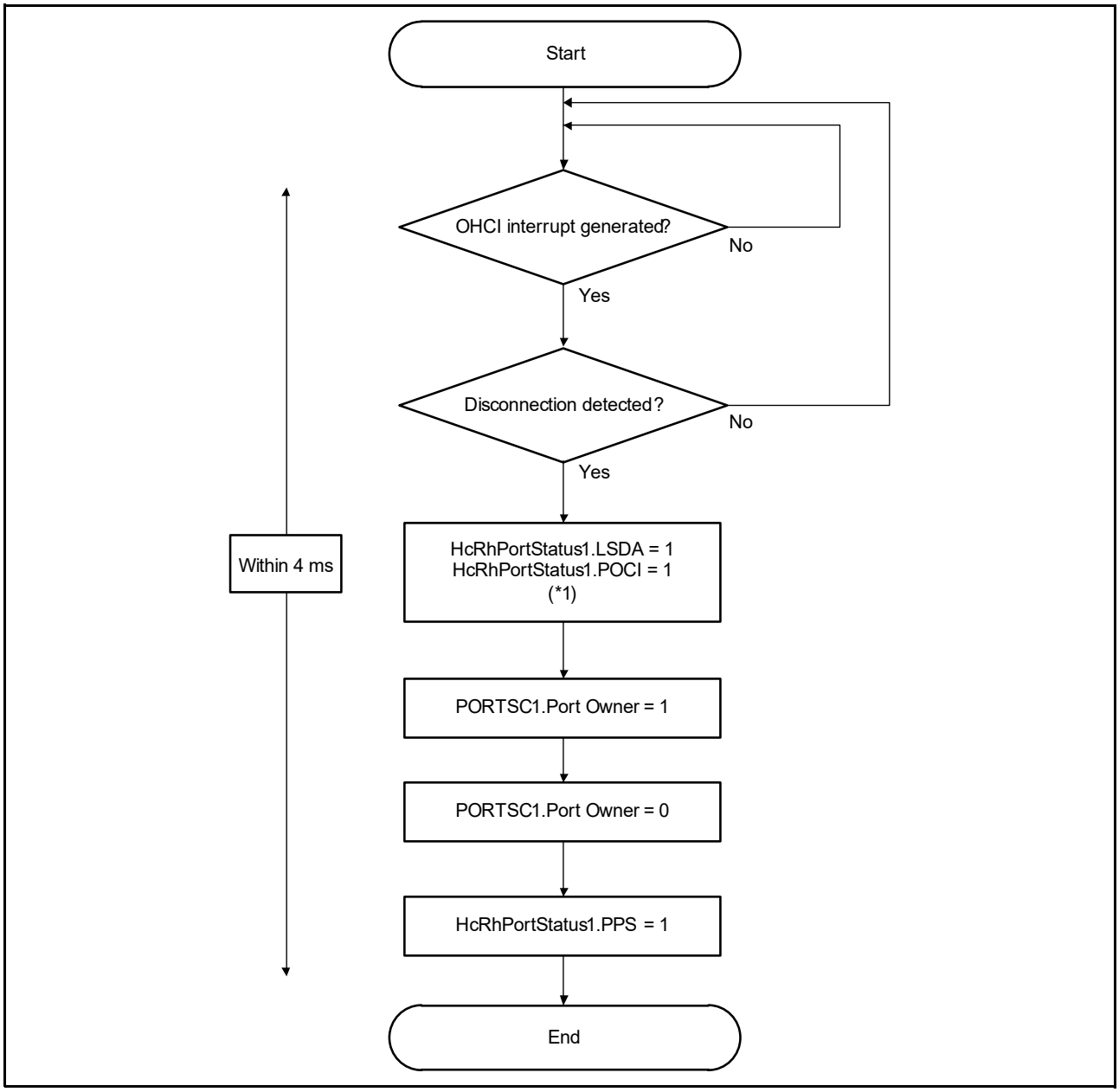


Figure 32.5 Flow 1

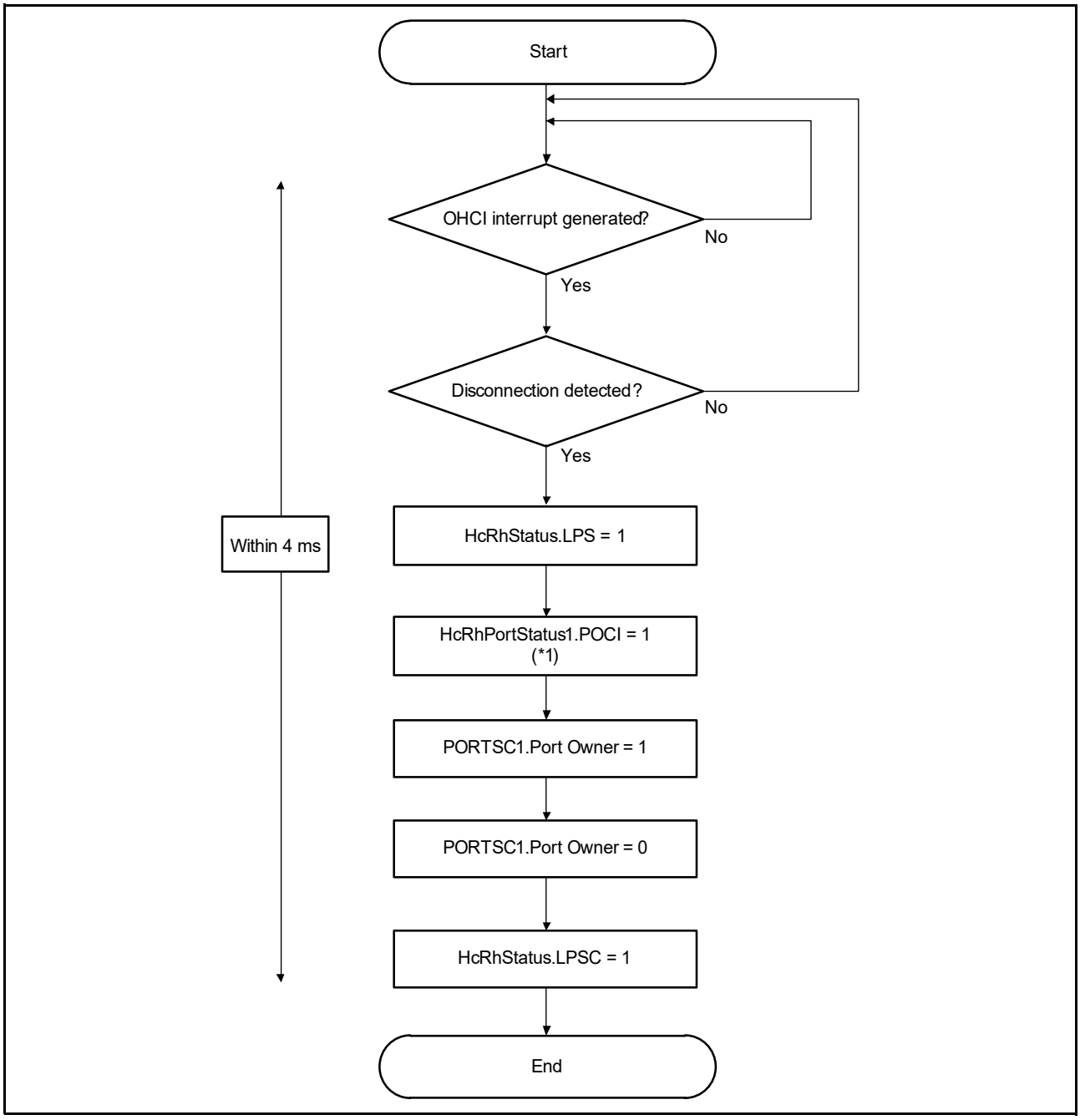


Figure 32.6 Flow 2

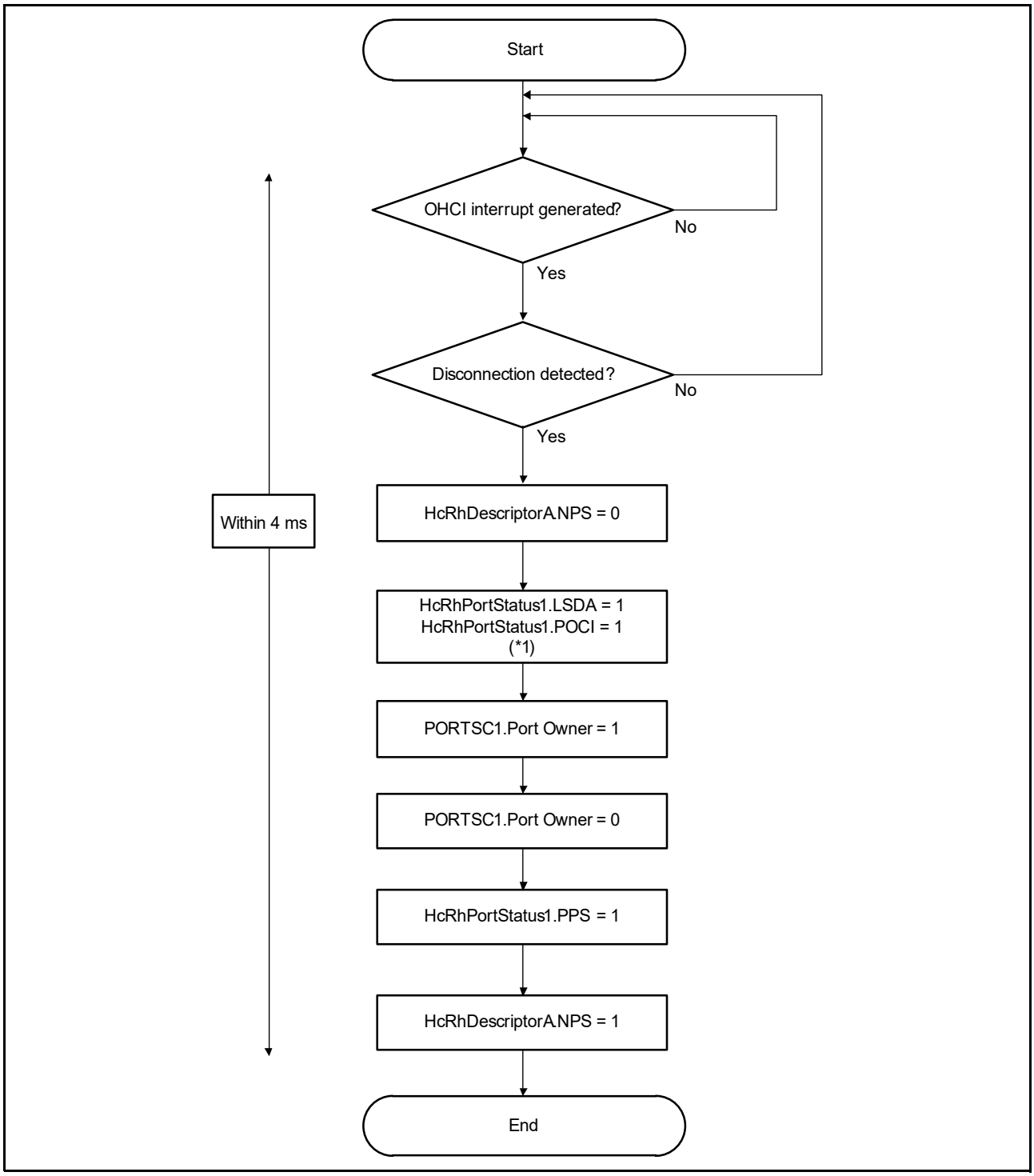


Figure 32.7 Flow 3

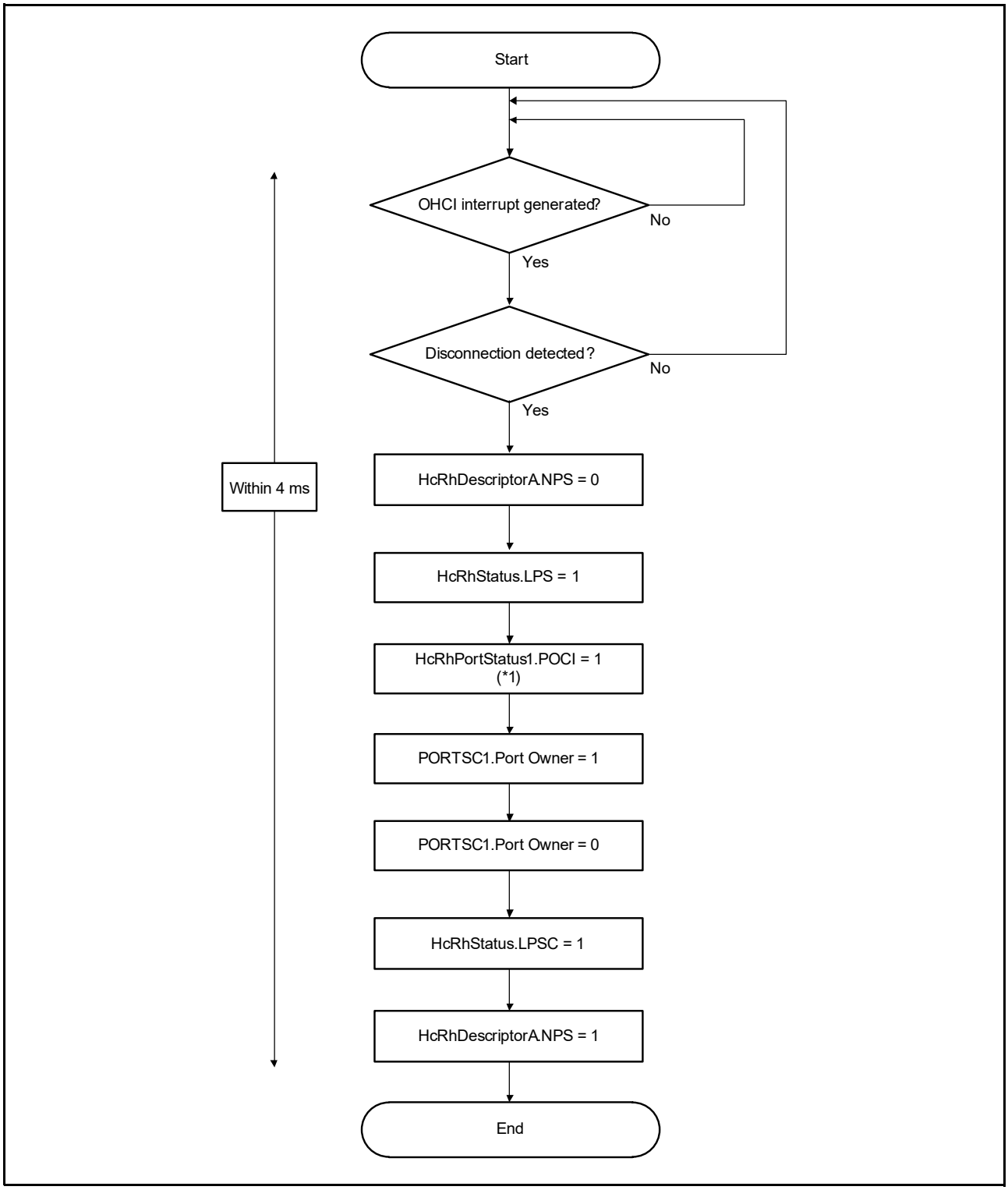


Figure 32.8 Flow 4

Point for caution on flows 1 to 4

Note 1. Making this setting leads to setting of the HcInterruptStatus.RHSC bit to 1, which causes the generation of an OHCI interrupt.
Since this interrupt is unnecessary, only clear the source bits (RHSC and CSC). Do not perform any other processing.
At this time, the settings of the bits in the HcRhPortStatus1 register which indicate the state of the port are as follows.
CCS = 0, PES = 0, CSC = 1, and PESC = 0

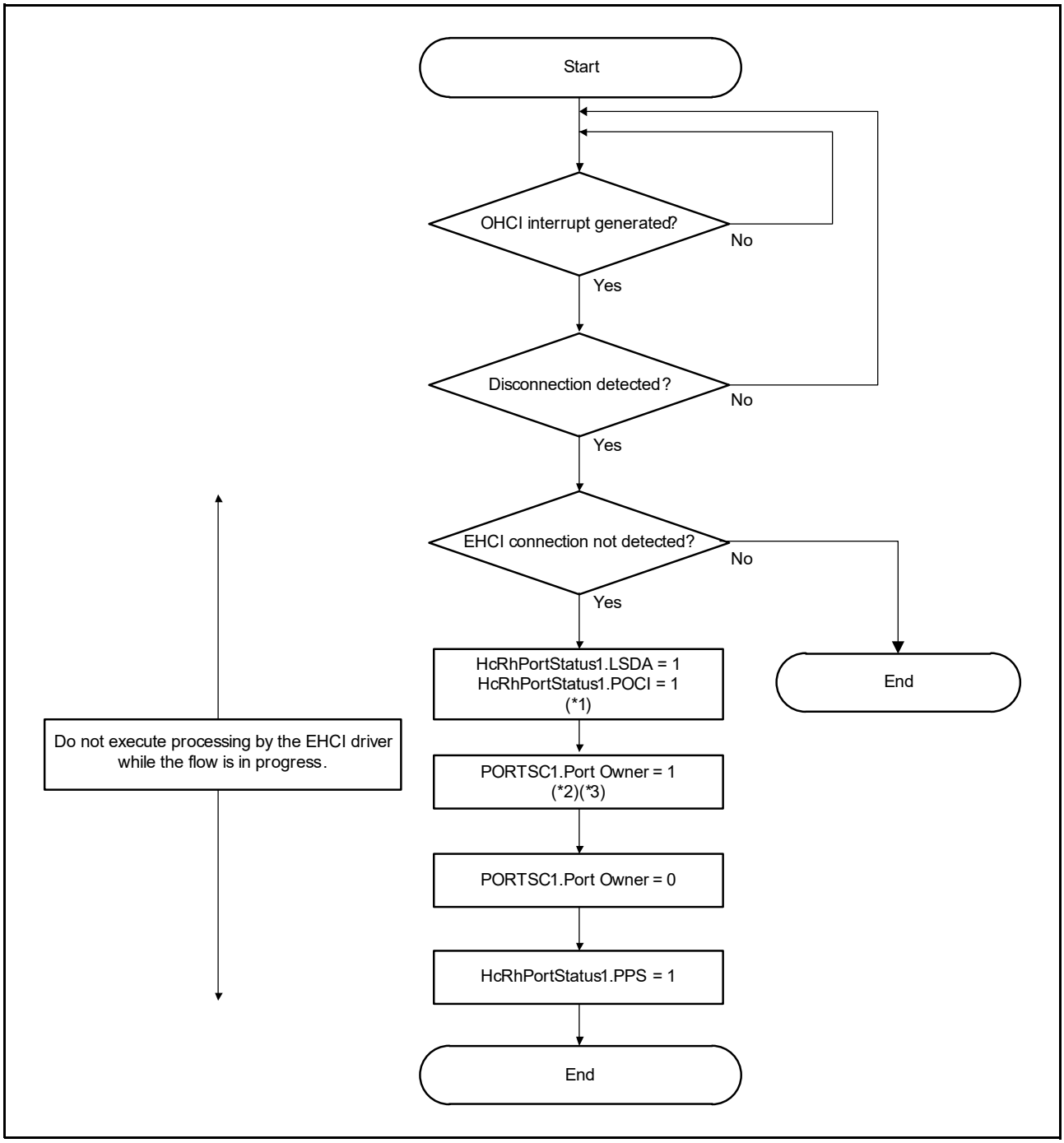


Figure 32.9 Flow 5

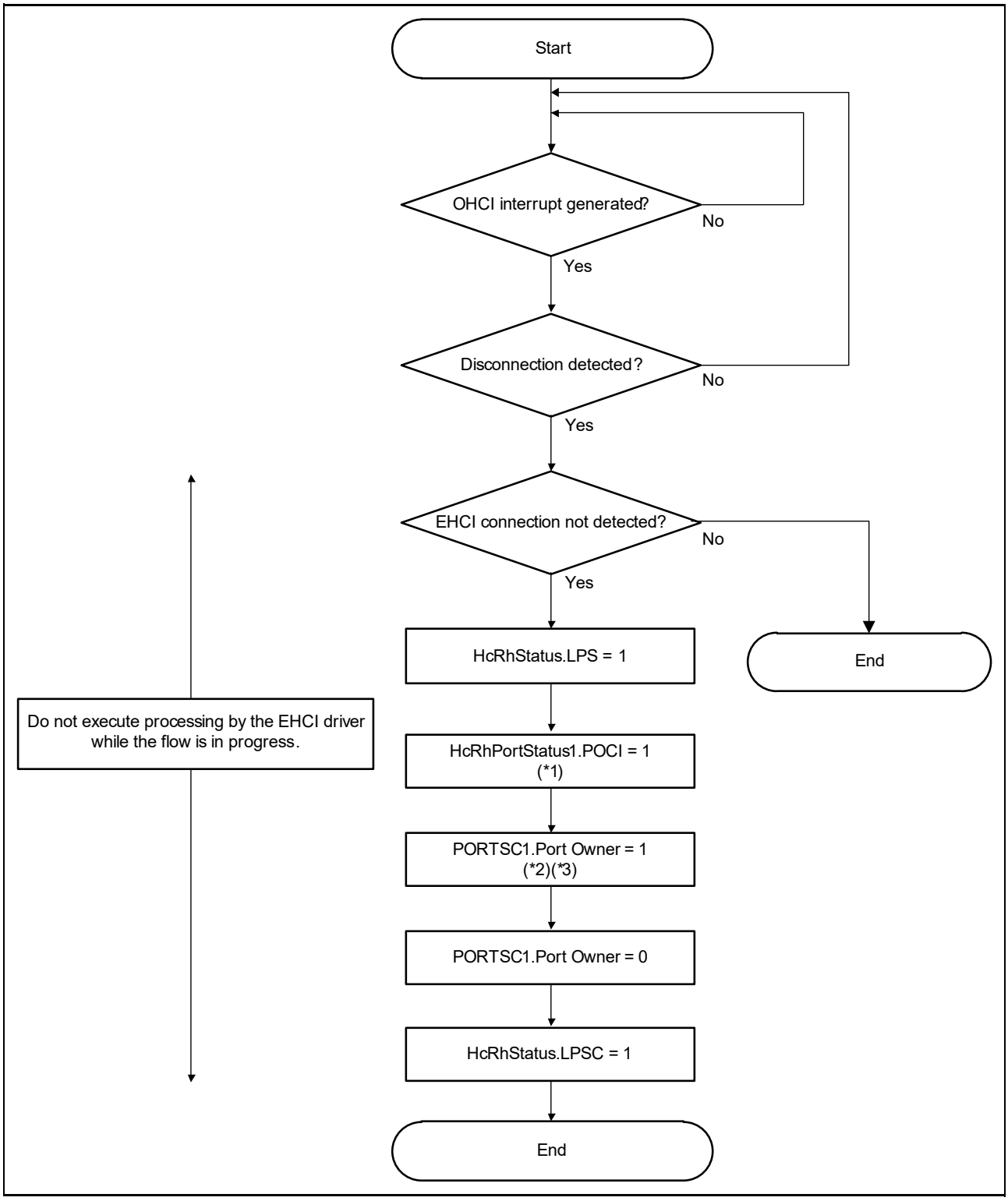


Figure 32.10 Flow 6

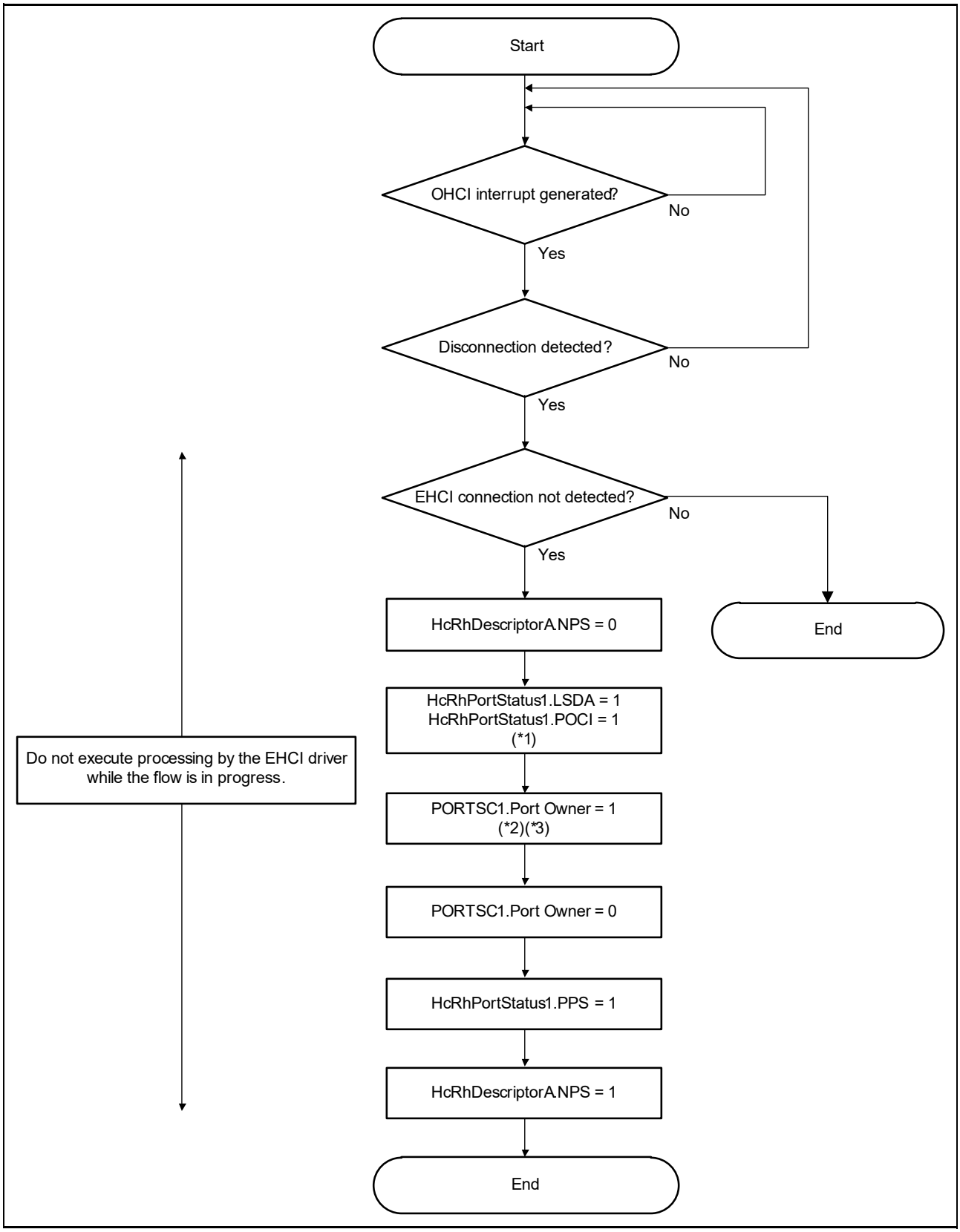


Figure 32.11 Flow 7

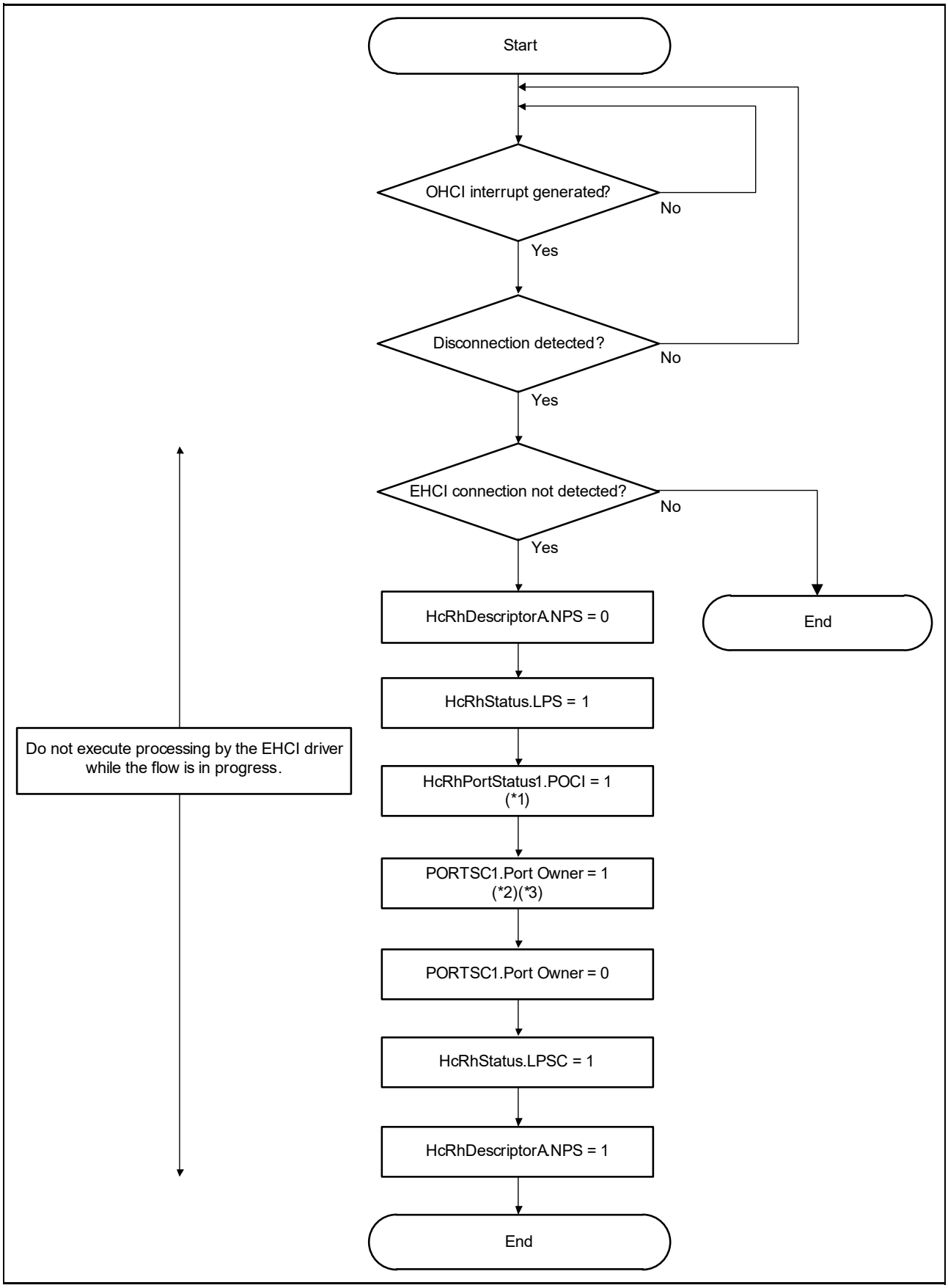


Figure 32.12 Flow 8

Points for caution on flows 5 to 8

Note 1. Making this setting leads to setting of the HcInterruptStatus.RHSC bit to 1, which causes the generation of an OHCI interrupt.
Since this interrupt is unnecessary, only clear the source bits (RHSC and CSC). Do not perform any other processing.
At this time, the settings of the bits in the HcRhPortStatus1 register which indicate the state of the port are as follows.
CCS = 0, PES = 0, CSC = 1, and PESC = 0

Note 2. No time constraint applies to these flows and a peripheral device can be reconnected while the flow is in progress.
However, contention between the detection of reconnection and this setting may lead to the HcInterruptStatus.RHSC bit being set to 1, which causes the generation of an OHCI interrupt. Since this interrupt is unnecessary, only clear the source bits (RHSC and CSC). Do not perform any other processing.
At this time, the settings of the bits in the HcRhPortStatus1 register which indicate the state of the port are as follows.
CCS = 0, PES = 0, CSC = 1, and PESC = 0

Note 3. After reconnection of the peripheral device is detected, changing the port owner to the OHCI leads to setting of the USBSTS.PortChangeDetect bit to 1, which causes the generation of an EHCI interrupt. Note that processing by the EHCI driver must not proceed until this flow is completed.

33. USB 2.0 Function Module

33.1 Overview

33.1.1 Overview

This LSI has two channels of USB 2.0 host/function module. Switching between the host and peripheral functions for each channel is possible by setting the UCOM register.
The setting for battery charging is handled by the host controller even if the peripheral controller is selected.

This chapter describes the peripheral controller. For details on the host/peripheral common circuit and battery charging, see section 32, USB 2.0 Host Module.

This module is a universal serial bus (USB) controller that has peripheral functions.
This module supports high-speed and full-speed transfer defined by the Universal Serial Bus Specification Revision 2.0.

This module supports all transfer types defined in the USB Specification. This module incorporates 8 Kbytes of buffer memory for data transfer, and can use a maximum of 16 pipes. You can assign any endpoint number to any pipe other than pipe 0 in conformity to the peripheral equipment or system to communicate with.

33.1.2 Features

33.1.2.1 Peripheral controller supporting high-speed USB

- On-chip peripheral USB controller

33.1.2.2 Support of all types of USB transfer

- Supporting all types of USB transfer, including isochronous transfer
- Control transfer
- Bulk transfer
- Interrupt transfer (high-bandwidth transfers not supported)
- Isochronous transfer (high-bandwidth transfers not supported)

33.1.2.3 Bus interface

- Includes a two-channel DMA interface

33.1.2.4 Pipe configuration

- 8 Kbytes of buffer memory for USB communications for each channel
- Up to 16 pipes (including the default control pipe) selectable for each channel
- Programmable pipe configuration
- Any endpoint number assignable to pipes other than pipe 0

Table 33.1 Pipe Settings

PIPE	Transfer Type	Double buffer	Continuous transfer mode	Buffer size
PIPE0	Control	—	—	Fixed to 64 bytes/ 256 bytes (CNTMD = 1)
PIPE1	Iso/Bulk	√	√ (Bulk only)	Up to 2 Kbytes
PIPE2	Iso/Bulk	√	√ (Bulk only)	Up to 2 Kbytes
PIPE3	Bulk	√	√	Up to 2 Kbytes
PIPE4	Bulk	√	√	Up to 2 Kbytes
PIPE5	Bulk	√	√	Up to 2 Kbytes
PIPE6	Int	—	—	Fixed to 64 bytes
PIPE7	Int	—	—	Fixed to 64 bytes
PIPE8	Int	—	—	Fixed to 64 bytes
PIPE9	Int/Bulk	√ (Bulk only)	√ (Bulk only)	Up to 2 Kbytes
PIPE10	Int/Bulk	√ (Bulk only)	√ (Bulk only)	Up to 2 Kbytes
PIPE11	Bulk	√	√	Up to 2 Kbytes
PIPE12	Bulk	√	√	Up to 2 Kbytes
PIPE13	Bulk	√	√	Up to 2 Kbytes
PIPE14	Bulk	√	√	Up to 2 Kbytes
PIPE15	Bulk	√	√	Up to 2 Kbytes

33.1.2.5 Features of peripheral functions

- Support of high-speed transfer (at 480 Mbps) and full-speed transfer (at 12 Mbps)
- Automatic recognition of high-speed or full-speed operation based on automatic response to the reset handshake
- Control transfer stage monitoring function
- Device state monitoring function
- Automatic response to SET_ADDRESS request
- NAK response interrupt (NRDY)
- SOF interpolation

33.1.2.6 Features of DMA transfer

DMA transaction mode:	Fetching in both register and link modes are supported.
Interrupt:	Level is supported.
Transfer size:	A transfer size from 1 to 128 bytes can be selected separately for the transfer source and transfer destination.
Skip (scatter/gather) function:	The access size and skip size can be specified separately for the transfer source and destination.
Suspend function:	A running DMA transaction can be suspended temporarily.
Interval function:	The interval of DMA transfers can be specified to control the bus occupancy.

33.1.2.7 Other functions

- Byte endian swap function to support both big endian and little endian as data formats (when using only CFIFO)
- Transfer ending function using a transaction counter
- SOF pulse output function
- BRDY interrupt event notification timing change function (BFRE)
- Function (SHTNAK) to set NAK in the response PID when transfer ends
- Support of the Link Power Management (LPM) ECN, making available a new low-power-consumption state (L1 state)

33.1.3 Overview of Functions

33.1.3.1 Automatic recognition of USB transfer speed

This module automatically recognizes USB transfer speed.

(1) Methods of FIFO buffer memory access

This module supports the two types of access described below to the FIFO buffer memory for USB data transfer.

(a) Access from the CPU

Specify a FIFO port address, and then write data to or read data from the FIFO buffer memory.

(b) Direct memory access (DMA)

Selecting a pipe window and setting the DMA control registers enables writing data to or reading data from the FIFO buffer memory.

33.1.3.2 USB event

This module notifies the event in USB operation by issuing an interrupt.
You can specify whether to enable notification by interrupt for individual interrupt types and sources through software settings.

33.1.3.3 USB data transfer

This module performs all types of USB data transfer: control transfer, bulk transfer, interrupt transfer, and isochronous transfer. The following pipe resources are available for individual transfer types:
(1) One pipe dedicated to control transfer
(2) Three pipes dedicated to interrupt transfer
(3) Eight pipes dedicated to bulk transfer
(4) Two pipes selectively used for bulk or isochronous transfer
(5) Two pipes selectively used for bulk or interrupt transfer

For each pipe, specify the settings, including transfer type, endpoint number, and maximum packet size, required for USB transfer according to the system.

This module can incorporate up to 8 Kbytes of buffer memory. For the pipes dedicated to bulk transfer and those selectively used for bulk transfer or isochronous transfer, allocate buffer memory and specify a buffer operating mode and other necessary settings according to the system. Setting the buffer operating mode enables high-speed data transfers with fewer interrupts to be performed by using double-buffering and continuous transfer of data packets.

33.1.3.4 SOF pulse output function

This module has a function to output an SOF pulse to indicate the timing of SOF packet transmission. This module asserts a SOF pulse output signal when an SOF packet is received. This module outputs pulses at regular intervals based on an SOF interpolation timer even when an SOF packet is damaged.

33.1.4 Restriction matter and Notes

33.1.4.1 Restriction matter

(1) Restrictions on the USB specifications

It's no support About below of USB 2.0 specification

- HighBandWidth transfer is no support

(2) Restrictions on DMA Master

- DAD = 1 (destination address fixed) and can not use skip transfer of destination side. When forwarding by such setting, movement is unsettled. Should not do such transmission.
- SAD = 1 (source address fixed), and can not use skip transfer of source side. When forwarding by such setting, movement is unsettled. Should not do such transmission.
- DAD = 1 (destination address fixed), and can not use the beat unalign transfer of destination side. When forwarding by such setting, movement is unsettled. Should not do such transmission.
- SAD = 1 (source address fixed), and can not use beat unalign transfer of source side. When forwarding by such setting, movement is unsettled. Should not do such transmission.
- When REQD =1, SBE = 1 (sweep mode) and compulsion discharge function can not use.

33.1.4.2 Notes

(1) DMA transfer and passing problem of interrupt signal in DMA Master construction.

(a) Overview

A USBFDMAmn interrupt (m, n = 0, 1) might occur before the last data of DMA transaction is written to the write-target device.

(b) Plan to avoid

A plan to avoid the above problem is indicated below.

Plan to avoid 1) HPROT is set as non bufferable.
According to the DMA mode, set 0 in the DPR[2] bit in the CHEXT_n register or the LDPR[2] bit in the DCTRL register, set the HPROT signal as non bufferable, and then perform the DMA transaction.

If all transfers are set to non bufferable, transfer efficiency might fall.

In such cases, set the most part of transaction as bufferable and perform a transfer, and then perform the last transfer by setting a register set or descriptor as non bufferable.

(2) Setting for battery charging when the peripheral controller is selected

The setting for battery charging is handled by the host controller even if the peripheral controller is selected.

33.2 Registers

How to read the register table

(1) Bit number:

(2) State after reset: Initial state of the register that occurs immediately after a reset
"Power on Reset" indicates the initial state at power-on reset.
The state after USB reset is the initial state of the register that occurs when this module detects a USB bus reset.
Significant points regarding reset operation are indicated in notes.
"-" indicates that a user's setting is retained without this module operation having been performed.
“X” indicates that the value is undefined.

(4) Access condition: The condition to be met when this module accesses the register for an operation.
R: Reading only
W: Writing only
R/W: Reading or writing
R(0): 0-reading only
W(1): 1-writing only

(5) Name: Bit symbol and bit name

(6) Function: Description of functions.

<Example of description>

(1) Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Symbol		A bit	B bit	C bit												
(2) Power on Reset	X	0	0	0												
USB Bus Reset	X	0	—	—												

bit	Name	Function	R/W
15	—	Nothing is assigned. Fix this bit to 0.	
14	A bit AAA enable	0: Disables operation 1: Enables operation	R/W
13	B bit BBB operation	0: Outputs low-level signal 1: Outputs high-level signal	R
12	C bit CCC control	0: +++++++ 1: +++++++	R(0)/W(1)
	(5)	(6)	(4)

33.2.1 Base Address

Table 33.2 Base addresses for each channel of the USB function module

Channel number	Base address
0	E821 9000h
1	E821 B000h

33.2.2 List of Registers

Table 33.3 lists the registers of this module.

Table 33.3 List of Registers

Address	Symbol	Name	Access unit
000	SYSCFG0	System Configuration Control Register 0	16-bit
002	SYSCFG1	System Configuration Control Register 1	16-bit
004	SYSSTS0	System Configuration Status Register	16-bit
008	DVSTCTR0	Device Control Register 0	16-bit
00C	TESTMODE	Test Mode Register	16-bit
014	CFIFO	CFIFO Port Register	8-/16-/32-bit
016			
020	CFIFOSEL	CFIFO Port Select Register	16-bit
022	CFIFOCTR	CFIFO Port Control Register	16-bit
028	D0FIFOSEL	D0FIFO Port Select Register	16-bit
02A	D0FIFOCTR	D0FIFO Port Control Register	16-bit
02C	D1FIFOSEL	D1FIFO Port Select Register	16-bit
02E	D1FIFOCTR	D1FIFO Port Control Register	16-bit
030	INTENB0	Interrupt Enable Register 0	16-bit
036	BRDYENB	BRDY Interrupt Enable Register	16-bit
038	NRDYENB	NRDY Interrupt Enable Register	16-bit
03A	BEMPENB	BEMP Interrupt Enable Register	16-bit
03C	SOFCFG	SOF Output Configuration Register	16-bit
040	INTSTS0	Interrupt Status Register 0	16-bit
046	BRDYSTS	BRDY Interrupt Status Register	16-bit
048	NRDYSTS	NRDY Interrupt Status Register	16-bit
04A	BEMPSTS	BEMP Interrupt Status Register	16-bit
04C	FRMNUM	Frame Number Register	16-bit
04E	UFRMNUM	Micro Frame Number Register	16-bit
050	USBADDR	USB Address Register	16-bit
054	USBREQ	USB Request Type Register	16-bit
056	USBVAL	USB Request Value Register	16-bit
058	USBINDX	USB Request Index Register	16-bit
05A	USBLENG	USB Request Length Register	16-bit
05C	DCPCFG	DCP Configuration Register	16-bit
05E	DCPMAXP	DCP Max. Packet Size Register	16-bit
060	DCPCTR	DCP Control Register	16-bit
064	PIPESEL	Pipe Window Select Register	16-bit
068	PIPECFG	Pipe Configuration Register	16-bit
06A	PIPEBUF	Pipe Buffer Setting Register	16-bit

Address	Symbol	Name	Access unit
06C	PIPEMAXP	Pipe Max. Packet Size Register	16-bit
06E	PIPEPERI	Pipe Cycle Control Register	16-bit
070	PIPE1CTR	PIPE1 Control Register	16-bit
072	PIPE2CTR	PIPE2 Control Register	16-bit
074	PIPE3CTR	PIPE3 Control Register	16-bit
076	PIPE4CTR	PIPE4 Control Register	16-bit
078	PIPE5CTR	PIPE5 Control Register	16-bit
07A	PIPE6CTR	PIPE6 Control Register	16-bit
07C	PIPE7CTR	PIPE7 Control Register	16-bit
07E	PIPE8CTR	PIPE8 Control Register	16-bit
080	PIPE9CTR	PIPE9 Control Register	16-bit
082	PIPEACTR	PIPEA Control Register	16-bit
084	PIPEBCTR	PIPEB Control Register	16-bit
086	PIPECCTR	PIPEC Control Register	16-bit
088	PIPEDCTR	PIPED Control Register	16-bit
08A	PIPEECTR	PIPEE Control Register	16-bit
08C	PIPEFCTR	PIPEF Control Register	16-bit
090	PIPE1TRE	PIPE1 Transaction Counter Enable Register	16-bit
092	PIPE1TRN	PIPE1 Transaction Counter Register	16-bit
094	PIPE2TRE	PIPE2 Transaction Counter Enable Register	16-bit
096	PIPE2TRN	PIPE2 Transaction Counter Register	16-bit
098	PIPE3TRE	PIPE3 Transaction Counter Enable Register	16-bit
09A	PIPE3TRN	PIPE3 Transaction Counter Register	16-bit
09C	PIPE4TRE	PIPE4 Transaction Counter Enable Register	16-bit
09E	PIPE4TRN	PIPE4 Transaction Counter Register	16-bit
0A0	PIPE5TRE	PIPE5 Transaction Counter Enable Register	16-bit
0A2	PIPE5TRN	PIPE5 Transaction Counter Register	16-bit
0A4	PIPEBTRE	PIPEB Transaction Counter Enable Register	16-bit
0A6	PIPEBTRN	PIPEB Transaction Counter Register	16-bit
0A8	PIPECTRE	PIPEC Transaction Counter Enable Register	16-bit
0AA	PIPECTRN	PIPEC Transaction Counter Register	16-bit
0AC	PIPEDTRE	PIPED Transaction Counter Enable Register	16-bit
0AE	PIPEDTRN	PIPED Transaction Counter Register	16-bit
0B0	PIPEETRE	PIPEE Transaction Counter Enable Register	16-bit
0B2	PIPEETRN	PIPEE Transaction Counter Register	16-bit
0B4	PIPEFTRE	PIPEF Transaction Counter Enable Register	16-bit
0B6	PIPEFTRN	PIPEF Transaction Counter Register	16-bit
0B8	PIPE9TRE	PIPE9 Transaction Counter Enable Register	16-bit
0BA	PIPE9TRN	PIPE9 Transaction Counter Register	16-bit
0BC	PIPEATRE	PIPEA Transaction Counter Enable Register	16-bit
0BE	PIPEATRN	PIPEA Transaction Counter Register	16-bit
0D0	DEVADD0	Device Address 0 Configuration Register	16-bit
0D2	DEVADD1	Device Address 1 Configuration Register	16-bit
0D4	DEVADD2	Device Address 2 Configuration Register	16-bit
0D6	DEVADD3	Device Address 3 Configuration Register	16-bit
0D8	DEVADD4	Device Address 4 Configuration Register	16-bit

Address	Symbol	Name	Access unit
0DA	DEVADD5	Device Address 5 Configuration Register	16-bit
0DC	DEVADD6	Device Address 6 Configuration Register	16-bit
0DE	DEVADD7	Device Address 7 Configuration Register	16-bit
0E0	DEVADD8	Device Address 8 Configuration Register	16-bit
0E2	DEVADD9	Device Address 9 Configuration Register	16-bit
0E4	DEVADDA	Device Address A Configuration Register	16-bit
100	LPCTRL	Low Power Control Register	16-bit
102	LPSTS	Low Power Status Register	16-bit
104	PHYFUNCTR	PHY Function Control Register	16-bit
10A	PHYOTGCTR	PHY OTG Control Register	16-bit
144	PL1CTRL1	Peripheral L1 Control Register 1	16-bit
146	PL1CTRL2	Peripheral L1 Control Register 2	16-bit
400	N0SA_0	Next0 Source Address Register 0	32-bit
404	N0DA_0	Next0 Destination Address Register 0	32-bit
408	N0TB_0	Next0 Transaction Byte Register 0	32-bit
40C	N1SA_0	Next1 Source Address Register 0	32-bit
410	N1DA_0	Next1 Destination Address Register 0	32-bit
414	N1TB_0	Next1 Transaction Byte Register 0	32-bit
418	CRSA_0	Current Source Address Register 0	32-bit
41C	CRDA_0	Current Destination Address Register 0	32-bit
420	CRTB_0	Current Transaction Byte Register 0	32-bit
424	CHSTAT_0	Channel Status Register 0	32-bit
428	CHCTRL_0	Channel Control Register 0	32-bit
42C	CHCFG_0	Channel Configuration Register 0	32-bit
430	CHITVL_0	Channel Interval Register 0	32-bit
434	CHEXT_0	Channel Extension Register 0	32-bit
438	NXLA_0	Next Link Address Register 0	32-bit
43C	CRLA_0	Current Link Address Register 0	32-bit
440	N0SA_1	Next0 Source Address Register 1	32-bit
444	N0DA_1	Next0 Destination Address Register 1	32-bit
448	N0TB_1	Next0 Transaction Byte Register 1	32-bit
44C	N1SA_1	Next1 Source Address Register 1	32-bit
450	N1DA_1	Next1 Destination Address Register 1	32-bit
454	N1TB_1	Next1 Transaction Byte Register 1	32-bit
458	CRSA_1	Current Source Address Register	32-bit
45C	CRDA_1	Current Destination Address Register 1	32-bit
460	CRTB_1	Current Transaction Byte Register 1	32-bit
464	CHSTAT_1	Channel Status Register 1	32-bit
468	CHCTRL_1	Channel Control Register 1	32-bit
46C	CHCFG_1	Channel Configuration Register 1	32-bit
470	CHITVL_1	Channel Interval Register 1	32-bit
474	CHEXT_1	Channel Extension Register 1	32-bit
478	NXLA_1	Next Link Address Register 1	32-bit
47C	CRLA_1	Current Link Address Register 1	32-bit
600	SCNT_0	Source Continuous Register 0	32-bit
604	SSKP_0	Source Skip Register 0	32-bit

Address	Symbol	Name	Access unit
608	DCNT_0	Destination Continuous Register 0	32-bit
60C	DSKP_0	Destination Skip Register 0	32-bit
620	SCNT_1	Source Continuous Register 1	32-bit
624	SSKP_1	Source Skip Register 1	32-bit
628	DCNT_1	Destination Continuous Register 1	32-bit
62C	DSKP_1	Destination Skip Register 1	32-bit
700	DCTRL	DMA Control Register	32-bit
704	DSCITVL	Descriptor Interval	32-bit
710	DST_EN	DMA Status EN Register	32-bit
714	DST_ER	DMA Status ER Register	32-bit
718	DST_END	DMA Status END Register	32-bit
71C	DST_TC	DMA Status TC Register	32-bit
720	DST_SUS	DMA Status SUS Register	32-bit

33.2.3 System Configuration Control Registers

33.2.3.1 System Configuration Control Register 0 [SYSCFG0] <Address: 000H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	—	—	CNEN	HSE	—	DRPD	DPRPU	—	—	—	USBE
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 9	—	Nothing is assigned to these bits. Fix these bits to 0.	
8	CNEN	This bit prohibits or enables single-ended receiver operation. 0: Single-ended receiver operation prohibited 1: Single-ended receiver operation enabled	R/W
7	HSE	This bit prohibits or enables High-Speed operation. 0: High-Speed operation prohibited (Full-Speed) 1: High-Speed operation enable(The controller detects the communication speed.)	R/W
6	—	Nothing is assigned to this bit. Fix this bit to 0.	
5	DRPD	D+/D- line resistor control Set this bit to 0 to use this module. For details, see Control of USB Data Bus Resistors.	R/W
4	DPRPU	D+ line resistor control This bit prohibits or enables D+ line pull-up for the peripheral controller function. For details, see Control of USB Data Bus Resistors. 0: Pull Up prohibited 1: Pull Up enabled	R/W
3 to 1	—	Nothing is assigned to these bits. Fix these bits to 0.	
0	USBE	USB block operation prohibited This bit prohibits or enables USB block operation. 0: USB block operation prohibited 1: USB block operation enabled	R/W

Note: Data can be written to and read from this register even while the UTMI+PHY clock is stopped.
However, if a value is set while the UTMI+PHY clock is stopped, the corresponding function takes effect after the oscillation of UTMI+PHY clock starts.

(1) Single-ended receiver operation enable (CNEN) bit

Setting this bit enables the single-ended receiver to operate. This bit is intended to prevent damage by inrush current that can be caused when the single-ended receiver in unattached status goes floating. This bit also allows the LNST bit to be referenced.
Set this bit when VBUS is detected as the result of a VBUS interrupt. Clear this bit when VBUS is removed.

(2) High-speed operation enable (HSE) bit

Setting this bit enables the high-speed operation. When this bit is 1, this module performs a high-speed or full-speed operation according to the result of reset handshake.

When the HSE bit is 0, this module performs a full-speed operation.
When the HSE bit is 1, this module executes the reset handshake protocol, and then automatically performs a high-speed or full-speed operation according to the result of reset handshake.

Rewriting the value of this bit must be done when the DPRPU bit is 0.

(3) D+/D- line resistor control (DRPD or DPRPU) bit

Table 33.4 shows available settings of the resistors for the USB data bus. Use the DPRPU bit to select the USB data bus resistors.

Table 33.4 Control of USB Data Bus Resistors

Setting		Control of USB Data Bus Resistors		
DRPD	DPRPU	D- Line	D+ Line	Remarks
0	0	Open	Open	
0	1	Open	Pull-Up	Specify the settings as shown in the left.
1	0	Pull-Down	Pull-Down	Initial state (When power on reset is canceled)
1	1	Pull-Down	Pull-Up	Setting prohibited

(a) D+ pull-up resistor control (DPRPU) bit

Setting this bit enables this module to notify the USB host of attaching by pulling up the D+ line voltage to 3.3 V. Clearing this bit enables this module to let the USB host know that the device has been detached by stopping pulling up the D+ line voltage.

(4) USB block operation enable (USBE) bit

This bit enables or disables the operation of the USB block of this module.
When this bit is changed from 1 to 0, this module initializes the bits shown in Table 33.5.

Table 33.5 Register Bits That Are Initialized by Writing 0 to the USBE Bit

Register Name	Bit Name
SYSSTS0	LNST
DVSTCTR0	RHST
INTSTS0	DVSQ
USBADDR	USBADDR
USBREQ	bRequest bmRequestType
USBVAL	wValue
USBINDX	wIndex
USBLENG	wLength

Note: Changing the value of this bit must be done when the SUSPENDM bit is 1 and after the oscillation of UTMI+PHY clock starts.

33.2.3.2 System Configuration Control Register 1 [SYSCFG1] <Address: 002H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	—	—	—	—	—	BWAIT					
0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 12	—	Nothing is assigned to these bits. Fix these bits to 0.	
11 to 8	—	Nothing is assigned to these bits. Fix these bits to 0.	
7, 6	—	Nothing is assigned to these bits. Fix these bits to 0.	
5 to 0	BWAIT	CPU bus access wait specification These bits specify the number of wait cycles for the access to this module. 000000: 0 wait cycles (2 access cycles) : 000010: 2 wait cycles (4 access cycles) : 000100: 4 wait cycles (6 access cycles) : 001111: 15 wait cycles (17 access cycles) (default) : 111111: 63 wait cycles (65 access cycles)	R/W

(1) CPU bus access wait specification (BWAIT) bits

These bits specify the wait cycles for the access to the HPB.
The following restriction is placed on the cycle of the access to the registers at address 04H or after of this module:
Restriction on wait cycle: The cycle of continuous accesses to registers of this module must be at least 67 ns.

To comply with this restriction, you must control the number of wait cycles with the internal bus clock (Bφ) frequency. The default of wait cycles is 17 clock cycles (maximum limit). Select an optimum setting.

This setting is also applied to accesses to FIFO port registers. The maximum speeds of accesses to FIFO ports are as follows:
MBW = 10 (32-bit access width): max 60 MBytes/sec
MBW = 01 (16-bit access width): max 30 MBytes/sec
MBW = 00 (8-bit access width): max 15 MBytes/sec

33.2.4 System Configuration Status

33.2.4.1 System Configuration Status Register (SYSSTS0) <Address: 004H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	LNST	
—	—	0	0	0	0	0	0	0	0	0	0	0	—	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 2	—	Nothing is assigned to these bits. Fix these bits to 0.	
1, 0	LNST	USB Line status monitor The USB line status is displayed. Note: See the detailed description.	R

(1) Line status monitor (LNST) bits

Table 33.6 shows the line status of the USB data bus of this module. This module monitors the line status (status of the D+ and D- lines) of the USB data bus in the LNST bits of the SYSSTS0 register.

Referencing the LNST bits must be done only after the USBE bit is set and attaching is performed (the DPRPU bit is set).

Table 33.6 Line Status of USB Data Bus

LNST [1]	LNST [0]	Full-Speed operation	High-Speed operation	Chirp operation
0	0	SE0	Squelch	Squelch
0	1	J State	Unsquelch	Chirp J
1	0	K State	Invalid	Chirp K
1	1	SE1	Invalid	Invalid

Note: Chirp: State in which high-speed operation is enabled (HSE = 1) and the reset handshake protocol is being executed
Squelch: SE0 or idle state
Unsquelch: High-speed J or high-speed K state
Chirp J: Chirp J State
Chirp K: Chirp K State

33.2.5 USB Signal Control Registers

33.2.5.1 Device State Control Register 0 [DVSTCTR0] <Address: 008H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRST 1	—	—	—	—	—	—	WKUP	—	—	—	—	—	RHST		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	0	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15	BRST1	This bit is used in resumption in response to a bus reset. Set this bit according to 33.9.16.3, Flowchart of resumption from deep standby in response to a bus reset. Operation cannot be guaranteed if the procedure in the flowchart is not observed.	R/W
14 to 9	—	Nothing is assigned to these bits. Fix these bits to 0.	—
8	WKUP	Remote wakeup output This bit prohibits or enables Remote wakeup (resume signal output). 0: Remote wakeup signal is not output. 1: Remote wakeup signal is output.	R/W(1)
7 to 3	—	Nothing is assigned to these bits. Fix these bits to 0.	
2 to 0	RHST	Reset handshake This bit indicates the reset handshake status. Note: See the detailed description.	R

(1) Remote wakeup (resume signal output) enable (WKUP) bit

When this bit is set, this module outputs the remote wakeup signal to the USB.
This module manages the time of remote wakeup signal output. When the WKUP bit is set, this module outputs the K state for 10 ms, and then clears the WKUP bit.
The USB Specification requires the USB idle state to be retained for at least 5 ms before the remote wakeup signal is sent. Therefore, even if the WKUP bit is set immediately after the suspended state is detected, this module waits for 2 ms, and then outputs the K state.

Writing 1 to the WKUP bit must be done only when the device is in the suspended state (DVSQ = 1xx) and remote wakeup is allowed by the USB host.
When setting the WKUP bit, do not stop the internal clock even if the device is in the suspended state. (Write 1 to the WKUP bit when the SUSPM bit is 1.)

When the WKUP bit is set at a transition to the L1 state, this module outputs the K state for 50 µs, and then clears the WKUP bit. In the L1 state, setting the WKUP bit must be done only when the DVSQ[4] bit is 1.

(2) Reset handshake status (RHST) bits

This module outputs the result of reset handshake to this bit. Table 33.7 lists the results of reset handshake.

Table 33.7 Reset Handshake Status

Bus State	Value of RHST Bit
Powered or disconnected state	000
Reset handshake in process	100
Full-speed connection	010
High-speed connection	011

If the HSE bit is 1, the RHST bits indicate 100 when this module detects a USB bus reset. Then, after this module has output Chirp K, these bits indicate 011 when this module detects Chirp JK from the USB host three times. If the status is not fixed to High-Speed within 2.5 ms after Chirp K is output, these bits indicate 010.

If the HSE bit is 0, the RHST bits indicate 010 when this module detects a bus reset.

After this module has detected a USB reset, a DVST interrupt occurs when the value of the RHST bits is fixed to 010 or 011.

33.2.6 Test Mode Register

33.2.6.1 USB Test Mode Register [TESTMODE] <Address: 00CH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	BRST 2	—	—	BRST 3	—	—	—	—	—	—	—	UTST			
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15	—	Nothing is assigned to this bit. Fix this bit to 0.	
14	BRST2	This bit is used in resumption in response to a bus reset. Set this bit according to 33.9.16.3, Flowchart of resumption from deep standby in response to a bus reset. Operation cannot be guaranteed if the procedure in the flowchart is not observed.	R/W
13, 12	—	Nothing is assigned to these bits. Fix these bits to 0.	
11	BRST3	This bit is used in resumption in response to a bus reset. Set this bit according to 33.9.16.3, Flowchart of resumption from deep standby in response to a bus reset. Operation cannot be guaranteed if the procedure in the flowchart is not observed.	R/W
10 to 4	—	Nothing is assigned to these bits. Fix these bits to 0.	
3 to 0	UTST	Test mode See the detailed description.	R/W

(1) Test mode (UTST) bits

When a value is written to these bits, this module outputs a USB test signal during high-speed operation.
Table 33.8 lists the test modes of this module.

Table 33.8 List of test mode operation

Test mode	Value of UTST Bits
Normal operation	0000
Test_J	0001
Test_K	0010
Test_SE0_NAK	0011
Test_Packet	0100
Test_Force_Enable	—
Reserved	0101 to 0111

Write a value to these bits according to the SetFeature request sent from the USB host during high-speed communication.
When these bits contain a value from 0001 to 0100, this module does not enter the suspended state.
To perform a normal USB communication after setting a test mode, perform Power on reset.

33.2.7 FIFO Port Registers

33.2.7.1 CFIFO Port Register [CFIFO] <Address: 014H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIFOPORT (Low)															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

33.2.7.2 CFIFO Port Register [CFIFO] <Address: 016H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIFOPORT (High)															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	FIFOPORT	FIFO port These bits are accessed to read received data from the FIFO buffer or to write send data to the FIFO buffer.	R/W

(1) FIFO port control bits (for FIFOPORT)

The send/receive buffer memory of this module has a FIFO structure (FIFO buffer). Use FIFO port registers to access the FIFO buffer. The FIFO port consists of the port register (CFIFO) to read data from and write data to the FIFO buffer, the register (CFIFOSEL) to select the pipe to be allocated to the FIFO port, and the control register (CFIFOCTR).

Individual FIFO ports have the following features:

1. The CFIFO port must be used to access the FIFO buffer through the DCP.
2. When functions specific to FIFO ports are used, the pipe number (selected pipe) specified in the CURPIPE bits cannot be changed.
3. The registers configured for a FIFO port do not affect any other FIFO ports.
4. The FIFO buffer memory can be accessed by either the CPU or SIE. Access by the CPU is not possible while the SIE has the right of access to the FIFO buffer memory.

(2) FIFO port bits (CFIFO)

When one of these registers is accessed, this module accesses the FIFO buffer allocated to the pipe number specified in the CURPIPE bits in the corresponding pipe select register (CFIFOSEL).

These registers can be accessed only when the FRDY bit of the respective control registers (CFIFOCTR) is 1 (or when the UCL_Dx_DREQ output is asserted by this module).
The valid bits of these registers vary depending on the values of the NBW and BIGEND bits. The valid bits are shown in Table 33.9 to Table 33.11.

Table 33.9 Endian Operation in 32-Bit Access (When MBW = 10)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	N + 3 address	N + 2 address	N + 1 address	N + 0 address
1	N + 0 address	N + 1 address	N + 2 address	N + 3 address

Table 33.10 Endian Operation in 16-Bit Access (When MBW = 01)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	Writing: invalid Reading: prohibited*1		•even-numbered address	odd-numbered address
1	even-numbered address	odd-numbered address	Writing: invalid Reading: prohibited*1	

Table 33.11 Endian Operation in 8-Bit Access (When MBW = 00)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	Writing: invalid Reading: prohibited*1			Writing: valid Reading: valid
1	Writing: valid Reading: valid	Writing: invalid Reading: prohibited*1		

Note: Reading words or bytes from an invalid register is prohibited.

33.2.7.3 CFIFO Port Select Register [CFIFOSEL] <Address: 020H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RCNT	REW	—	—	MBW		—	BIGEND	—	—	ISEL	—	CURPIPE			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15	RCNT	Read count mode This bit specifies DTLN read mode for the CFIFOCTR register. 0: Clears the DTLN bits when all received data is read. 1: Decrements the value of the DTLN bits each time received data is read.	R/W
14	REW	Buffer pointer rewind Set this bit to 1 to rewind the buffer pointer. 0: Does not rewind the buffer pointer. 1: Rewinds the buffer pointer.	R(0)/W
13, 12	—	Nothing is assigned to these bits. Fix these bits to 0.	
11, 10	MBW	CFIFO port access bit width This bit specifies the bit width for access to the CFIFO port. 00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited	R/W
9	—	Nothing is assigned to this bit. Fix this bit to 0.	
8	BIGEND	FIFO port byte endian control This bit specifies the byte endian of the CFIFO port. 0: Little endian 1: Big endian	R/W
7, 6	—	Nothing is assigned to these bits. Fix these bits to 0.	
5	ISEL	FIFO port access direction with DCP selected This bit specifies the FIFO port access direction when DCP is selected for the CURPIPE bits. 0: Selects reading of buffer memory. 1: Selects writing of buffer memory.	R/W
4	—	Nothing is assigned to this bit. Fix this bit to 0.	
3 to 0	CURPIPE	FIFO port access pipe specification This bit specifies the pipe number used for accessing the CFIFO port. 0000: DCP 0001: PIPE1 0010: PIPE2 ↓ 1000: PIPE8 1001: PIPE9 ↓ 1110: PIPE14 1111: PIPE15	R/W

(1) Read count mode (RCNT) bit

When this bit is 0, this module clears the DTLN bits in the CFIFOCTR register when all received data has been read from the FIFO buffer allocated to the pipe (selected pipe) specified in the CURPIPE bits (or, in the case of double-buffer configuration, when all relieved data has been read from one buffer).

With this bit is 1, this module decrements the value of the DTLN bits in the CFIFOCTR register each time received data is read from the FIFO buffer allocated to the specified pipe.

(2) Buffer pointer rewind (REW) bit

When this bit is set during data reading from the FIFO buffer while the selected pipe is in the receiving direction, reading can be restarted from the first data in the FIFO buffer (or, in the case of a double-buffer configuration, rereading can be started from the first data in the FIFO buffer being read).
Do not set this bit at the same time as changing the value of the CURPIPE bits. Before setting this bit, always check that the FRDY bit is 1.
If you want to redo writing to the FIFO buffer from the first data in the FIFO buffer when the selected PIPE is in the sending direction, use the BCLR bit.

(3) CFIFO port access bit width (MBW) bits

These bits are used to specify the bit width for the access to the CFIFO port.
If you start reading after setting a value in these bits when the pipe specified in the CURPIPE bits is in the receiving direction, do not change the value of these bits until data is all read.
Also, to set a value in these bits when the pipe specified in the CURPIPE bits is in the receiving direction, temporarily change the original value of the CURPIPE bits to a different value, and then set the values of the CURPIPE and MBW bits at the same time.
For how to change the value of the CURPIPE bits, see the description of the CURPIPE bits.
When the pipe specified in the CURPIPE bits is in the sending direction, you cannot change the bit width from 8 bits to 16 bits or 32 bits or from 16 bits to 32 bits while writing to the buffer memory is in process.
Even with the 16-bit width or 32-bit width setting, you can write data also to odd bytes by using byte access control.

(4) FIFO port byte endian control (BIGEND) bit

This bit is used to specify the byte endian of the CFIFO port.
For details see section 33.2.7.2 (1), FIFO port control bits (for FIFOPORT).

(5) FIFO port access direction with DCP selected (ISEL) bit

To change the value of this bit when the specified pipe is DCP, write a value to this bit, read the bit, and then check that the written value is the same as the read value before proceeding to the next processing.
If the value of the bit is changed in the middle of an access to the FIFO buffer, the access is held. Therefore, the same access can be resumed after these bits are restored to the original value.
Set this bit simultaneously with setting of the CURPIPE bits.

(6) FIFO port access pipe specification (CURPIPE) bits

These bits are used to specify the pipe number of the pipe through which to read or write data via the CFIFO port.
If you change the value of these bits, write a desired value to these bits, read these bits, and then check that the written value is the same as the read value before proceeding to the next processing.
If the value of these bits is changed in the middle of an access to the FIFO buffer, the access is held. Therefore, the same access can be resumed after these bits are restored to the original value.

33.2.7.4 D0FIFO Port Select Register [D0FIFOSEL] <Address: 028H>
D1FIFO Port Select Register [D1FIFOSEL] <Address: 02CH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RCNT	REW	DCLRM	DREQE	MBW		—	—	—	—	—	—	CURPIPE			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15	RCNT	Read count mode This bit specifies Dx_FIFOCTR DTLN read mode. 0: Clears the DTLN bits when all received data is read. 1: Decrements the value of the DTLN bits each time received data is read.	R/W
14	REW	Buffer pointer rewind Set this bit to 1 to rewind the buffer pointer. 0: Does not rewind the buffer pointer. 1: Rewinds the buffer pointer.	R(0)/W
13	DCLRM	Automatic buffer memory clear mode after reading data through the specified pipe This bit prohibits or enables automatic buffer memory clear after data is read through the specified pipe. 0: Automatic FIFO buffer clear prohibited 1: Automatic FIFO buffer clear enabled	R/W
12	DREQE	UCL_Dx_DREQ output enable This bit prohibits or enables the output of the UCL_Dx_DREQ signal. 0: Output prohibited 1: Output enabled	R/W
11, 10	MBW	DxFIFO port access bit width This bit specifies the bit width for access to the DxFIFO port. 00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited	R/W
9 to 4	—	Nothing is assigned to these bits. Fix these bits to 0.	
3 to 0	CURPIPE	FIFO port access pipe specification 0000: No specification 0001: PIPE1 0010: PIPE2 ↓ 1000: PIPE8 1001: PIPE9 ↓ 1110: PIPE14 1111: PIPE15	R/W

(1) Read count mode (RCNT) bit

When this bit is 0, this module clears the DTLN bits in the DxFIFOCTR register when all received data has been read from the FIFO buffer allocated to the pipe (selected pipe) specified in the CURPIPE bits (or, in the case of double-buffer configuration, when all relieved data has been read from one buffer).

With this bit is 1, this module decrements the value of the DTLN bits in the DxFIFOCTR register each time received data is read from the FIFO buffer allocated to the specified pipe.

(2) Buffer pointer rewind (REW) bit

When this bit is set during data reading from the FIFO buffer while the selected pipe is in the receiving direction, reading can be restarted from the first data in the FIFO buffer (or, in the case of a double-buffer configuration, rereading can be started from the first data in the FIFO buffer being read).
Do not set this bit at the same time as changing the value of the CURPIPE bits. Before setting this bit, always check that the FRDY bit is 1.
If you want to redo writing to the FIFO buffer from the first data in the FIFO buffer when the selected PIPE is in the sending direction, use the BCLR bit.

(3) Automatic FIFO buffer clear enable (DCLRM) bit

This bit is used to enable or disable the mode to automatically clear the FIFO buffer memory after reading data from the specified pipe. With this bit set, this module performs the "BCLR = 1" processing on the FIFO buffer, if a zero-length packet is received when the FIFO buffer allocated to the specified pipe is empty or if data reading ends because a short packet is received when the BFRE bit is 1.
Always clears this bit when you use this module with the BRDYM bit set.

(4) UCL_Dx_DREQ output enable (DREQE) bit

This bit is used to enable or disable the output of the UCL_Dx_DREQ signal.
When enabling the UCL_Dx_DREQ signal, set this bit always after setting a value in the CURPIPE bits.
When changing the value of the CURPIPE bits, change the value always after clearing this bit.

(5) DxFIFO port access bit width (MBW) bits

These bits are used to specify the bit width for the access to the DxFIFO port.
For details, see 33.2.7.3, (3), CFIFO port access bit width (MBW) bits.

(6) FIFO port access pipe specification (CURPIPE) bits

These bits are used to specify the pipe number through which to read or write data via the DxFIFO port.
If you change the value of these bits, write a desired value to these bits, read these bits, and then check that the written value is the same as the read value before proceeding to the next processing.
Do not specify the same pipe number for the CURPIPE bits in the CFIFOSEL, D0FIFOSEL, and D1FIFOSEL registers.
If the value of these bits is changed in the middle of an access to the FIFO buffer, the access is held. Therefore, the same access can be resumed after these bits are restored to the original value.

33.2.7.5 CFIFO Port Control Register [CFIFOCTR] <Address: 022H>
D0FIFO Port Control Register [D0FIFOCTR] <Address: 02AH>
D1FIFO Port Control Register [D1FIFOCTR] <Address: 02EH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BVAL	BCLR	FRDY	—	DTLN											
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15	BVAL	Buffer memory valid flag Specify 1 for this bit when writing to the FIFO buffer on the CPU side through the pipe specified in CURPIPE (current pipe) ends. 0: Invalid 1: Writing complete	R/W(1)
14	BCLR	CPU buffer clear Specify 1 for this bit to clear the FIFO buffer on the CPU side of the current pipe. 0: Invalid 1: CPU buffer memory clear	R(0)/W(1)
13	FRDY	FIFO port ready This bit indicates whether the FIFO port can be accessed. 0: The FIFO port cannot be accessed. 1: The FIFO port can be accessed.	R
12	—	Nothing is assigned to this bit. Fix this bit to 0.	
11 to 0	DTLN	Receive data length These bits indicate the length of receive data.	R

(1) Buffer memory valid flag (BVAL)

When the pipe (selected pipe) specified in the CURPIPE bits is in the sending direction, this bit must be set in the cases descried below. This module switches the FIFO buffer from the CPU side to the SIE side to enable data sending.

(1) To send short packets, set this bit when data writing ends.
(2) To send zero length packets, set this bit before writing data to the FIFO buffer.
(3) Set this bit after writing, to the pipe in continuous transfer mode, the data of which the size is a positive integral multiple of the maximum packet size and less than the buffer size.

When the maximum packet size of data is written to the pipe in non-continuous transfer mode, this module sets this bit to switch the FIFO buffer from the CPU side to the SIE side and enable data sending.
When this bit and the BCLR bit are set at the same time when the specified pipe is in the sending direction, this module clears the data that has been written so far and enables zero-length packets to be sent.
Setting this bit must be done only when the FRDY bit in the corresponding port control register is 1. If you want to check the value of the FRDY bit after setting this bit, wait at least 80 ns after setting this bit, and then reference the FRDY bit.
Do not set this bit when the specified pipe is in the receiving direction.

(2) CPU buffer clear (BCLR) bit

When this bit is set, this module clears the FIFO buffer on the CPU side among the FIFO buffers allocated to the specified pipe.
Even if the two FIFO buffers in a double-buffer configuration are allocated to the specified pipe and the both buffers can be read, this module clears only one of the two buffers.

If this bit is set when the specified pipe is the DCP, this module clears the FIFO buffer regardless of whether the FIFO buffer is on the CPU side or it is on the SIE side. To clear the buffer on the SIE side, set this bit always after setting the PID bits for DCP to "NAK".
When the specified pipe is other than the DCP, setting this bit must be done only when the FRDY bit in the corresponding port control register is 1. If you want to check the value of the FRDY bit after setting this bit, wait at least 80 ns after setting this bit, and then reference the FRDY bit.

(3) FIFO port ready (FRDY) bit

This bit indicates whether the FIFO port can be accessed from the CPU. This bit is operated by this module.
In the cases described below, even when this module sets this bit, data cannot be read from the FIFO port because the FIFO buffer does not contain data to be read. In these cases, set the BCLR bit to clear the FIFO buffer to enable the next sending and receiving of data.

(1) A zero-length packet has been received while the FIFO buffer allocated to the specified pipe is empty.
(2) A short packet has been received and data reading has ended while the BFRE bit is 1.

(4) Receive data length (DTLN) bits

These bits indicate the length of receive data. These bits are operated by this module. During reading of the FIFO buffer, the value of these bits varies depending on the value of the RCNT bit as described below.

(1) When the RCNT bit is 0:
This module indicates a receive data length by using these bits until the CPU ends reading all received data from one FIFO buffer.
When the BFRE bit is 1, this module retains the receive data length until the BCLR bit is set even if reading of received data has ended.
(2) When the RCNT bit is 1:
This module decrements the value of these bits each time the CPU reads data.
(The value is decremented by 1 when the MBW bits are 00, by 2 when the MBW bits are 01 or by 4 when the MBW bits are 10.)

When the CPU ends reading from one FIFO buffer, this module clears these bits. If reading of one of the FIFO buffers in a double-buffer configuration, however, ends before reading of received data from the other FIFO buffer ends, these bits indicate the receive data length for the other FIFO buffer at the end of reading from the FIFO buffer of which reading ends earlier.
When these bits are read during reading of the FIFO buffer when the RCNT bit is 1, this module updates the value of these bits within 150 ns after a cycle of read access to the corresponding FIFO port.

33.2.8 Interrupt Enable Registers (INTENBx, BRDYENB, NRDYENB, BEMPENB)

33.2.8.1 Interrupt Enable Register 0 [INTENB0] <Address: 030H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE	—	—	—	—	—	—	—	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15	VBSE	VBUS Interrupt Enable This bit prohibits or enables a USB interrupt when a VBINT interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled	R/W
14	RSME	Frame number update interrupt enable Resume interrupt enable This bit prohibits or enables a USB interrupt when a RESM interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled	R/W
13	SOFE	This bit prohibits or enables a USB interrupt when a SOF interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled	R/W
12	DVSE	Device state transition interrupt enable This bit prohibits or enables a USB interrupt when a DVST interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled	R/W
11	CTRE	Control transfer stage transition interrupt enable This bit prohibits or enables a USB interrupt when a CTRT interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled	R/W
10	BEMPE	Buffer empty interrupt enable This bit prohibits or enables a USB interrupt when a BEMP interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled	R/W
9	NRDYE	Buffer not ready response interrupt enable This bit prohibits or enables a USB interrupt when an NRDY interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled	R/W
8	BRDYE	Buffer ready interrupt enable This bit prohibits or enables a USB interrupt when a BRDY interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled	R/W
7 to 0	—	Nothing is assigned to these bits. Fix these bits to 0.	

33.2.8.2 BRDY Interrupt Enable Register [BRDYENB] <Address: 036H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIPEBRDYE															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 0	PIPEBRDYE	Pipe BRDY interrupt enable These bits prohibit or enable the BRDY bit to be set when a BRDY interrupt to a pipe is detected. 0: Interrupt output prohibited 1: Interrupt output enabled	R/W

Note: * Bit numbers correspond to pipe numbers.

(1) Pipe BRDY interrupt enable (PIPEBRDYE) bits

When this module detects a BRDY interrupt to the pipe for which is set 1 in this register, this module sets the corresponding PIPEBRDY bit in the BRDYSTS register, sets the BRDY bit in the INTSTS0 register, and asserts the interrupt.

When at least one of the PIPEBRDY bits in the BRDYSTS register is 1 and software changes the corresponding interrupts enable bit in this register changes from 0 to 1, this module asserts the interrupt.

33.2.8.3 NRDY Interrupt Enable Register [NRDYENB] <Address: 038H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIPENRDYE															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 0	PIPENRDYE	Pipe NRDY interrupt enable These bits prohibit or enable the NRDY bit to be set when a BRDY interrupt to a pipe is detected. 0: Interrupt output prohibited 1: Interrupt output enabled	R/W

Note: * Bit numbers correspond to pipe numbers.

(1) Pipe NRDY interrupt enable (PIPENRDYE) bits

When this module detects a BRDY interrupt to the pipe for which is set 1 in this register, this module sets the corresponding PIPENRDY bit in the NRDYSTS register, sets the NRDY bit in the INTSTS0 register, and asserts the interrupt.

When at least one of the PIPENRDY bits in the NRDYSTS register is 1 and the corresponding interrupts enable bit in this register changes from 0 to 1, this module asserts the interrupt.

33.2.8.4 BEMP Interrupt Enable Register [BEMPENB] <Address: 03AH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIPEBEMPE															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 0	PIPEBEMPE	Pipe BEMP interrupt enable These bits prohibit or enable the BEMP bit to be set when a BRDY interrupt to a pipe is detected 0: Interrupt output prohibited 1: Interrupt output enabled	R/W

Note: * Bit numbers correspond to pipe numbers.

(1) Pipe BEMP interrupt enable (PIPEBEMPE) bits

When this module detects a BEMP interrupt to the pipe for which is set 1 in this register, this module sets the corresponding PIPEBEMP bit in the BEMPSTS register, sets the BEMP bit in the INTSTS0 register, and asserts the interrupt.

When at least one of the PIPEBEMP bits in the BEMPSTS register is 1 and the corresponding interrupts enable bit in this register changes from 0 to 1, this module asserts the interrupt.

33.2.9 SOF Control Register

33.2.9.1 SOF Pin Configuration Register [SOFCFG] <Address: 03CH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	—	—	—	—	BRDYM	—	—	SOFM		—	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 7	—	Nothing is assigned to these bits. Fix these bits to 0.	
6	BRDYM	PIPEBRDY interrupt status clear timing This bit specifies the timing at which the PIPEBRDY interrupt is to be cleared. 0: Software clears the status. 1: Hardware clears the status by reading from or writing to the FIFO buffer. This bit can be set only during initialization (before communication). The setting cannot be changed after communication.	R/W
5, 4	—	Nothing is assigned to these bits. Fix these bits to 0.	
3, 2	SOFM	SOF function setting These bits are used to select SOF pulse output mode. 00: SOF output disabled 01: SOF output in units of 1 ms 10: µSOF output in units of 125 µs 11: Reserved	R/W
1, 0	—	Nothing is assigned to these bits. Fix these bits to 0.	

33.2.10 Interrupt Status

33.2.10.1 Interrupt Status Register 0 [INTSTS0] <Address: 040H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY	VBSTS	DVSQ			VALID	CTSQ		
0	0	0	0	0	0	0	0	—	0	0	0	0	0	0	0
—	—	—	1	—	—	—	—	—	0	0	1	—	—	—	—

Bit	Name	Function	R/W
15	VBINT	Change detection interrupt status This bit indicates the VBUS change detection interrupt status. 0: No VBUS interrupt is generated. 1: A VBUS interrupt is generated.	R/W(0)
14	RESM	Resume interrupt status This bit indicates the resume detection interrupt status. 0: No resume interrupt is generated. 1: A resume interrupt is generated.	R/W(0)
13	SOFR	Frame number update interrupt status This bit indicates the frame number update interrupt status. 0: No SOF interrupt is generated. 1: A SOF interrupt is generated.	R/W(0)
12	DVST	Device state transition interrupt status This bit indicates the device state transition interrupt. 0: No device state transition interrupt is generated. 1: A device state transition interrupt is generated.	R/W(0)
11	CTRT	Control transfer stage transition interrupt status This bit indicates the status of a control transfer stage transition interrupt. 0: No control transfer stage transition interrupt is generated. 1: A control transfer stage transition interrupt is generated.	R/W(0)
10	BEMP	BEMP interrupt status This bit indicates the BEMP interrupt status. 0: No BEMP interrupt is generated. 1: A BEMP interrupt is generated.	R
9	NRDY	NRDY interrupt status This bit indicates the NRDY interrupt status. 0: No NRDY interrupt is generated. 1: An NRDY interrupt is generated.	R
8	BRDY	BRDY interrupt status This bit indicates the BRDY interrupt status. 0: No BRDY interrupt is generated. 1: A BRDY interrupt is generated.	R
7	VBSTS	VBUS input status This bit indicates the VBUS pin input status. 0: The VBUS pin is at the low level. 1: The VBUS pin is at the high level.	R
6 to 4	DVSQ	Device state These bits indicate the device state. 000: Powered state 001: Default state 010: Address state 011: Configured state 1xx: Suspended state	R
3	VALID	USB request reception This bit indicates whether USB request reception is detected. 0: Not detected 1: A setup packet is received.	R/W(0)

Bit	Name	Function	R/W
2 to 0	CTSQ	Control transfer stage These bits indicate the control transfer stage. 000: Idle or setup stage 001: Control reading data stage 010: Control reading status stage 011: Control writing data stage 100: Control writing status stage 101: Control writing (No Data) status stage 110: Control transfer sequence error 111: Reserved	R

Note: * When you want to clear the status indicated by the VBINT, RESM, SOFR, DVST, or CTRT bit, write 0 to only the bit to be cleared and 1 to other bits. Do not write 0 to any status bit that is currently 0.

Note: * This module detects a status change indicated by the VBINT or RESM bit in this register even while the clock is stopped (the SUSPM bit is 0), and reports an interrupt corresponding to the status bit if the interrupt is enabled. Clearing the interrupt status must be done after the clock enabled.

(1) VBUS change interrupt status (VBINT) bit

This module sets this bit when it detects a change of the level of the VBUS pin input (from the high level to low level, or vice versa). This module indicates the level of the VBUS pin input by the VBSTS bit. When a VBINT interrupt occurs, read the VBSTS bit several times to check for consistency and remove chattering.

(2) Resume interrupt status (RESM) bit

This module sets this bit when it is in the suspended state (DVSQ bits are 1XX) and detects a falling edge of the signal at the DP pin.

(3) Frame number update interrupt status (SOFR) bit

This module sets this bit under the following conditions:
This module sets this bit when the frame number is updated. (The frame number update interrupt is monitored at intervals of 1 ms.)
This module detects an SOFR interrupt based on SOF interpolation even when an SOF packet from the USB host is damaged.

(4) Device state transition interrupt status (DVST) bit

When this module detects a change of the device state, this module updates the value of the DVSQ bits, and sets this bit. When a device state transition interrupt occurs, clear the interrupt status before this module detects the next device state transition.

(5) Control transfer stage transition interrupt status (CTRT) bit

When this module detects a transition of control transfer stage, this module updates the value of the CTSQ bits and sets this bit.
When a control transfer stage transition interrupt occurs, clear the interrupt status before this module detects the next transition of control transfer stage.

(6) Buffer empty interrupt status (BEMP) bit

This module sets this bit when at least one of the PIPEBEMP bits in the BEMPSTS register corresponding to the pipes for which the PIPEBEMPE bit in the BEMPENB register is set (that is, when this module detects a BEMP interrupt to at least one of the pipes for which is enabled BEMP interrupt notification).
For the conditions to assert the PIPEBEMP status signal, see the description of the BEMPSTS register.
This module clears this bit when writes 0 to all the PIPEBEMP bits corresponding to the pipes for which a BEMP interrupt has been enabled by setting the PIPEBEMPE bit.
Cannot clear this bit even by writing 0.

(7) Buffer not-ready interrupt status (NRDY) bit

This module sets this bit when at least one of the PIPENRDY bits in the BNRDYSTS register corresponding to the pipes for which the PIPENRDYE bit in the NRDYENB register is set (that is, when this module detects an NRDY interrupt to at least one of the pipes for which is enabled NRDY interrupt notification).
For the conditions to assert the PIPENRDY status signal, see the description of the NRDYSTS register.
This module clears this bit when writes 0 to all the PIPENRDY bits corresponding to the pipes for which a NRDY interrupt has been enabled by setting the PIPENRDYE bit.
Software cannot clear this bit even by writing 0.

(8) Buffer ready interrupt status (BRDY) bit

This module sets this bit when at least one of the PIPEBRDY bits in the BRDYSTS register corresponding to the pipes for which the PIPEBRDYE bit in the BRDYENB register is set (that is, when this module detects an BRDY interrupt to at least one of the pipes for which is enabled BRDY interrupt notification).
For the conditions to assert the PIPEBRDY status signal, see the description of the BRDYSTS register.
This module clears this bit when writes 0 to all the PIPEBRDY bits corresponding to the pipes for which a BRDY interrupt has been enabled by setting the PIPEBRDYE bit.
Software cannot clear this bit even by writing 0.

33.2.10.2 BRDY Interrupt Status Register [BRDYSTS] <Address: 046H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIPEBRDY															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 0	PIPEBRDY	Pipe BRDY interrupt status These bits indicate the BRDY interrupt status of each pipe. 0: No interrupt is generated. 1: An interrupt is generated.	R/W(0)

Note: * Bit numbers correspond to pipe numbers.
* To clear the interrupt status indicated by a bit in this register when the BRDYM bit is 0, write 0 to only the bit to be cleared and 1 to other bit.
* When the BRDYM bit is 0, clearing the BRDY interrupt status must be done always before the next access to the FIFO.

(1) Pipe BRDY interrupt status (PIPEBRDY) bit

When this module detects a BRDY interrupt to a pipe, this module sets the corresponding PIPEBRDY bit in the BRDYSTS register. At that time, if already set the corresponding bit in the BRDYENB register, this module sets the BRDY bit in the INTSTS0 register, and asserts the interrupt.
Conditions to generate and clear a BRDY interrupt vary depending on the values of the BRDYM bit, and the BFRE bit for each pipe.

(a) BRDYM = 0 and BFRE = 0

When the BRDYM and BFRE bits are 0, the BRDY interrupt is generated to indicate that the FIFO port has become ready for access.
Under the conditions described below, this module generates an internal BRDY interrupt request trigger, and sets the PIPEBRDY bit corresponding to the pipe for which the request trigger is generated.

1. For the pipe in the sending direction
   (a) When the DIR bit changes from 0 to 1
   (b) When this module has ended sending packets through a pipe when data writing by the CPU to the FIFO buffer allocated to that pipe is disabled (when the value read from the BSTS bit is 0)
   In continuous transfer mode, a request trigger is generated when data has all been sent from one FIFO buffer.
   (c) When, in a double-buffer configuration, one FIFO buffer is empty when writing to the other FIFO buffer has ended
   If sending to one FIFO buffer has ended during writing to the other FIFO buffer, no request trigger is generated until the ongoing writing to the other FIFO buffer ends.
   (d) When this module flushes the FIFO buffer allocated to the pipe of which the transfer type is isochronous.
   (e) When the state of the FIFO buffer is changed from the write-disabled state to the write-enabled state by writing 1 to the ACLRM bit

No request trigger is generated when the pipe is the DCP (in other words, when data is sent by a control transfer).

2. For the pipe in the receiving direction
   (a) When this module has ended receiving packets through a pipe and reading from the FIFO buffer is enabled when data reading by the CPU from the FIFO buffer allocated to that pipe is disabled (when the value read from the BSTS bit is 0)
   No request trigger is generated for the transaction that involves a data PID mismatch.
   In continuous transmission/reception mode, no request trigger is generated when the data size is the maximum packet size and the FIFO buffer still has a free space.
   If a short packet is received, a request trigger is generated even when the FIFO buffer has a free space.
   When the transaction counter is used, a request trigger is generated when the specified number of packets have been received.
   In that case, the request trigger is generated even if the FIFO buffer has a free space.
   (b) When, in a double-buffer configuration, one FIFO buffer is in the read-enabled state when reading from the other FIFO buffer has ended
   If receiving from one FIFO buffer has ended during reading from the other FIFO buffer, no request trigger is generated until the ongoing reading from the other FIFO buffer ends.

This interrupt does not occur during the communication at the status stage of a control transfer.

The PIPEBRDY interrupt status of the corresponding pipe can be cleared to "0" by writing "0" to the corresponding bit. When clearing a PIPEBRDY bit by writing 0, write 1 to all the PIPEBRDY bits corresponding to other pipes.
Clearing the pipe BRDT interrupt status must be done always before the next access to the FIFO buffer.

(b) When BRDYM = 0 and BFRE = 1

When the BRDYM bit is 0 and the BFRE bit is 1, this module determines that a BRDY interrupt occurs when all the data for a transfer has been read through a receiving pipe, and sets the PIPEBRDY bit corresponding to the pipe.

This module determines that the last data in a transfer has been received when one of the following conditions is met:
1. A short packet or a zero-length packet has been received.
2. The transaction counter (TRNCNT bits) is used, and as many packets as the value of the TRNCNT bits have been received.

When one of the above conditions is met and reading of the relevant data has ended, this module determines that all the data in a transfer has been read.

If a zero-length packet is received when the FIFO buffer is empty, this module determines that all the data in a transfer has been read when the FRDY bit is set and the DTLN bits are cleared in the corresponding FIFO Port Control Register. To start the next transfer in that case, write 1 to the BCLR bit in the corresponding FIFOCTR.

When the BRDYM bit is 0 and the BFRE bit is 1, this module does not detect any BRDY interrupt to the pipe in the sending direction.

The PIPEBRDY interrupt status of the corresponding pipe can be cleared to "0" by writing "0" to the corresponding bit. When clearing a PIPEBRDY bit by writing 0, write 1 to the PIPEBRDY bits corresponding to other pipes.

In this mode, do not change the value of the BFRE bit until all the processing for a transfer ends.
If you need to change the value of the BFRE bit during the transfer, set the ACLRM bit to clear all the FIFO buffers for the specified pipe.

(c) When BRDYM = 1 and BFRE = 0

When the BRDYM bit is 1 and the BFRE bit is 0, the values of individual PIPEBRDY bits interlock with the values of the BSTS bits for individual pipes. In other words, this module sets or clears the BRDY interrupt status of a pipe according to the state of the FIFO buffer allocated to the pipe.

(a) For the pipe in the sending direction
This module sets the PIPEBRDY bit for the pipe when data can be written to the FIFO port or clears the PIPEBRDY bit when data cannot be written to the FIFO port.
The BRDY interrupt signal, however, is not asserted even when the sending pipe is write-enabled if the pipe is the DCP.

(b) For the pipe in the receiving direction
This module sets the PIPEBRDY bit for the pipe when data can be read from the FIFO port or clears the PIPEBRDY bit when all data has been read (that is, when data reading from the FIFO port is disabled).
If a zero-length packet is received when the FIFO buffer is empty, the PIPEBRDY bit corresponding to the specified pipe is kept being set and the BRDY interrupt signal is kept being asserted until sets the BCLR bit.

When the BRDYM bit is 1 and the BFRE bit is 0, this module cannot clear any PIPEBRDY bit.

When the BRDYM bit is 1, all the BFRE bits (for all pipes) must be 0.

33.2.10.3 NRDY Interrupt Status Register [NRDYSTS] <Address: 048H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIPENRDY															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 0	PIPENRDY	Pipe NRDY interrupt status These bits indicate the NRDY interrupt status of each pipe. 0: No interrupt is generated. 1: An interrupt is generated.	R/W(0)

<<Remarks>>

* Bit numbers correspond to pipe numbers.
* To clear the interrupt status indicated by a bit in this register, write 0 only to the bit and 1 to all other bit.

(1) Pipe NRDY interrupt status (PIPENRDY) bit

When this module issues an internal NRDY interrupt request for a pipe of which the PID is set to BUF, this module sets the PIPENRDY bit corresponding to the pipe in the NRDYSTS register. At that time, if the NRDYENB register bit corresponding to the pipe is set, this module sets the NRDY bit in the INTSTS0 register, and asserts the interrupt.
This module issues an internal NRDY interrupt request for individual pipes under the conditions described below.
This module does not issue any interrupt request at the status stage of a control transfer.

(a) For the pipe in the sending direction

(1) When an IN token is received in a situation where the PID bits corresponding to the specified pipe are 01 (BUF) and the FIFO buffer does not contain send data
When an IN token is received, this module issues an NRDY interrupt request, and sets the PIPENRDY bit.
If the transfer type of the pipe for which the interrupt is generated is isochronous transfer, this module sends a zero-length packet, and sets the OVRN bit.

(b) For the pipe in the receiving direction

(1) When an OUT token is received in a situation where the PID bits corresponding to the specified pipe are 01 (BUF) and the FIFO buffer is full
If the transfer type of the pipe for which the interrupt is generated is isochronous transfer, this module issues an NRDY interrupt request, sets the PIPENRDY bit, and sets the OVRN bit.
If the transfer type of the pipe for which the interrupt is generated is not isochronous transfer, this module issues an NRDY interrupt request when sending an NAK Handshake signal after receiving the data that follows the OUT token, and sets the PIPENRDY bit.
Note, however, that this module does not issue an NRDY interrupt request when resending data (when a DATA-PID mismatch has occurred).
This module does not issue an NRDY interrupt request also when an error has occurred in a data packet.

(2) When a PING token is received in a situation where the PID bits corresponding to the specified pipe are 01 (BUF) and the FIFO buffer is full
When a PING token is received, this module issues an NRDY interrupt request, and sets the PIPENRDY bit.

(3) When the transfer type of the specified pipe is isochronous transfer, the PID bits corresponding to the specified pipe are 01 (BUF), and data has not been received normally within an interval frame
When an SOF token is received, this module issues an NRDY interrupt request, and sets the PIPENRDY bit for the specified pipe.

33.2.10.4 BEMP Interrupt Status Register [BEMPSTS] <Address: 04AH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIPEBEMP															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 0	PIPEBEMP	Pipe BEMP interrupt status These bits indicate the BEMP interrupt status of each pipe. 0: No interrupt is generated. 1: An interrupt is generated.	R/W(0)

<<Remarks>>

* Bit numbers correspond to pipe numbers.
* To clear the interrupt status indicated by a bit in this register, write 0 only to the bit and 1 to all other bit.

(1) Pipe BEMP interrupt status (PIPEBEMP) bit

When this module detects a BEMP interrupt to a pipe of which the PID is set to BUF, this module sets the PIPEBEMP bit corresponding to the pipe in the BEMPSTS register. At that time, if the BEMPENB register bit corresponding to the pipe is set, this module sets the BEMP bit in the INTSTS0 register, and asserts the interrupt.

This module issues an internal BEMP interrupt request for individual pipes under the conditions described below.

1. When data sending (including sending of zero-length packets) to a pipe in the sending direction has ended and the FIFO buffer allocated to the pipe is empty
   In a single-buffer configuration, this module generates a BRDY interrupt at the same time as issuing an internal BEMP interrupt request for the pipes other than the DCP.
   Note, however, that this module does not issue the internal BEMP interrupt request in the following cases:
   (a) When software (DMAC) has already started writing to the CPU-side FIFO buffer, in a double-buffer configuration, when sending data for one buffer ends
   (b) When the buffer is cleared (to empty the buffer) by writing 1 to the ACLRM or BCLR bit
   (c) During an IN transfer (sending zero-length packets) at the status stage of a control transfer

2. For the pipe in the receiving direction
   When the data larger than the specified maximum packet size has been received normally
   In that case, this module issues a BEMP interrupt request, sets the PIPEBEMP bit for the specified pipe, discards the received data, and changes the value of the PID bits for the specified pipe to 11 (STALL).
   Then, this module returns a STALL packet.
   Note, however, that this module does not issue the internal BEMP interrupt request in the following cases:
   (a) When a CRC or bit stuff error has been detected in the received data
   (b) When a SETUP transaction is being executed

Writing 0 to this bit clears the interrupt status.
Writing 1 to this bit causes no effect.

33.2.11 Frame Number Registers (FRMNUM, UFRMNUM)

33.2.11.1 Frame Number Register [FRMNUM] <Address: 04CH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVRN	CRCE	—	—	—	FRNM										
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15	OVRN	Overrun/underrun detection status This bit indicates whether an overrun or underrun is detected in the pipe being used to perform an isochronous transfer. 0: No error 1: An error occurred.	R/W(0)
14	CRCE	CRC error detection status This bit indicates the CRC error detection status for the pipe being used to perform an isochronous transfer. 0: No error 1: An error occurred.	R/W(0)
13 to 11	—	Nothing is assigned to these bits. Fix these bits to 0.	
10 to 0	FRNM	Frame number These bits indicate the latest frame number.	R

Note: * The OVRN bit is intended for debugging. When designing a system, design transfer timings appropriately to prevent buffer overruns and underruns.

(1) Overrun/underrun detection status (OVRN) bit

This module sets this bit when this module detects an overrun or underrun in a pipe of which the transfer type is isochronous transfer.
When this module detects an overrun or underrun, this module issues an internal NRDY interrupt request. For details, see section 33.2.10.3, (1), Pipe NRDY interrupt status (PIPENRDY) bit.

Software can clear this bit by writing 0 to this bit. If the CRCE bit should not be cleared together when this bit is cleared, write 0x40.

When the peripheral controller function is selected
This module sets this bit in the following cases:

1. When an IN token is received although writing send data to the FIFO has not ended, in the case of a pipe that is in the sending direction and performs an isochronous transfer
2. When an OUT token is received although the free space of FIFO buffer is less than the size of one FIFO buffer, in the case of a pipe that is in the sending direction and performs an isochronous transfer

(2) CRC error detection status (CRCE) bit

This module sets this bit when this module detects a CRC or bit stuff error in a pipe of which the transfer type is isochronous transfer.
Software can clear this bit by writing 0 to this bit. If the OVRN bit should not be cleared together when this bit is cleared, write 0x80.
When this module detects a CRC error, this module issues an internal NRDY interrupt request. For details, see section 33.2.10.3, (1), Pipe NRDY interrupt status (PIPENRDY) bit.

(3) Frame number (FRNM) bits

This module updates the value of these bits each time an SOF packet is received (once per 1 ms), and indicates the latest frame number in these bits.
When reading these bits, read them twice and check for consistency.

33.2.11.2 Micro Frame Number Register [UFRMNUM] <Address: 04EH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DVCHG	—	—	—	—	—	—	—	—	—	—	—	—	UFRNM		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15	DVCHG	0: Writing to the USBADDR register is disabled 1: Writing to the USBADDR register is enabled This bit is used in resumption in response to reception of the resume signal. Set this bit and the USBADDR register according to 33.9.16.4, Flowchart of resumption from deep standby in response to reception of the resume signal. Operation cannot be guaranteed if the procedure in the flowchart is not observed.	R/W
14 to 3	—	Nothing is assigned to these bits. Fix these bits to 0.	
2 to 0	UFRNM	Micro frame These bits indicate the micro frame number.	R

(1) Micro frame number (UFRNM) bit

In high-speed transfer mode, this module writes the micro frame number to these bits. In a mode other than high-speed transfer mode, this module writes 0x00 to these bits.

When reading these bits, read them twice and check for consistency.

33.2.12 USB Address

33.2.12.1 USB Address Register [USBADDR] <Address: 050H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	STSRECOV0			—	USBADDR						
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0

Bit	Name	Function	R/W
15 to 11	—	Nothing is assigned to these bits. Fix these bits to 0.	
10 to 8	STSRECOV0	Status Recovery Bits	R/W

b10	b9	b8	Description
0	0	1	FS default state
0	1	0	FS address state
0	1	1	FS configured state
1	0	1	HS default state
1	1	0	HS address state
1	1	1	HS configured state
1	0	0	Suspend state

Set these bits according to section 33.9.16.4, Flowchart of resumption from deep standby in response to reception of the resume signal.
Operation cannot be guaranteed if the procedure in the flowchart is not observed.

Bit	Name	Function	R/W
7	—	Nothing is assigned to this bit. Fix this bit to 0.	
6 to 0	USBADDR	USB address These bits indicate the USB address allocated by the host.	R

(1) USB address (USBADDR) bits

When this module has received and normally processed a SetAddress request, this module writes the received USB address to these bits.
When this module detects a USB bus reset, this module writes 0x00 to these bits.

33.2.13 USB Request Registers

USB request registers are used to store control transfer setup requests.
These registers store the values set in the received USB requests.

33.2.13.1 USB Request Type Register [USBREQ] <Address: 054H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
bRequest								bmRequestType							
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Name	Function	R/W
15 to 8	bRequest	Request Value of USBRequestbRequest	R
7 to 0	bmRequestType	Request type Value of USBRequestbmRequestType	R

(1) USB request (bRequest) bits

These bits indicate the value of the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

(2) USB request type (bRmRequestType) bits

These bits indicate the value of the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

33.2.13.2 USB Request Value Register [USBVAL] <Address: 056H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
wValue															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Name	Function	R/W
15 to 0	wValue	Value Value of the wValue field in a USB request	R

(1) Value (wValue) bits

These bits indicate the value of the wValue field in a USB request.
These bits indicate the value of wValue field in the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

33.2.13.3 USB Request Index Register [USBINDX] <Address: 058H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
wIndex															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Name	Function	R/W
15 to 0	wIndex	Index Value of the wIndex field in a USB request	R

(1) Index (wIndex) bits

These bits indicate the value of the wIndex field in a USB request.
These bits indicate the value of wIndex field in the USB request data this module has received in a SETUP transaction.
Writing to these bits is ignored.

33.2.13.4 USB Request Length Register [USBLENG] <Address: 05AH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
wLength															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Name	Function	R/W
15 to 0	wLength	Length Value of the wLength field in a USB request	R

(1) Length (wLength) bits

These bits indicate the value of the wLength field in a USB request.
These bits indicate the value of wLength field in the USB request data this module has received in a SETUP transaction.
Writing to these bits is ignored.

33.2.14 DCP Configuration

When performing a data communication by a control transfer, use the default control pipe (DCP).

33.2.14.1 DCP Configuration Register [DCPCFG] <Address: 05CH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	—	—	CNTMD	SHTNAK	—	—	—	—	—	—	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 9	—	Nothing is assigned to these bits. Fix these bits to 0.	
8	CNTMD	Continuous transfer mode This bit specifies whether to use the default control pipe to perform communication in continuous transfer mode. 0: Non-continuous transfer mode 1: Continuous transfer mode	R/W
7	SHTNAK	Pipe disable at transfer end When the default control pipe is in the receiving direction, this bit specifies whether to change the PID setting to NAK when the transfer ends. 0: Continues the pipe when the transfer ends 1: Disables the pipe when the transfer ends	R/W
6 to 0	—	Nothing is assigned to these bits. Fix these bits to 0.	

33.2.14.2 DCP Max. Packet Size Register [DCPMAXP] <Address: 05EH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	—	—	—	—	MXPS						
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 7	—	Nothing is assigned to these bits. Fix these bits to 0.	
6 to 0	MXPS	Maximum packet size These bits specify the maximum data payload size (maximum packet size) for the DCP.	R/W

(1) Maximum packet size (MXPS) bits

These bits are used to specify the maximum data payload size (maximum packet size) for the DCP.
The default is 0x40 (64 bytes).

The value of the MXPS bits must conform to the USB Specification.
Writing a value to the MXPS bits must be done when the PID is set to NAK and no value is set in the CURPIPE bits.
If you need to change the value of these bits after changing the PID setting for the specified pipe from BUF to NAK, check the PBUSY bit is 0 before changing the value of these bits. If, however, this module has changed the PID setting from BUF to NAK, the value of the PBUSY bit need not be checked.

When the MXPS bits are all 0, data must not be written to the FIFO buffer, and the PID setting must not be changed to BUF.

33.2.14.3 DCP Control Register [DCPCTR] <Address: 060H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BSTS	—	—	—	—	—	—	SQCLR	SQSET	SQMON	PBUSY	—	—	CCPL	PID	
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0

Bit	Name	Function	R/W
15	BSTS	Buffer status This bit indicates the accessibility status of the DCP FIFO buffer. 0: Buffer access is not possible. 1: Buffer access is possible.	R
14 to 9	—	Nothing is assigned to these bits. Fix these bits to 0.	
8	SQCLR	Toggle bit clear This bit can set DATA0 as the expected value of the sequence toggle bit for the next transaction in DCP transfer. 0: Writing disabled 1: DATA0 specified	R(0)/W(1)
7	SQSET	Toggle bit set This bit can set DATA1 as the expected value of the sequence toggle bit for the next transaction in DCP transfer. 0: Writing disabled 1: DATA1 specified	R(0)/W(1)
6	SQMON	Sequence toggle bit monitor This bit indicates the expected value of the sequence toggle bit for the next transaction in DCP transfer. 0: DATA0 1: DATA1	R
5	PBUSY	Pipe busy This bit indicates whether the specified pipe is being used in the USB bus. 0: The specified pipe is not used in the USB bus. 1: The specified pipe is being used in the USB bus.	R
4 to 3	—	Nothing is assigned to these bits. Fix these bits to 0.	
2	CCPL	Control transfer end enable Setting this bit permits the status stage of the control transfer to end. 0: Does not permit the control transfer to end. 1: Permits the control transfer to end.	R/W
1, 0	PID	Response PID These bits control responses from this module in control transfer. 00: NAK response 01: BUF response (according to the buffer status) 10: STALL response 11: STALL response	R/W

(1) Buffer status (BSTS) bit

This bit indicates whether the CPU can access the FIFO buffer allocated to the DCP. This bit is operated by this module. The meaning of this bit varies as follows depending on the value of the ISEL bit:

1. When ISEL = 0: This bit indicates whether receive data can be read from the FIFO buffer.
2. When ISEL = 1: This bit indicates whether send data can be written to the FIFO buffer.

(2) Sequence toggle bit clear (SQCLR) bit

When software sets this bit, this module specifies DATA0 as the expected value of the sequence toggle bit for the specified pipe. This module always clears this bit.

Do not set the SQCLR and SQSET bits at the same time.

Setting this bit must be done when the PID is set to NAK and no value is set in the CURPIPE bits.
If you need to set this bit after changing the PID setting for the specified pipe from BUF to NAK, check, that the PBUSY bit is 0 before setting this bit. If, however, this module has changed the PID setting from BUF to NAK, the value of the PBUSY bit need not be checked.

(3) Sequence toggle bit set (SQSET) bit

When software sets this bit, this module specifies DATA1 as the expected value of the sequence toggle bit for the specified pipe. This module always clears this bit.
Do not set the SQCLR and SQSET bits at the same time.
Setting this bit must be done when the PID is set to NAK and no value is set in the CURPIPE bits.
If you need to set this bit after changing the PID setting for the specified pipe from BUF to NAK, check, that the PBUSY bit is 0 before setting this bit. If, however, this module has changed the PID setting from BUF to NAK, the value of the PBUSY bit need not be checked.

(4) Sequence toggle bit monitor (SQMON) bit

This bit indicates the expected value of the sequence toggle bit for the specified pipe. This bit is operated by this module.
When a transaction ends normally, this module toggles this bit.
This module, however, does not toggle this bit if a DATA-PID mismatch occurs during a transfer in the receiving direction.
When a SETUP packet is received normally, this module sets this bit (to specify DATA1 as the expected value).
This module does not reference this bit in an IN or OUT transaction at the status stage. Also, this module does not toggle this bit even when the transaction ends normally.

(5) Pipe busy (PBUSY) bit

This module changes this bit from 0 to 1 when a USB transaction using the specified pipe starts. This module changes this bit from 1 to 0 when the transaction ends.
Reading this bit after software sets the PID to NAK enables you to check whether you can change pipe settings.

(6) Control transfer end enable (CCPL) bit

When software sets this bit when the PID of the specified pipe is BUF, this module ends the status stage of the ongoing control transfer.
In other words, in a control read transfer, this module sends an ACK handshake in response to an OUT transaction request from the USB host, and, in a control write or no-data control transfer, this module sends a zero-length packet in response to an IN transaction request from the USB host. If, however, a SetAddress request is detected, this module performs automatic response throughout the period from the setup stage to the end of the status stage regardless of the value of this bit.
When a new SETUP packet is received, this module changes this bit from 1 to 0.
When the VALID bit is 1, software cannot set this bit.

(7) Response PID (PID) bits

The setting of these bits must be changed from NAK to BUF when the data stage or status stage of a control transfer is executed.

This module changes the value of these bits in the following cases:

1. This module changes the value of these bits to 00 (NAK) when it receives a SETUP packet. At that time this module sets the VALID bit. Software cannot change the value of these bits until it clears the VALID bit.
2. When set these bits to 01 (BUF), this module changes the value of these bits to 11 (STALL) when it receives the data exceeding the specified maximum packet size.
3. This module changes the value of these bits to 1x (STALL) when it detects a sequence error in a control transfer.
4. This module changes the value of these bits to 00 (NAK) when it detects a USB reset.

This module does not reference these bits during SetAddress request processing (automatic processing).

33.2.15 Pipe Configuration Registers (PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI)

To configure pipes 1 to 15, use the PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI, PIPExCTR, PIPExTRE, and PIPExTRN registers.

Select the pipes to be used by using the PIPESEL register, and then configure functions of individual pipes by using the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers. Note that you can use the PIPExCTR, PIPExTRE, and PIPExTRN registers for setting regardless of the pipe selection by the PIPESEL register.

33.2.15.1 Pipe Window Select Register [PIPESEL] <Address: 064H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	—	—	—	—	—	—	—	PIPESEL			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 4	—	Nothing is assigned to these bits. Fix these bits to 0.	
3 to 0	PIPESEL	Pipe window select These bits specify a pipe for registers at addresses 68H to 6EH. 0000: No selection 0001: PIPE1 0010: PIPE2 0011: PIPE3 0100: PIPE4 0101: PIPE5 0110: PIPE6 0111: PIPE7 1000: PIPE8 1001: PIPE9 1010: PIPE10 1011: PIPE11 1100: PIPE12 1101: PIPE13 1110: PIPE14 1111: PIPE15	R/W

* When the PIPESEL bits are 0000, all bits of the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers are cleared. When the PIPESEL bits are 0000, writing to the registers at addresses 68H to 6EH is ignored.

(1) Pipe window select (PIPESEL) bits

When software a value from 0001 to 1111 to these bits, this module indicate the pipe information and settings corresponding to the registers at addresses 68H to 6EH. After a pipe is selected by these bits, the values set in the areas at addresses 68H to 6EH are applied, by this module, to the transfer operation using the selected pipe.

When writes 0000 to these bits, this module writes 0 to all bits of the registers at addresses H68 to H6E. Then, writing to the areas at addresses H68 to H6E is ignored.

33.2.15.2 Pipe Configuration Register [PIPECFG] <Address: 068H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TYPE		—	—	—	BFRE	DBLB	CNTMD	SHTNAK	—	—	DIR	EPNUM			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15, 14	TYPE	Transfer type These bits specify the transfer type of the pipe specified in the PIPESEL bit. 00: The pipe cannot be used. 01: Bulk transfer 10: Interrupt transfer 11: Isochronous transfer	R/W
13 to 11	—	Nothing is assigned to these bits. Fix these bits to 0.	
10	BFRE	BRDY interrupt operation specification This bit specifies the timing at which this module notifies a BRDY interrupt relating to the specified pipe. 0: A BRDY interrupt is notified when data is sent or received. 1: A BRDY interrupt is notified when reading of data is completed.	R/W
9	DBLB	Double-buffer mode This bit specifies a single or double FIFO buffer to be used by the specified pipe. 0: Single buffer 1: Double buffer	R/W
8	CNTMD	Continuous transfer mode This bit specifies whether to use the specified pipe to perform communication in continuous transfer mode. 0: Non-continuous transfer mode 1: Continuous transfer mode	R/W
7	SHTNAK	Pipe disable at transfer end When the specified pipe is in the receiving direction, this bit specifies whether to change the PID setting to NAK when the transfer ends. 0: Continues the pipe when the transfer ends. 1: Disables the pipe when the transfer ends.	R/W
6, 5	—	Nothing is assigned to these bits. Fix these bits to 0.	
4	DIR	Transfer direction This bit specifies the transfer direction of the specified pipe. 0: Receiving direction 1: Sending direction	R/W
3 to 0	EPNUM	Endpoint number These bits specify the endpoint number of the specified pipe.	R/W

(1) Transfer type (TYPE) bits

These bits are used to specify the USB transfer type of the pipe (selected pipe) specified in the PIPESEL bits.
Table 33.12 lists pipes and the transfer types specifiable in these bits.

Table 33.12 Selected Pipes and the Transfer Types Specifiable in the TYPE Bits

Selected Pipe	TYPE Bits	USB Transfer Type
PIPE1 or PIPE2	01 or 11	Bulk or isochronous transfer
PIPE3 to PIPE5	01	Bulk transfer
PIPE6 to PIPE8	10	Interrupt transfer
PIPE9	01 or 10	Bulk or interrupt transfer
PIPE10	01 or 10	Bulk or interrupt transfer
PIPE10 to PIPE15	01	Bulk transfer

Always specify a value other than 00 in these bits for a selected pipe before setting the PID of the selected pipe to BUF (to start USB communication using the selected pipe).

Changing the value of these bits for a selected pipe must be done while the PID of the selected pipe is NAK. If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(2) BRDY interrupt operation specification (BFRE) bit

This bit is valid when the selected PIPE is PIPE1 to PIPE5 or PIPE9 to PIPE15.

When set this bit and been using the selected pipe in the receiving direction (in other words, the DIR bit is 0), this module detects the end of transfer (when it occurs) and generates a BRDY interrupt when reading of the last packet ends.
If a BRDY interrupt occurs with the above settings, software must write 1 to the BCLR bit. The FIFO buffer allocated to the selected pipe remains unready for reception until 1 is written to the BCLR bit.

When set this bit and been using the selected pipe in the sending direction (in other words, the DIR bit is 1), this module does not generate any BRDY interrupt.
For details, see the description of the PIPEBRDY interrupt status bit.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.
If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(3) Double-buffer mode (DBLB) bit

This bit is valid when the selected pipe is PIPE1 to PIPE5 or PIPE9 to PIPE15.

When set this bit for a selected pipe, this module allocates, to the selected pipe, two FIFO buffers, each of which has the FIFO buffer size specified in the BUFSIZE bits in the PIPEBUF register.
The size of the FIFO buffer this module allocates to the selected pipe is as follows:
(BUFSIZE + 1) × 64 × (DBLB + 1) [bytes]

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.
If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(4) Continuous transfer mode (CNTMD) bit

This bit is valid when the selected pipe is PIPE1 to PIPE5 or PIPE9 to PIPE15, and the transfer type of the selected pipe is bulk transfer.
This module determines whether data sending from or receiving in the FIFO buffer allocated to the selected pipe has ended according to the value of this bit in the way described in Table 33.14.

Table 33.13 How to Determine the End of Data Sending from or Receiving in the FIFO Buffer According to the Value of the CNTMD Bit

CNTMD Bit Setting Value	How to Determine Whether Reading or Sending is Enabled
0	Condition for enabling reading from the FIFO buffer when the receiving direction is set ("DIR = 0"): This module receives one packet
	Condition for enabling sending from the FIFO buffer when the sending direction is set ("DIR = 1"): Either of the following conditions (1) and (2) is met: (1) Data for the maximum packet size is written to the FIFO buffer. (2) Data for the short packet (including the case of zero-byte data) is written to the FIFO buffer and then 1 is written in the BVAL bit.
1	Conditions for enabling reading from the FIFO buffer when the receiving direction is set ("DIR = 0"): (1) The number of bytes in data received in the FIFO buffer allocated to the selected pipe becomes equal to the number of allocated bytes ((BUFSIZE + 1) * 64). (2) This module receives a short packet other than a zero-length packet. (3) This controller receives a zero-length packet when the FIFO buffer allocated to the selected pipe already contains data. (4) Packets are received as many times as the value of the transaction counter set for the selected pipe.
	Condition for enabling sending from the FIFO buffer when the sending direction is set ("DIR = 1"): One of the following conditions (1) to (3) is met: (1) The amount of written data becomes equal to the size of one FIFO buffer allocated to the selected pipe. (2) Data for less than the size of one FIFO buffer allocated to the selected pipe (including the case of zero-byte data) is written to the FIFO buffer and then 1 is written in the BVAL bit. (3) Data for less than the size of one FIFO buffer allocated to the selected pipe (including the case of zero-byte data) is written to the FIFO buffer, and a transfer end signal is asserted at the same time of the last writing.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.
If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.
If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(5) Pipe disable at transfer end (SHTNAK) bit

This bit is valid when the selected pipe is PIPE1 to PIPE5 or PIPE9 to PIPE15 and is in the receiving direction.
When set this bit for a selected pipe in the receiving direction, this module changes the PID of the selected pipe to NAK when this module determines the end of data transfer to the selected pipe. This module determines the end of transfer when one of the following conditions (1) and (2) is met:

(1) This module has normally received short packet data (including zero-length packets).
(2) When using a transaction counter, this module has normally received as many packets as the value set in the transaction counter.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

For the pipes in the sending direction, this bit must be cleared.

(6) Transfer direction (DIR) bit

When writes 0 to this bit for a selected pipe, this module uses the selected pipe in the receiving direction. When software writes 1 to this bit for the selected pipe, this module uses the selected pipe in the sending direction.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.
If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(7) Endpoint number (EPNUM) bits

These bits are used to specify the endpoint number of the endpoint of a selected pipe.
Note that specifying 0000 in these bits for a pipe means that the pipe is not used.

Changing the value of these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

The combination of the values of the DIR bit and EPNUM bits for a pipe must be unique among those for all pipes. (The setting "EPNUM = 000" [the selected pipe is not used] can be duplicated for multiple pipes.)

33.2.15.3 Pipe Buffer Setting Register [PIPEBUF] <Address: 06AH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	BUFSIZE							BUFNMB							
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15	—	Nothing is assigned to this bit. Fix this bit to 0.	
14 to 10	BUFSIZE	Buffer size These bits specify the size of the FIFO buffer for the pipe specified in the PIPESEL bit. 0x00: 64 bytes 0x01: 128 bytes ... (0x1F: 2 Kbytes)	R/W
9, 8	—	Nothing is assigned to these bits. Fix these bits to 0.	
7 to 0	BUFNMB	Buffer number These bits specify the FIFO buffer number of the specified pipe. (0x4 to 0x7F)	R/W

Note: * Changing values of these register bits for a selected pipe must be done when sets the PID of the selected pipe to NAK, and specifies no pipe in the CURPIPE bits.

Note: * If you change values of these register bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the values of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(1) Buffer size (BUFSIZE) bits

These bits are used to specify the size of the FIFO buffer to be allocated to the selected pipe.
Specify the FIFO buffer size in units of blocks. One block has 64 bytes.
When set the DBLB bit for a selected pipe, this module allocates, to the selected pipe, two FIFO buffers, each of which has the FIFO buffer size specified in these bits.
The size of the FIFO buffer this module allocates to the selected pipe is as follows:
(BUFSIZE + 1) × 64 × (DBLB + 1) [bytes]

The following value can be specified in these bits:
(1) Any value from 0x0 to 0x1F when the selected pipe is PIPE1 to PIPE5 or PIPE9 to PIPE15.
(2) 0x0 only when the selected pipe is PIPE6 to PIPE8.
In continuous transfer mode (CNTMD = 1), specify an integral multiple of the maximum packet size in the BUFSIZE bits.

(2) Buffer number (BUFNMB) bits

These bits are used to specify the block number of the first block in the FIFO buffer to be allocated to the selected pipe.
This module allocates the following blocks of FIFO buffer to the selected pipe:
Block with block number "BUFNMB" to the block with block number "BUFNMB + (BUFSIZE + 1) × (DBLB + 1) - 1"

The value of these bits must be 0x04 to 0x7F. Note, however, that the following rules must be observed:
Value "0x00" is exclusively used for DCP.
Value "0x04" is exclusively used for PIPE6. When, however, PIPE6 is not used, this value can be used for another pipe. If PIPE6 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets 0x04 in the BUFNMB bits for PIPE6.
Value "0x05" is exclusively used for PIPE7. When, however, PIPE7 is not used, this value can be used for another pipe. If PIPE7 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets 0x05 in the BUFNMB

bits for PIPE7.
Value "0x06" is exclusively used for PIPE8. When, however, PIPE8 is not used, this value can be used for another pipe. If PIPE8 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets 0x06 in the BUFNMB bits for PIPE8.
Value "0x07" is exclusively used for PIPE9. When, however, PIPE9 is not used, this value can be used for another pipe. If PIPE9 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets 0x07 in the BUFNMB bits for PIPE9.

33.2.15.4 Pipe Maximum Packet Size Register [PIPEMAXP] <Address: 06CH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	MXPS										
0	0	0	0	0	0	0	0	0	0 (1)	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 11	—	Nothing is assigned to these bits. Fix these bits to 0.	
10 to 0	MXPS	Maximum packet size These bits specify the maximum data payload size (maximum packet size) for the specified pipe. For PIPE6 to PIPE8, a value from 0x1 to 0x40 (bytes) can be set.	R/W

Note: * The initial value of the MXPS bits is 0x00 when no pipe is specified in the PIPESEL bits in the PIPESEL register or 0x40 when a pipe is specified in the PIPESEL bits.

(1) Maximum packet size (MXPS) bits

These bits are used to specify the maximum data payload size (maximum packet size) for the selected pipe.
The initial value of these bits is 0x40 (64 bytes).

For PIPE1 and PIPE2, a value from 0x1 (1 byte) to 0x400 (1024 bytes) can be specified.
For PIPE3 to PIPE5, 0x8 (8 bytes), 0x10 (16 bytes), 0x20 (32 bytes), 0x40 (64 bytes), or 0x200 (512 bytes) can be specified. (Bits [2:0] are excluded.)
For PIPE6 to PIPE8, a value from 0x1 (1 byte) to 0x40 (64 bytes) can be specified.
For PIPE9, 0x8 (8 bytes), 0x10 (16 bytes), 0x20 (32 bytes), 0x40 (64 bytes), or 0x200 (512 bytes) can be specified.
For PIPE10 to PIPE15, 0x8 (8 bytes), 0x10 (16 bytes), 0x20 (32 bytes), 0x40 (64 bytes), or 0x200 (512 bytes) can be specified. (Bits [2:0] are excluded.)

The value of the MXPS bits for individual transfer type must conform to the USB Specification.

Setting a value in these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.
If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check the PBUSY bit is 0 before changing the value of thee bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.
When the MXPS bits are all 0, data must not be written to the FIFO buffer, and the PID setting must not be changed to BUF.

33.2.15.5 Pipe Cycle Control Register [PIPEPERI] <Address: 06EH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	IFIS	—	—	—	—	—	—	—	—	—	IITV		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 13	—	Nothing is assigned to these bits. Fix these bits to 0.	
12	IFIS	Isochronous IN buffer flush This bit specifies whether to perform a buffer flush when the pipe specified in the PIPESEL bit is used for isochronous IN transfer. 0: Does not perform a buffer flush. 1: Performs a buffer flush.	R/W
11 to 3	—	Nothing is assigned to these bits. Fix these bits to 0.	
2 to 0	IITV	These bits specify the transfer interval of the specified pipe. The value to be specified is the frame timing multiplied by an n-th power of 2.	R/W

(1) Isochronous IN buffer flush (IFIS) bit

When the selected pipe is used for isochronous IN transfer, this bit is used to specify this module automatically clears the FIFO buffer if this module fails to receive the IN token from the USB host in a (micro) frame sent at intervals specified in the IITV bits.
In double-buffer mode (DBLB = 1), this module clears only the data in one buffer used earlier than the other.
This module clears the FIFO buffer when it receives an SOF packet immediately after the (micro) frame in which the IN token hag to be received. Even if the SOF packet is corrupted, this module clears the FIFO buffer in the same timing to receive the SOF packet by the use of the internal interpolation function.

(2) Interval error detection interval (IITV) bits

These bits specify the interval of interval error detection for the selected pipe. The value to be specified is the frame timing multiplied by an n-th power of 2.
Setting a value in these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.
If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of thee bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.
If you change the value specified in these bits to another after a USB communication, change the PID to NAK, and then set the ACLRM bit to initialize the interval timer before changing the value.
For PIPE3 to PIPE5 and PIPE10 to PIPE15, these bits are ignored. Write 0 to all these bits corresponding to PIPE3 to PIPE5 and PIPE10 to PIPE15.
You can specify a value in these bits when the transfer type of the selected pipe is isochronous.

(a) When the selected pipe is used for isochronous OUT transfer

If this module does not receive any data packet in the (micro) frame sent at intervals specified in the IITV bits, this module generates an NRDT interrupt.
This module generates an NRDY interrupt also if this module cannot receive data because an error, e.g., CRC error, has occurred in a data packet or because the FIFO buffer is full (such a situation might result if, for example, software [DMAC] delays in reading data from the FIFO buffer).

This module generates the NRDY interrupt when it receives an SOF packet. Even if the SOF packet is corrupted, this module generates the NRDY interrupt in the same timing to receive the SOF packet by the use of the internal interpolation function.
When, however, the value of the IITV bits is not 0, this module generates the NRDY interrupt every time an SOF packet is received at the specified intervals after interval counting starts.

If the PID of the selected pipe is changed to NAK after the interval timer starts, this module does not generate the NRDY interrupt even when it receives an SOF packet.

The condition for starting interval counting varies by the value of the IITV bits.

(a) When IITV = 0: Interval counting starts when the PID of the selected pipe is changed to BUF.

(Micro) frame	SOF	SOF	SOF	OUT	DATA0	SOF	OUT	DATA0
Setting of PID bits	NAK	NAK	BUF			BUF		
Whether token reception is expected (0: reception expected -: non-reception expected)	-	-	0			0		
Start of interval counting			↑					

Figure 33.1 Relationship between (Micro) Frames and Whether Token Reception Is Expected When IITV = 0

(b) When IITV is not 0: Interval counting starts at the end of the first normal reception of data packet after the PID of the selected pipe is changed to BUF.

	SOF	SOF	SOF	OUT	DATA0	SOF	SOF	OUT	DATA0	SOF	SOF	OUT	DATA0
Setting of PID bits	NAK	BUF	BUF			BUF	BUF			BUF	BUF		
Whether token reception is expected (0: reception expected -: non-reception expected)	-	-	0			-	0			-	0		
Start of interval counting			↑										

Figure 33.2 Relationship between (Micro) Frames and Whether Token Reception Is Expected When IITV = 1

(b) When the selected pipe is used for isochronous IN transfer

The IITV bits are used in combination with the setting of the IFIS bit to 1. When the IFIS bit is 0, this module sends a data packet in response to a received token regardless of the value of the IITV bits.

When the IFIS bit is 1, if this module does not receive any IN token in the (micro) frame sent at intervals specified in the IITV bits although the FIFO buffer has sendable data, this module clears the FIFO buffer.
This module clears the FIFO buffer also when it cannot receive an IN token normally because of a bus error, e.g., CRC error.
This module clears the FIFO buffer when it receives an SOF packet. Even if the SOF packet is corrupted, this module clears the FIFO buffer in the same timing to receive the SOF packet by the use of the internal interpolation function.

The condition for starting interval counting varies by the value of the IITV bits. (The condition is the same as that for isochronous OUT transfer.)

The interval counter is cleared when one of the following conditions (1) to (3) is met:
(1) This module is reset (then, also the IITV bits are cleared).
(2) The ACLRM bit is set.
(3) This module detects a USB bus reset.

33.2.16 Pipe Control Registers (PIPExCTR)

33.2.16.1 PIPE1 Control Register [PIPE1CTR] <Address: 070H>
PIPE2 Control Register [PIPE2CTR] <Address: 072H>
PIPE3 Control Register [PIPE3CTR] <Address: 074H>
PIPE4 Control Register [PIPE4CTR] <Address: 076H>
PIPE5 Control Register [PIPE5CTR] <Address: 078H>
PIPE9 Control Register [PIPE9CTR] <Address: 080H>
PIPEA Control Register [PIPEACTR] <Address: 082H>
PIPEB Control Register [PIPEBCTR] <Address: 084H>
PIPEC Control Register [PIPECCTR] <Address: 086H>
PIPED Control Register [PIPEDCTR] <Address: 088H>
PIPEE Control Register [PIPEECTR] <Address: 08AH>
PIPEF Control Register [PIPEFCTR] <Address: 08CH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BSTS	INBUFM	—	—	—	ATREPM	ACLRM	SQCLR	SQSET	SQMON	PBUSY	—	—	—	PID	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0

Bit	Name	Function	R/W
15	BSTS	Buffer status This bit indicates the FIFO buffer status of the specified pipe. 0: Buffer access is not possible. 1: Buffer access is possible.	R
14	INBUFM	Transmit buffer monitor When the specified pipe is in the sending direction, this bit indicates the FIFO buffer status of the specified pipe. 0: The FIFO buffer does not contain data that can be sent. 1: The FIFO buffer contains data that can be sent.	R
13 to 11	—	Nothing is assigned to these bits. Fix these bits to 0.	
10	ATREPM	Automatic response mode This bit prohibits or enables automatic response of the specified pipe. 0: Automatic response prohibited 1: Automatic response enabled. (A zero-length packet response is sent during transmission. For reception, an NAK response is sent and an NRDY interrupt is generated.)	R/W
9	ACLRM	Automatic buffer clear mode This bit prohibits or enables automatic buffer clear mode for the specified pipe. 0: Prohibited 1: Enabled (all buffers are initialized)	R/W
8	SQCLR	Toggle bit clear Specify 1 in this bit to clear the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA0. 0: Writing disabled 1: DATA0 specified	R(0)/W(1)
7	SQSET	Toggle bit set Specify 1 in this bit to set the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA1. 0: Writing disabled 1: DATA1 specified	R(0)/W(1)
6	SQMON	Sequence toggle bit monitor This bit indicates the expected value of the sequence toggle bit for the next transaction in the specified pipe. 0: DATA0 1: DATA1	R

Bit	Name	Function	R/W
5	PBUSY	Pipe busy This bit indicates whether the specified pipe is being used in the USB bus. 0: The specified pipe is not used in the USB bus. 1: The specified pipe is being used in the USB bus.	R
4 to 2	—	Nothing is assigned to these bits. Fix these bits to 0.	
1, 0	PID	Response PID These bits specify the response method for the next transaction in the specified pipe. 00: NAK response 01: BUF response (according to the buffer status) 10: STALL response 11: STALL response	R/W

(1) Buffer status (BSTS) bit

This bit indicates whether the CPU can access the FIFO buffer allocated to the selected pipe. This bit is operated by this module.

The meaning of this bit varies as follows depending on the value of the values of the DIR, BFRE, and DCLRM bits:

Table 33.14 BSTS Bit Operations

DIR bit Setting Value	BFRE bit Setting Value	DCLRM bit Setting Value	Meaning of the BSTS Bit
0	0	0	This bit indicates 1 when reading of received data from the FIFO buffer becomes possible, and indicates 0 when reading data has finished.
		1	Setting prohibited
	1	0	This bit indicates 1 when reading of received data from the FIFO buffer becomes possible. This bit indicates 0 when 1 is written in the BCLF bit after reading data has finished.
		1	This bit indicates 1 when reading of received data from the FIFO buffer becomes possible, and indicates 0 when reading data has finished.
1	0	0	This bit indicates 1 when writing of send data in the FIFO buffer becomes possible, and indicates 0 when writing data has finished.
		1	Setting prohibited
	1	0	Setting prohibited
		1	Setting prohibited

(2) Transmit buffer monitor (INBUFM) bit

When the selected pipe is in the sending direction (DIR = 1), this module sets this bit when software (or the DMAC) has finished writing data to at least one FIFO buffer.

This module clears this bit when this module finishes sending all data from the FIFO buffer to which the data has been written. In double-buffer mode (DBLB = 1), this module clears this bit when this module has finished sending all data from the two FIFO buffers and software (or the DMAC) has not yet finished writing data to one FIFO buffer.

When the selected pipe is in the receiving direction (DIR = 0), this bit indicates the same value as that of the BSTS bit.

(3) Automatic response mode (ATREPM) bit

This bit can be set when the transfer type of the selected pipe is bulk transfer.
When this bit is 1, this module responds to tokens sent from the USB host as described below.

1. When the selected pipe is used for bulk IN transfer (TYPE = 01 and DIR = 1)

When the ATREPM bit is 1 and the PID of the selected pipe is BUF, this module responds to an IN token by sending a zero-length packet.
Each time this module receives ACK from the USB host (the sequence of one transaction is receiving an IN token, sending a zero-length packet, and then receiving ACK), this module updates (toggles) the sequence toggle bit (DATA-PID).
This module does not generate BRDY and BEMP interrupts.

2. When the selected pipe is used for bulk OUT transfer (TYPE = 01 and DIR = 0)

When the ATREPM bit is 1 and the PID of the selected pipe is BUF, this module responds to an OUT token (or a PING token) by sending an NAK response and generates an NRDY interrupt.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK.
If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

To perform a USB communication with this bit set, the FIFO buffer must be empty. No data must be written to the FIFO buffer during the USB communication with this bit set.

When the transfer type of the selected pipe is isochronous transfer, this bit must always be 0.

(4) Automatic buffer clear mode (ACLRM) bit

When you need to clear the whole FIFO buffer allocated to the selected pipe, write 1 and 0 successively to the ACLRM bit.
Table 33.15 shows the buffer contents this module clears when 1 and 0 are written successively to the ACLRM bit.
Table 33.16 shows the cases that require this processing.

Table 33.15 Buffer Contents This Core Clears When the ACLRM Bit is Set

No.	Contents to be Cleared by Setting the ACLRM Bit
(1)	Whole contents of the FIFO buffer allocated to the specified pipe (if the double buffer is set, both FIFO buffers are cleared)
(2)	If the transfer type of the specified pipe is Isochronous transfer, the interval count value is cleared.

Table 33.16 Cases Requiring the ACLRM Bit to be Set

No.	Cases when Data Clear is Required
(1)	The whole contents of the FIFO buffer allocated to the specified pipe needs to be cleared.
(2)	The interval count value needs to be reset.
(3)	The value of the BFRE bit is changed.
(4)	The value of the DBLB bit is changed.
(5)	Forced termination of the transaction count function is performed.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(5) Sequence toggle bit clear (SQCLR) bit

When software sets this bit, this module specifies DATA0 as the expected value of the sequence toggle bit for the selected pipe. This module always clears this bit.

Writing 1 to the SQCLR bit must be done when the PID of the selected pipe is NAK.
If you write 1 to this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(6) Sequence toggle bit set (SQSET) bit

When software sets this bit, this module specifies DATA1 as the expected value of the sequence toggle bit for the selected pipe. This module always clears this bit.
Writing 1 to the SQSET bit must be done when the PID of the selected pipe is NAK.
If you write 1 to this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(7) Sequence toggle bit monitor (SQMON) bit

This bit indicates the expected value of the sequence toggle bit for the selected pipe. This bit is operated by this module.

When the transfer type of the selected pipe is other than isochronous transfer, this module toggles this bit when a transaction ends normally. This module, however, does not toggles this bit if a DATA-PID mismatch occurs during a transfer in the receiving direction.

(8) Pipe busy (PBUSY) bit

This module changes this bit from 0 to 1 when a USB transaction using the selected pipe starts. This module changes this bit from 1 to 0 when the transaction ends normally.
Reading this bit after software sets the PID to NAK enables you to check whether you can change pipe settings.

(9) Response PID (PID) bits

These bits are used to specify, the type of response of this module for individual pipes.
The default of PID is NAK. When the selected pipe is used for USB transfers, the PID setting muse be changed to BUF.
For the basic operations (without communication packet errors involved) of this module depending on the value of the PID bits, see Table 33.17.
If you have changed the PID of a selected pipe from BUF to NAK while the selected pipe is performing a USB communication, check, that the PBUSY bit is 0 to confirm that the USB transfer through the selected pipe has actually changed to the NAK status. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

This module changes the value of the PID bits in the following cases:

1. When the selected pipe is in the receiving direction and set the SHTNAK bit for the selected pipe, this module sets the PID to NAK when this module recognizes the end of a transfer.
2. When this module has received a data packet of which the payload size is larger than the maximum packet size, this module sets the PID to STALL (PID = 11).
3. If this module detects a USB bus reset, this module sets the PID to NAK.

To change the PID from NAK (PID = 00) to STALL, write 10 to the PID bits.
To change the PID from BUF (PID = 01) to STALL, write 11 to the PID bits.
To change the PID from STALL (PID = 11) to NAK, write 10 to the PID bits once, and then write 00 to the PID bits.
To change the PID from STALL to BUF, change the PID to NAK once, and then change it to BUF.

Table 33.17 Core Operations Depending on the PID Setting

PID bit Setting Value	Transfer Type (TYPE Bit Setting Value)	Transfer Direction (DIR Bit Setting Value)	Operation of This Core
00 (NAK)	Bulk ("TYPE = 01"), or Interrupt ("TYPE = 10")	Independent of the setting value	Sends a NAK response for a token from the USB host.
	Isochronous ("TYPE = 11")	Receiving direction ("DIR = 0")	Does not respond to a token from the USB host.
		Sending direction ("DIR = 1")	Sends a zero-length packet for a token from the USB host.
01 (BUF)	Bulk ("TYPE = 01")	Receiving direction ("DIR = 0")	For an OUT token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module receives data and then sends an ACK or NYET response. If receiving data is not possible, this module sends an NAK response. For a PING Token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module sends an ACK response. If receiving data is not possible, this module sends an NAK response.
	Interrupt ("TYPE = 10")	Receiving direction ("DIR = 0")	For an OUT token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module receives data and then sends an ACK response. If receiving data is not possible, this module sends an NAK response.
	Bulk ("TYPE = 01") or Interrupt ("TYPE = 10")	Sending direction ("DIR = 1")	If the corresponding FIFO buffer is available for sending data, this module sends data in response to a token from the USB host. If sending data is not possible, this module sends an NAK response.
	Isochronous ("TYPE = 11")	Receiving direction ("DIR = 0")	For an OUT token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module receives data. If receiving data is not possible, this module discards data.
		Sending direction ("DIR = 1")	If the corresponding FIFO buffer is available for sending data, this module sends data in response to a token from the USB host. If sending data is not possible, this module sends a zero-length packet.
10 (STALL) or 11 (STALL)	Bulk ("TYPE = 01") or Interrupt ("TYPE = 10")	Independent of the setting value	Sends a STALL response for a token from the USB host.
	Isochronous ("TYPE = 11")	Independent of the setting value	Does not respond to a token from the USB host.

33.2.16.2 PIPE6 Control Register [PIPE6CTR] <Address: 07AH>
PIPE7 Control Register [PIPE7CTR] <Address: 07CH>
PIPE8 Control Register [PIPE8CTR] <Address: 07EH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BSTS	—	—	—	—	—	ACLRM	SQCLR	SQSET	SQMON	PBUSY	—	—	—	PID	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0

Bit	Name	Function	R/W
15	BSTS	Buffer status This bit indicates the FIFO buffer status of the specified pipe. 0: Buffer access is not possible. 1: Buffer access is possible.	R
14 to 10	—	Nothing is assigned to these bits. Fix these bits to 0.	
9	ACLRM	Automatic buffer clear mode This bit prohibits or enables automatic buffer clear mode for the specified pipe. 0: Automatic buffer clear mode prohibited 1: Automatic buffer clear mode enabled (all buffers are initialized)	R/W
8	SQCLR	Toggle bit clear Specify 1 in this bit to clear the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA0. 0: Disabled 1: DATA0 specified	R(0)/W(1)
7	SQSET	Toggle bit set Specify 1 in this bit to set the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA1 0: Disabled 1: DATA1 specified	R(0)/W(1)
6	SQMON	Toggle bit monitor This bit indicates the expected value of the sequence toggle bit for the next transaction in the specified pipe. 0: DATA0 1: DATA1	R
5	PBUSY	Pipe busy This bit indicates whether the specified pipe is being used in the USB bus. 0: The specified pipe is not used in the USB bus. 1: The specified pipe is being used in the USB bus.	R
4 to 2	—	Nothing is assigned to these bits. Fix these bits to 0.	
1, 0	PID	Response PID These bits specifies the response method for the next transaction in the specified pipe 00: NAK response 01: BUF response (according to the buffer status) 10: STALL response 11: STALL response	R/W

(1) Buffer status (BSTS) bit

See section 33.2.16.1, (2), Transmit buffer monitor (INBUFM) bit.

(2) Automatic buffer clear mode (ACLRM) bit

When you need to clear the whole FIFO buffer allocated to the selected pipe, write 1 and 0 successively to the ACLRM bit.
Table 33.18 shows the buffer contents this module clears when 1 and 0 are written successively to the ACLRM bit.
Table 33.19 shows the cases that require this processing.

Table 33.18 Buffer Contents This Core Clears when the ACLRM Bit is Set

No.	Contents Cleared by ACLRM Bit Operation
(1)	All contents of the FIFO buffer allocated to the selected pipe

Table 33.19 Cases Requiring the ACLRM Bit to be Set

No.	Contents Cleared by ACLRM Bit Operation
(1)	When clearing contents of the FIFO buffer allocated to the selected pipe
(2)	When resetting the interval counter
(3)	When the value of the BFRE bit is changed
(4)	When the transaction count function is terminated forcibly

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.
If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(3) Sequence toggle bit clear (SQCLR) bit

See section 33.2.16.1, (5), Sequence toggle bit clear (SQCLR) bit.

(4) Sequence toggle bit set (SQSET) bit

See section 33.2.16.1, (6), Sequence toggle bit set (SQSET) bit.

(5) Sequence toggle bit monitor (SQMON) bit

See section 33.2.16.1, (7), Sequence toggle bit monitor (SQMON) bit.

(6) Pipe busy (PBUSY) bit

See section 33.2.16.1, (8), Pipe busy (PBUSY) bit.

(7) Response PID (PID) bits

See section 33.2.16.1, (9), Response PID (PID) bits.

33.2.17 Transaction Counters (PIPExTRE)

33.2.17.1 PIPE1 Transaction Counter Enable Register [PIPE1TRE] <Address: 090H>
PIPE2 Transaction Counter Enable Register [PIPE2TRE] <Address: 094H>
PIPE3 Transaction Counter Enable Register [PIPE3TRE] <Address: 098H>
PIPE4 Transaction Counter Enable Register [PIPE4TRE] <Address: 09CH>
PIPE5 Transaction Counter Enable Register [PIPE5TRE] <Address: 0A0H>
PIPEB Transaction Counter Enable Register [PIPEBTRE] <Address: 0A4H>
PIPEC Transaction Counter Enable Register [PIPECTRE] <Address: 0A8H>
PIPED Transaction Counter Enable Register [PIPEDTRE] <Address: 0ACH>
PIPEE Transaction Counter Enable Register [PIPEETRE] <Address: 0B0H>
PIPEF Transaction Counter Enable Register [PIPEFTRE] <Address: 0B4H>
PIPE9 Transaction Counter Enable Register [PIPE9TRE] <Address: 0B8H>
PIPE9 Transaction Counter Enable Register [PIPEATRE] <Address: 0BCH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	—	TRENB	TRCLR	—	—	—	—	—	—	—	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 10	Nothing is assigned to these bits. Fix these bits to 0.		
9	TRENB Transaction counter enable	This bit enables or disables the transaction counter. 0: Disables the transaction counter. 1: Enables the transaction counter.	R/W
8	TRCLR Transaction counter clear	This bit clears the transaction counter to 0. To clear the counter, write 1 to this bit. 0: Invalid 1: Clears the current transaction counter.	R(0)/W(1)
7 to 0	Nothing is assigned to these bits. Fix these bits to 0.		

Note: * Changing values of these register bits for a selected pipe must be done when the PID of the selected pipe is NAK.
If you change the value of a bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of the bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(1) Transaction counter enable (TRENB) bit

When software sets this bit for the selected pipe in the receiving direction after specifying a total number of packets in the TRNCNT bits, this module performs the following control when it finishes receiving the same number of packets as the total number specified in the TRNCNT bits:

1. When the continuous transfer mode is used (CNTMD = 1), this module switches the FIFO buffer to the CPU side at the end of reception even if the FIFO buffer is not full.
2. When the SHTNAK bit is 1, this module changes the PID of the selected pipe to NAK when it finishes receiving the same number of packets as the total number specified in the TRNCNT bits.
3. When the DENDE bit is 1 and the PKTMD bit is 0, this module asserts the DEND signal when reading the last data after having received the same number of packets as the total number specified in the TRNCNT bits.
4. When the BFRE bit is 1, this module asserts the BRDY interrupt signal when it finishes reading the last data after having received the same number of packets as the total number specified in the TRNCNT bits.

For the pipe in the sending direction, write 0 to this bit (TRENB bit).
When not using the transaction count function, write 0 to this bit.
When using the transaction count function, specify a value in the TRNCNT bits before writing 1 to this bit. Also, write 1 to this bit before receiving the first packet among those to be counted by the transaction count function.

(2) Transaction counter clear (TRCLR) bit

When software sets this bit for a selected pipe, this module clears the current value of the transaction counter corresponding to the selected pipe, and then clears this bit.

33.2.17.2 PIPE1 Transaction Counter Register [PIPE1TRN] <Address: 092H>
PIPE2 Transaction Counter Register [PIPE2TRN] <Address: 096H>
PIPE3 Transaction Counter Register [PIPE3TRN] <Address: 09AH>
PIPE4 Transaction Counter Register [PIPE4TRN] <Address: 09EH>
PIPE5 Transaction Counter Register [PIPE5TRN] <Address: 0A2H>
PIPEB Transaction Counter Register [PIPEBTRN] <Address: 0A6H>
PIPEC Transaction Counter Register [PIPECTRN] <Address: 0AAH>
PIPED Transaction Counter Register [PIPEDTRN] <Address: 0AEH>
PIPEE Transaction Counter Register [PIPEETRN] <Address: 0B2H>
PIPEF Transaction Counter Register [PIPEFTRN] <Address: 0B6H>
PIPE9 Transaction Counter Register [PIPE9TRN] <Address: 0BAH>
PIPEA Transaction Counter Register [PIPEATRN] <Address: 0BEH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRNCNT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 0	TRNCNT	When writing: Specifies the total number of packets to be received by the pertinent pipe (number of transactions). When reading: Indicates the specified number of transactions if TRENB is 0. Indicates the number of the currently counted transaction if TRENB is 1.	R/W

(1) Transaction counter (TRNCNT) bits

When software writes 1 to the TRENB bit after setting the total number of packets to be received for the selected pipe in the receiving direction, this module performs the control described in section 33.2.17.1 Transaction counter enable (TRENB) bit.

When the TRENB bit is 0, this module indicates, by these bits, the number of transactions set.

When the TRENB bit is 1, this module indicates, by these bits, the current number of transactions counted.

This module increments the value of the TRNCNT bits by 1 when the status of reception meets all the following conditions (a) to (c):

(a) The TRENB bit is 1.

(b) When a packet is received, the value of the TRCNT bits is not equal to "current count + 1."

(c) The payload size of received packets has reached the value of the MXPS bits.

This module clears the TRNCNT bits to 0 when any of the following conditions (1) to (3) is met:

(1) All the following conditions (a) to (c) are met:

(a) The TRENB bit is 1.

(b) When a packet is received, the value of the TRCNT bits is equal to "current count + 1."

(c) The payload size of received packets has reached the value of the MXPS bits.

(2) Both of the following conditions (a) and (b) are met:

(a) The TRENB bit is 1.

(b) A short packet has been received.

(3) The following condition is met:
(a) Written 1 to the TRCLR bit.

For the pipe in the sending direction, write 0 to these bits (TRNCNT bits).
When not using the transaction count function, write 0 to these bits.

Changing the value of these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the TRENB bit is 0.
If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.
If you change the value of these bits, write 1 to the TRCLR bit before writing 1 to the TRENB bit.

33.2.18 Low Power Control Register

33.2.18.1 Low Power Control Register [LPCTRL] <Address: 100H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	—	—	—	HWUPM	—	—	—	—	—	—	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 8	Reserved	Nothing is assigned to these bits. Fix these bits to 0.	
7	HWUPM	0: Resumes the PHY from the low-power mode while the internal bus clock (Bφ) is operating. 1: Enables resume from the low-power mode while the internal bus clock (Bφ) is stopped.	R/W
6 to 0		Nothing is assigned to these bits. Fix these bits to 0.	

(1) HWUPM

This bit is used to specify whether to enable resumption from the low-power mode even while the internal bus clock (Bφ) is stopped.

0: Disables resumption while the internal bus clock (Bφ) is stopped.
1: Enables resumption while the internal bus clock (Bφ) is stopped.

This bit specifies whether to detect resume signaling while the internal bus clock (Bφ) is stopped. Whether to resume is controlled by the L1EXTMD bit. To resume from the low-power mode (LPM L1 state) while the internal bus clock (Bφ) is stopped, set both this bit and the L1EXTMD bit.

33.2.19 Low Power Status Register

33.2.19.1 Low Power Status Register [LPSTS] <Address: 102H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	SUSPM	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15	—	Nothing is assigned to this bit. Fix this bit to 0.	
14	SUSPM	USBPHY Suspend M control This bit controls the Suspend M signal to the USBPHY. 0: USBPHY suspend mode 1: USBPHY normal mode	R/W
13 to 0	—	Nothing is assigned to these bits. Fix these bits to 0.	

(1) USBPHY SuspendM control (SUSPM) bit

This bit is used to control the SuspendM signal to the USBPHY. By default, the value of this bit is 0 and the USBPHY is in suspend mode. To operate this module, write 1 to this bit.

When the SUSPM bit is 0 (that is, when the UTMI clock is stopped), data cannot be written to this module, but can only be read from this module. Note, however, that data can be written to the registers listed in Table 33.20.

Table 33.20 Registers That Allow Writing when the SUSPM Bit is 0

Address	Register Name
000H	SYSCFG0
002H	BUSWAIT
100H	LPCTRL
102H	SUSPMODE

Note that the values written to the SYSCFG0 register while the USBPHY clock is stopped (SUSPM = 0) will be applied after the USBPHY clock starts (SUSPM = 1).
When the L1EXTMD bit is 0, this bit (SUSPM bit) is controlled (set or cleared) by software. When L1EXTMD bit is 1, this bit is controlled by software for the transition to the L1 or L2 state, and controlled by hardware for resumption from the L1 or L2 state, regardless of the level (L1 or L2).

33.2.20 PHY Function Control Register

33.2.20.1 PHY Function Control Register [PHYFUNCTR] <Address: 104H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	SusMon	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15	—	Nothing is assigned to this bit. Fix this bit to 0.	
14	SusMon	This bit allows reading of the status of the Suspend M signal.	R
13 to 0	—	Nothing is assigned to these bits. Fix these bits to 0.	

(1) SusMon

SuspendM monitor bit (read only)

The status of the Suspend M signal can be read.

33.2.21 PHY_OTG Control Register (PHYOTGCTR)

33.2.21.1 PHY_OTG Control Register [PHYOTGCTR] <Address: 10AH>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	DmPuDwn	DpPuDwn	—	—	—	—	—	—	—	—	—
0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 11	—	Nothing is assigned to these bits. Fix these bits to 0.	
10	DmPuDwn	Dm Pulldown monitor bit	R
9	DpPuDwn	Dp Pulldown monitor bit	R
8 to 0	—	Nothing is assigned to these bits. Fix these bits to 0.	

(1) DmPuDwn

DmPulldown monitor bit (read only)

0: 15-kΩ Pulldown resistor control on the DM side is disabled.

1: 15-kΩ Pulldown resistor control on the DM side is enabled.

(2) DpPuDwn

DpPulldown monitor bit (read only)

0: 15-kΩ Pulldown resistor control on the DP side is disabled.

1: 15-kΩ Pulldown resistor control on the DP side is enabled.

33.2.22 Peripheral L1 Control Register 1 (PL1CTRL)

33.2.22.1 Peripheral L1 Control Register 1 [PL1CTRL1] <Address: 144H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	L1EXTMD	—	—	HIRDTHR[3:0]				DVSQ[3:0]				L1NEGOMD	L1RESPMD[1:0]		L1RESPEN
0	0	0	0	0				0	0	0	0	0	0	0	0
—	—	—	—	—				—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15	—	Nothing is assigned to this bit. Fix this bit to 0.	
14	L1EXTMD	USBPHY control mode on resumption from the L1 state This bit controls the USBPHY resume operation on resumption from the L1 state. 0: Does not set the Suspend M bit when the Host K signal is received. 1: Sets the Suspend M bit when the Host K signal is received	R/W
13, 12	—	Nothing is assigned to these bits. Fix these bits to 0.	
11 to 8	HIRDTHR[3:0]	L1 response negotiation threshold HIRD threshold to be used for the L1NEGOMD bit The format is the same as the HIRD field of the HL1CTRL register.	R/W
7	DVSQ[3]	DVSQ extension bit This bit combined with the device state (DVSQ[2:0]) bit indicates the L1 state. 4'b 0000: Powered state 4'b 0001: Default state 4'b 0010: Address state 4'b 0011: Configured state 4'b 01xx: Suspended state 4'b 10xx: L1 state	R
6 to 4	DVSQ[2:0]	These bits mirror the DVSQ[2:0] bits in INTSTS0.	R
3	L1NEGOMD	L1 response negotiation control This bit sets the negotiation function using for the HIRD value. 0: Returns an ACK response if the received HIRD value is larger than the value of the HIRDTHR[3:0] bits. In other cases (including same values), an NYET response is returned. 1: Returns an ACK response if the received HIRD value is smaller than the value of the HIRDTHR[3:0] bits. In other cases (including same values), an NYET response is returned. This bit is valid only when the value of the L1RESPMD[1:0] bit is 2'b11.	R/W
2, 1	L1RESPMD[1:0]	L1 response mode These bits specify how to respond to an LPM token. 2'b 00: NYET 2'b 01: ACK 2'b 10: STALL 2'b 11: Response according to the value of the L1NEGOMD bit	R/W
0	L1RESPEN	L1 response enable This bit enables an L1 response. 0: Does not support LPM. 1: Supports LPM.	R/W

(1) L1 EXT mode (L1EXTMD) bit

This bit specifies how to control the SuspendM bit upon receiving the Host K signal when the USBPHY is stopped by setting the SuspendM bit in the L1 state.

0: Does not set the SuspendM bit when this module is resumed from the L1 state.

1: Sets the SuspendM bit when this module is resumed from the L1 state.

Note 1. The Host K period lasts a minimum of 50 µs. Therefore, the USBPHY might be unable to be resumed within the Host K period if the software settings for resume same as those for the suspend state are applied. Because the initial value of this bit is controlled by software, set this bit at initialization when the L1 state is to be supported.
Note 2. For the transition to the L1 state, the SuspendM bit is controlled by software regardless of the value of this bit.
Note 3. When this bit is set, the SuspendM bit will be set also at resumption from the L2 state.

(2) HIRD negotiation threshold (HIRDTHR[3:0]) bits

These bits specify the value of HIRD threshold to be used for the negotiation specified by the L1NEGOMD bit.
The format of the value is the same as that of the HIRD field in the HL1CTRL register.

(3) Device state extension (DVSQ[3]) bit

This bit is used as the fourth bit for the device state (DVSQ) bits.
4’b 0000: Powered state
4’b 0001: Default state
4’b 0010: Address state
4’b 0011: Configured state
4’b 01xx: Suspended state
4’b 10xx: L1 state

(4) Device status (DVSQ[2:0]) bits

These bits mirror the DVSQ[2:0] bits in the Interrupt Status Register (INTSTS0).

(5) L1 negotiation mode (L1NEGOMD) bit

This bit is used to specify the negotiation function using the HIRD value.
0: Returns an ACK response when the received HIRD value is larger than the value in HIRDTHR[3:0] bits, or returns an NYET response in other cases.
1: Returns an ACK response when the received HIRD value is smaller than the value in HIRDTHR[3:0] bits, or returns an NYET response in other cases.

This bit is valid only when the value of L1RESPMD[1:0] bits is 2'b11.

(6) L1 response mode (L1RSPMD[1:0]) bits

When the L1RSPED bit is set, this module respond to an LPM token according to the value of these bits. These bits specify how to respond to the LPM token.
2’b00: NYET
2’b01: ACK
2’b10: STALL
2’b11: Response according to the value of L1NEGOMD bit

(7) L1 response enable (L1RSPEN) bit

When this bit is 0, this module does not respond to the LPM token it receives. When this bit is 1, this module responds to the LPM token (it receives) according to the value of the LPMRESPMD[1:0] bits.

33.2.23 Peripheral L1 Control Register 2

33.2.23.1 Peripheral L1 Control Register 2 [PL1CTRL2] <Address: 146H>

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	RWE MON	HIRDMON[3:0]				—	—	—	—	—	—	—	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
15 to 13	—	Nothing is assigned to these bits. Fix these bits to 0.	
12	RWEMON	This bit reflects the value of the RWE bit in the LPM token received last.	R/W
11 to 8	HIRDMON [3:0]	These bits reflect the value of the HIRD field in the LPM token received last.	R/W
7 to 0	—	Nothing is assigned to these bits. Fix these bits to 0.	

(1) RWE value monitor (RWEMON) bit

This bit is referenced to monitor the value of the RWE bit in a received LPM token.
This bit reflects the value of the RWE bit in the LPM token received last.

(2) HIRD value monitor (HIRDMON) bits

These bits are referenced to monitor the value of the HIRD field in a received LPM token.
These bits reflect the value of the HIRD field in the LPM token received last.

33.3 Next Register Set

33.3.1 Next Source Address Register n

33.3.1.1 Next0 Source Address Register ch0 [N0SA_0] <Address: 400H>
Next1 Source Address Register ch0 [N1SA_0] <Address: 40CH>
Next0 Source Address Register ch1 [N0SA_1] <Address: 440H>
Next1 Source Address Register ch1 [N1SA_1] <Address: 44CH>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SA (in normal mode), WD (in write-only mode)															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SA (in normal mode), WD (in write-only mode)															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	SA (in normal mode)	Source Address These bits specify the start address of the DMA transfer source.	R/W
	WD (in write-only mode)	Write Data These bits specify the write data in write-only mode.	R/W

Note: In a transfer in link mode, the data in the N0SA_n register is overwritten with descriptor read data.

33.3.2 Next Destination Address Register n

33.3.2.1 Next0 Destination Address Register ch0 [N0DA_0] <Address: 404H>
Next1 Destination Address Register ch0 [N1DA_0] <Address: 410H>
Next0 Destination Address Register ch1 [N0DA_1] <Address: 444H>
Next1 Destination Address Register ch1 [N1DA_1] <Address: 450H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DA															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DA															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	DA	Destination Address These bits specify the start address of the DMA transfer destination.	R/W

Note: In a transfer in link mode, the data in the N0DA_n register is overwritten with descriptor read data.

33.3.3 Next Transaction Byte Register n

33.3.3.1 Next0 Transaction Byte Register ch0 [N0TB_0] <Address: 408H>
Next1 Transaction Byte Register ch0 [N1TB_0] <Address: 414H>
Next0 Transaction Byte Register ch1 [N0TB_1] <Address: 448H>
Next1 Transaction Byte Register ch1 [N1TB_1] <Address: 454H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TB															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TB															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	TB	Transaction Byte These bits specify the total number of transfer bytes. (Note: Do not start a DMA transaction with 0 set in these bits.)	R/W

Note: The N0TB_n register is overwritten by the descriptor read data during link mode transfer.

33.4 Current Register Set

33.4.1 Current Source Address Register

33.4.1.1 Current Source Address Register ch0 [CRSA_0] <Address: 418H>
Current Source Address Register ch1 [CRSA_1] <Address: 458H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CRSA															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRSA															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	CRSA	Current Source Address Register This register indicates the read address for the next DMA transaction. The value of this register is incremented automatically while DMA transactions are in process. (When the SAD bit in the CHCFG_n register is 1, the value of this register is fixed. When the WONLY bit in the CHCFG_n register is 1, the value of this register is undefined.) The initial value of this register is loaded from the following register: In register mode: A transfer source address is loaded from the Next0/1 Register Set. In link mode: A transfer source address is loaded from the descriptor. (The descriptor read data is input to the N0SA_n register, and is loaded into the CRSA_n register when a transfer starts.) The value of this register is incremented when a read transfer ends. Read this register after DMA stops (that is, after the TACT bit in the CHSTAT_n register is cleared). (Handle the value read during DMA only as a reference value.)	R

33.4.2 Current Destination Address Register

33.4.2.1 Current Destination Address Register ch0 [CRDA_0] <Address: 41CH>
Current Destination Address Register ch1 [CRDA_1] <Address: 45CH>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CRDA															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRDA															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	CRDA	Current Destination Address Register This register indicates the write address for the next DMA transaction. The value of this register is incremented automatically while DMA transactions are in process. (When the DAD bit in the CHCFG_n register is 1, the value of this register is fixed.) The initial value of this register is loaded from the following register: In register mode: A transfer destination address is loaded from the Next0/1 Register Set. In link mode: A transfer destination address is loaded from the descriptor. (The descriptor read data is input to the N0DA_n register, and is loaded into the CRDA_n register when a transfer starts.) The value of this register is incremented when a write transfer ends. Read this register after DMA stops (that is, after the TACT bit in the CHSTAT_n register is cleared). (Handle the value read during DMA only as a reference value.)	R

33.4.3 Current Transaction Byte Register

33.4.3.1 Current Transaction Byte Register ch0 [CRTB_0] <Address: 420H>
Current Transaction Byte Register ch1 [CRTB_1] <Address: 460H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CRTB															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRTB															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	CRTB	Current Transaction Byte Register This register indicates the remaining number of transfer bytes in the ongoing DMA transaction. The value of this register is decremented automatically while the DMA transaction is in process. The initial value of this register is loaded from the following register: In register mode: The number of transfer bytes is loaded from the Next0/1 Register Set. In link mode: The number of transfer bytes is loaded from the descriptor. (The descriptor read data is input to the N0TB_n register, and is loaded into the CRTB_n register when a transfer starts.) The value of this register is decremented when a write transfer ends. Read this register after DMA stops (that is, after the TACT bit in the CHSTAT_n register is cleared). (Handle the value read during DMA only as a reference value.)	R

33.5 Channel Register Set

33.5.1 Channel Status Register n

33.5.1.1 Channel Status Register ch0 [CHSTAT_0] <Address: 424H>
Channel Status Register ch1 [CHSTAT_1] <Address: 464H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DNUM								—					SWPRQ	DMARQM	INTM
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—				MODE	DER	DW	DL	SR	TC	END	ER	SUS	TACT	RQST	EN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 24	DNUM	Data Number These bits indicate the amount of valid data in the buffer. The indicated amount of data is the amount of the data that was read from the source but has not yet been written to the destination. (Unit: byte) Incrementing condition: - A DMA read transfer ends. Decrementing condition: - A DMA write transfer ends. Clearing conditions: - A condition for clearing the EN bit is met. - "1" is written to the SWRST bit in the CHCTRL_n register.	R
23 to 19	—	Reserved area. Write 0 to these bits. When these bits are read, 0 is read.	
18	SWPRQ	Sweep Request This bit indicates the state of sweep request. This bit indicates the state of software sweep request (which has been activated by the SETSSWPRQ bit in the CHCTRL_n register). 1: The sweep request signal has been asserted. 0: The sweep request signal has not been asserted. Setting condition: - The SETSSWPRQ bit in the CHCTRL_n register is asserted. Clearing conditions: - The buffer becomes empty because of sweep. - "1" is written to the CLRHSWPRQM bit in the CHCTRL_n register. - "1" is written to the SWRST bit in the CHCTRL_n register.	R
17	DMARQM	DMAREQ Mask This bit indicates the state of temporary masking of the DMA transfer request from the USB control. 1: The request is temporarily masked. 0: The request is released from temporary masking. Setting condition: - The SETDMARQM bit in the CHCTRL_n register is set. Clearing conditions: - "1" is written to the CLRDMARQM bit in the CHCTRL_n register. - "1" is written to the SWRST bit in the CHCTRL_n register.	R

Bit	Name	Function	R/W
16	INTM	Interrupt Mask This bit indicates the state of temporary masking of the output from the USBFDMAmn interrupt. 1: The output is temporarily masked. 0: The output is released from temporary masking. Setting condition: - "1" is written to the SETINTM bit in the CHCTRL_n register. Clearing conditions: - "1" is written to the CLRINTM bit in the CHCTRL_n register. - "1" is written to the SWRST bit in the CHCTRL_n register.	R
15 to 12	—	Reserved area. Write 0 to these bits. When these bits are read, 0 is read.	
11	MODE	DMA Mode This bit indicates the DMA mode. The indicated value is the value of the DMS bit in the CHCFG_n register. 0: Register mode 1: Link mode	R
10	DER	Descriptor Error This bit indicates whether the data read from the descriptor is invalid (LV = 0) (regardless of the value of the DIM bit in the CHCFG_n register). 0: No descriptor error has occurred. 1: A descriptor error has occurred. Setting condition: - In link mode, when the DRRP bit in the CHCFG_n register is 0, the LV bit value read from the descriptor is 0. Clearing conditions: - "1" is written to the CLRDER bit in the CHCTRL_n register. - "1" is written to the SWRST bit in the CHCTRL_n register.	R
9	DW	Descriptor WriteBack This bit indicates whether data is being written back to the descriptor. If a bus error occurs during write-back to the descriptor, this bit retains 1. 0: Status other than write-back to the header in link mode 1: (When the ER bit in the CHSTAT_n register is 0) Data is being written back to the header in link mode. (When the ER bit in the CHSTAT_n register is 1) A bus error has occurred during write-back to the header in link mode. Setting condition: - Write-back to the header is started in link mode. Clearing conditions: - Write-back to the header in link mode ends with an OK response. - "1" is written to the SWRST bit in the CHCTRL_n register. If the bit retains 1 because of an error response, this bit can be cleared only by setting the SWRST bit.	R
8	DL	Descriptor Load This bit indicates whether data is being read from the descriptor. If a bus error occurs during descriptor reading, this bit retains 1. 0: Status other than descriptor reading 1: (When the ER bit is 0) Descriptor reading is in process in link mode. (When the ER bit is 1) A bus error has occurred during descriptor reading in link mode. Setting condition: - Descriptor reading is started in link mode. Clearing conditions: - Descriptor reading in link mode ends with an OK response. - "1" is written to the SWRST bit in the CHCTRL_n register. If the bit retains 1 because of an error response, this bit can be cleared only by setting the SWRST bit.	R

Bit	Name	Function	R/W
7	SR	Selected Register Set In register mode, this bit indicates the register set that is selected. 0: Next0 Register Set 1: Next1 Register Set Setting condition: - The RSEL bit in the CHCFG_n register is set. Clearing condition: - The RSEL bit in the CHCFG_n register is cleared.	R
6	TC	Terminal Count This status bit indicates whether the DMA transaction has ended. This bit is set only when the TCM bit in the CHCFG_n register is 0. 0: The DMA transfer has not ended. 1: The DMA transfer has ended. Setting conditions: - In register mode, transfer of the total number of transfer bytes specified in the CRTB bits ends. - In link mode, when the WBD bit in the header of the descriptor is 1, transfer of the total number of transfer bytes specified in the CRTB bits ends. - In link mode, when the WBD bit in the header of the descriptor is 0, write-back to the descriptor ends. Clearing conditions: - "1" is written to the CLRTC bit in the CHCTRL_n register. - "1" is written to the SWRST bit in the CHCTRL_n register.	R
5	END	USBFDMAmn Interrupted This bit indicates whether the DMA transaction has ended and an USBFDMAmn interrupt has occurred. 0: The DMA transfer has not ended. 1: The DMA transfer has ended. Setting conditions: - The condition for setting the TC bit is met, and the DEM bit in the CHCFG_n register is 0. - In link mode, when the descriptor is read, the LV bit in the header is 0, and the DRRP and DIM bits in the CHCFG_n register are 0. Clearing conditions: - "1" is written to the CLREND bit in the CHCTRL_n register. - "1" is written to the SWRST bit in the CHCTRL_n register.	R
4	ER	Error This bit indicates whether an error response has been received and a DMAERR interrupt has occurred during the DMA transfer. 0: No error response has been received. 1: An error response has been received. Setting condition: - An error response is received in a bus cycle. Clearing condition: - "1" is written to the SWRST bit in the CHCTRL_n register.	R
3	SUS	Suspend This bit indicates whether the channel is suspended. For details, see section 33.9.13.8, (2), Suspension 0: Channel_n is not suspended. 1: Channel_n is suspended. Setting condition: - "1" is written to the SETSUS bit in the CHCTRL_n register during the DMA transfer using channel_n, and, thereby, the inside of the channel is suspended. Clearing conditions: - "1" is written to the CLRSUS bit in the CHCTRL_n register. - "1" is written to the CLREN bit in the CHCTRL_n register. - A condition for clearing the EN bit in the CHSTAT_n register.	R

Bit	Name	Function	R/W
2	TACT	Transaction Active This bit indicates whether the DMAC is operating. This bit is used to check whether the channel is stopped fully. For details, see section 33.9.13.8, Transfer state. 0: The DMA in channel_n is stopped. 1: The DMA in channel_n is operating. Setting condition: - "1" is written to the SETEN bit in the CHCTRL_n register (to start descriptor reading or wait for a DMA request). Clearing condition: - The internal state is the idle state (the EN bit in the register has been cleared, and all transfers have ended).	R
1	RQST	Request This bit indicates whether a transfer request has been received. 0: No DMA transfer request has been received. 1: A DMA transfer request has been received. Setting conditions: - "1" is written to the STG bit in the CHCTRL_n register. - A DMA transfer request is received from the USB control. Clearing conditions: - "1" is written to the SWRST bit in the CHCTRL_n register. - "1" is written to the CLRRQ bit in the CHCTRL_n register. - In single transfer mode (the TM bit in the CHCFG_n register is 0), a transfer is executed on the side specified by the REQD bit in the CHCFG_n register - In register mode, all DMA transactions are complete (the REN bit in the CHCFG_n register is 0). - In link mode, the DMA transfer of the last descriptor (LE = 1) ends. - In link mode, a DMA transfer is stopped during descriptor reading (the LV bit is 0 and the DRRP bit in the CHCFG_n register is 0). - In link mode, when the DEM bit in the CHCFG_n register is 0, a DMA transaction ends. - The master interface receives a bus error signal.	R
0	EN	Enable This bit indicates whether the operation of DMA channel n is enabled or stopped. 0: Operation is stopped. 1: Operation is enabled. Setting conditions: - "1" is written to the SETEN bit in the CHCTRL_n register. Clearing conditions: - "1" is written to the SWRST bit in the CHCTRL_n register. - "1" is written to the CLREN bit in the CHCTRL_n register. - An error response is received during transfer. - In register mode, all DMA transactions are completed (the REN bit in the CHCFG_n register is 0). - In link mode, DMA transfer of the last descriptor (the LE bit is 1) ends (if the WBD bit is 0, write-back to the descriptor ends). - In link mode, reading of a descriptor is stopped (the LV bit is 0 and the DRRP bit in the CHCFG_n register is 0).	R

Note 1. When the ER bit in the CHSTAT_n register is set, treat the corresponding series of transfers as invalid transactions.
Note 2. To interrupt a DMA transaction, mask or clear transfer requests or clear the EN bit in the CHSTAT_n register. (For the procedure to interrupt, see section 33.9.13.8, (3), Transfer suspension.)
Note 3. If the DMA transfer request from the USB control and the transfer request by software (setting the STG bit in the CHCFG_n register) are used together, the cause of activating the request that takes effect cannot be identified. Therefore, design the system so that only one type of transfer requests is used.
Note 4. When using the transfer request by software, operate the STG bit for a new transfer request only after the DMA transfer requested last ends (after checking the end of the last DMA transfer by referencing the Current Register Set or another method).

33.5.2 Channel Control Register n

33.5.2.1 Channel Control Register ch0 [CHCTRL_0] <Address: 428H>
Channel Control Register ch1 [CHCTRL_1] <Address: 468H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
—												CLR DMA RQM	SET DMA RQM	CLR INTM	SET INTM
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	SET SSWP RQ	—	SETR EN	—		CLR SUS	SET SUS	CLR DER	CLR TC	CLR END	CLR RQ	SW RST	STG	CLR EN	SET EN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 20	—	Reserved area. Write 0 to these bits. When these bits are read, 0 is read	
19	CLRDMARQM	Clear DMAREQ Mask Writing 1 to this bit releases the DMA transfer requests from the USB control from temporary masking. Writing 1 to this bit also clears the DMARQM bit in the CHSTATn register. When this bit is read, 0 is read. 1: Releases the DNA transfer requests from masking set by using the SETDMARQM bit. 0: Has no effect on operation.	R/W
18	SETDMARQM	SET DMAREQ Mask Writing 1 to this bit temporarily masks the DMA transfer requests from the USB control. Writing 1 to this bit also sets the DMARQM bit in the CHSTATn register. When this bit is read, 0 is read. 1: Masks the DMA transfer requests from the USB control. 0: Has no effect on operation.	R/W
17	CLRINTM	Clear Interrupt Mask Writing 1 to this bit releases the USBFDMAmn interrupt from masking. Writing 1 to this bit also clears the INTM bit in the CHSTATn register. Releasing the INT_DMA[n] pin output from masking when the LVINT bit in the DCTRL register and the END bit in the CHSTAT_n register are 1 activates the INT_DMA[n] pin output. (The pin output is not activated if the LVINT bit is 0.) When this bit is read, 0 is read. 1: Releases the pin output from masking set by using the SETINTM bit. 0: Has no effect on operation.	R/W
16	SETINTM	SETINTMSet Interrupt Mask Writing 1 to this bit temporarily masks the USBFDMAmn interrupt. Writing 1 to this bit also sets the INTM bit in the CHSTATn register. When this bit is read, 0 is read. 1: Masks the USBFDMAmn interrupt. 0: Has no effect on operation.	R/W
15	—	Reserved area. Write 0 to this bit. When this bit are read, 0 is read.	
14	SETSSWPRQ	Set Software Sweep Request Writing 1 to this bit sweeps out the data stored in the buffer to the destination (see section 33.9.13.3, (1), Forced software sweeping request). When this bit is read, 0 is read. 1: Writes, to the destination, the data that is stored in the buffer and has not yet been written to the destination. 0: Has no effect on operation. If the destination asserts a hardware request (REQD = 1), the sweep operation cannot be used.	R/W
13	—	Reserved area. Write 0 to this bit. When this bit are read, 0 is read.	

Bit	Name	Function	R/W
12	SETREN	Set Register Set Enable Writing 1 to this bit sets the REN bit in the CHCFG_n register. When this bit is read, 0 is read. 1: Sets the REN bit in the CHCFG_n register. 0: Has no effect on operation.	R/W
11, 10	—	Reserved area. Write 0 to these bits. When these bits are read, 0 is read.	
9	CLRSUS	Clear Suspend Writing 1 to this bit when the SUS bit in the CHSTAT_n register is 1 releases the ongoing DMA transfer from the suspended state. When this bit is read, 0 is read. 1: Releases the ongoing DMA transfer from the suspended state. 0: Has no effect on operation.	R/W
8	SETSUS	Set Suspend Writing 1 to this bit when the EN bit in the CHSTAT_n register is 1 suspends the ongoing DMA transfer. When this bit is read, 0 is read. 1: Suspends the ongoing DMA transfer. 0: Has no effect on operation.	R/W
7	CLRDER	Clear DER Writing 1 to this bit clears the DER bit in the CHSTAT_n register. Writing 1 to this bit also clear the USBFDMAmn interrupt. When this bit is read, 0 is read. 1: Clears the DER bit. 0: Has no effect on operation.	R/W
6	CLRTC	Clear TC Writing 1 to this bit clears the TC bit in the CHSTAT_n register. When this bit is read, 0 is read. 1: Clears the TC bit. 0: Has no effect on operation.	R/W
5	CLREND	Clear End Writing 1 to this bit clears the END bit in the CHSTAT_n register. Writing 1 to this bit also clear the USBFDMAmn interrupt. When this bit is read, 0 is read. 1: Clears the END bit. 0: Has no effect on operation.	R/W
4	CLRRQ	Clear Request Writing 1 to this bit clears the RQST bit in the CHSTAT_n register. When this bit is read, 0 is read. 1: Clears the RQST bit in the CHSTAT_n register. 0: Has no effect on operation.	R/W
3	SWRST	Software Reset Writing 1 to this bit clears individual bits in the CHSTAT_n register (for the bits to be cleared, see the description of each bit). Setting this bit must be done when the EN and TACT bits are 0. When this bit is read, 0 is read. 1: Clears individual bits in the CHSTAT_n register. 0: Has no effect on operation.	R/W
2	STG	Software Trigger Writing 1 to this bit makes software set an internal transfer request. If this bit and the SWRST bits are set at the same time, clearing by the SWRST bit takes priority. When this bit is read, 0 is read. 1: Makes software set a transfer request (set the RQST bit in the CHSTAT_n register). 0: Has no effect on operation.	R/W
1	CLREN	Clear Enable Writing 1 to this bit clears the EN bit in the CHSTAT_n register (for details, see 4.13.8.3). When this bit is read, 0 is read. 1: Disables DMA transfers (clears the EN bit in the CHSTAT_n register). 0: Has no effect on operation.	R/W
0	SETEN	Set Enable Writing 1 to this bit enables DMA transfers in DMA channel n. If this bit and the SWRST bits are set at the same time, clearing by the SWRST bit takes priority, and DMA transfers do not start. When this bit is read, 0 is read. 1: Enables DMA transfers (sets the EN bit in the CHSTAT_n register). 0: Has no effect on operation.	R/W

Note: Temporary masking (using the CLRDMARQM, and SETDMARQM bits) of the DMA transfer requests from the USB control applies to only the resources for channel n. Setting the SETDMARQM bit for channel n does not affect the operation of channel m.

33.5.3 Channel Configuration Register

33.5.3.1 Channel Configuration Register ch0 [CHCFG_0] <Address: 42CH> Channel Configuration Register ch1 [CHCFG_1] <Address: 46CH>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMS	REN	RSW	RSEL	SBE	DIM	TCM	DEM	WONLY	—	DAD	SAD	DDS			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDS				DRRP	—			—	—			REQD	—		SEL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31	DMS	DMA Mode Select This bit specifies the DMA mode to be used. 0: Register mode (default) 1: Link mode	R/W
30	REN	Register Set Enable This bit specifies whether to successively perform, after a DMA transaction ends, another DMA transaction using the Next Register Set selected by the RSEL bit. This bit is valid only in register mode. 0: Does not perform the DMA transaction successively. 1: Performs the DMA transaction successively. Setting conditions: - "1" is written to this bit. - "1" is written to the SETREN bit in the CHCTRL_n register. Clearing conditions: - "0" is written to this bit. - The REN bit is 1, and a DMA transaction ends. To re-set the REN bit during a transaction, we recommend you to use the SETREN bit in the CHCTRL_n register.	R/W
29	RSW	Register Select Switch This bit specifies whether to automatically invert the RSEL bit after a DMA transaction ends. This bit is valid only in register mode. 0: Does not invert the RSEL bit after a DMA transaction ends. (Default) 1: Inverts the RSEL bit after a DMA transaction ends.	R/W
28	RSEL	Register Set Select This bit is used to select the Next Register Set to be used for the next DMA transaction. This bit is valid only in register mode. When the RSW bit is 1, this bit is inverted automatically at the end of a DMA transaction. 0: Uses the Next0 Register Set. (Default) 1: Uses the Next1 Register Set. Transition condition: A DMA transaction ends with the RSW bit set.	R/W
27	SBE	Sweep Buffer Enable This bit specifies whether to sweep (write) the data already read and stored in the buffer and stop transfer when the EN bit in the CHSTAT_n register is cleared during a DMA transaction. The sweep mode can be used only when the REQD bit is 0. 0: Stops transfer without sweeping out the buffer. (Default) 1: Stops transfer after sweeping out the buffer.	R/W
26	DIM	Descriptor Interrupt Mask This bit specifies whether to mask the USBFDMAmn interrupt when the LV bit value read from descriptor header is 0. 0: Does not mask the USBFDMAmn interrupt. (Default) 1: Mask the USBFDMAmn interrupt.	R/W

Bit	Name	Function	R/W
25	TCM	DMATC Mask This bit is used to mask the DMATC signal, which is sent from the DMAC to USB control. When this bit is 1 at the time the DMATC signal is to be output, the DMATC signal is not asserted. Also, the TC bit in the CHSTAT_n register is not asserted in that case. In register mode, this bit is automatically cleared. In link mode, this bit is not cleared automatically. Use this bit when you control DMA transfers by software. 0: Does not mask the DMATC signal. (Default) 1: Masks the DMATC signal. Clearing condition: A DMA transaction ends with the TCM bit set.	R/W
24	DEM	USBFDMAmn Mask When this bit is 1 at the time the USBFDMAmn interrupt is not asserted. Also, the END bit in the CHSTAT_n register is not asserted in that case. In register mode, this bit is not cleared automatically. In link mode, this bit is automatically cleared. 0: Does not mask the USBFDMAmn interrupt. (Default) 1: Masks the USBFDMAmn interrupt. Clearing condition: A DMA transaction ends with the DEM bit set.	R/W
23	WONLY	Write Only Mode This bit is used to switch the transfer operation mode to the write-only mode (see section 33.9.12.2, Write only mode). 0: Normal operation (default) 1: Write-only mode	R/W
22	—	Nothing is assigned to this bit. Fix this bit to 0.	
21	DAD	Destination Address Direction This bit specifies the direction of counting the transfer-destination address in DMA channel n. If the transfer destination is on the USB control side, write 1 (fixed) to this bit. 0: Incrementing (default) 1: Fixed When the transfer destination uses the skip mode or is beat-unaligned, do not specify 1 (fixed) in this bit.	R/W
20	SAD	Source Address Direction This bit specifies the direction of counting the transfer-source address in DMA channel n. If the transfer source is on the USB control side, write 1 (fixed) to this bit. 0: Incrementing (default) 1: Fixed When the transfer source uses the skip mode or is beat-unaligned, do not specify 1 (fixed) in this bit.	R/W
19 to 16	DDS[3:0]	Destination Data Size These bits specify the size of DMA transfer data. When the transfer destination is on the USB control side, select the normal mode. Use bit 3 to switch between the normal and skip modes. 0: Normal mode (default) 1: Skip mode Use bits 2 to 0 to specify the size of transfer data. (For specifiable values, see Table 33.21.) 000: 8 bits (default) 001: 16 bits 010: 32 bits 011: 64 bits 100: 128 bits 101: 256 bits 110: 512 bits 111: Setting prohibited	R/W

Bit	Name	Function	R/W
15 to 12	SDS[3:0]	Source Data Size These bits specify the size of DMA transfer data. Use bit 3 to switch between the normal and skip modes. 0: Normal mode (default) 1: Skip mode Use bits 2 to 0 to specify the size of transfer data. (For specifiable values, see Table 33.29.) 000: 8 bits (default) 001: 16 bits 010: 32 bits 011: 64 bits 100: 128 bits 101: 256 bits 110: 512 bits 111: Setting prohibited	R/W
11	DRRP	Descriptor Read Repeat This bit switches the operation to be performed when the LV value in the header read from the descriptor is 0. (See section 33.9.12.2, (a), Operation flow of link mode.) 0: This module sets the DER bit in the CHSTAT_n register, and then stops descriptor reading. (Default) 1: This module keeps reading the same descriptor until the LV value changes to 1, and, when the LV value becomes 1, starts the DMA transfer using the values in the descriptor. The interval of descriptor reading is controlled by using the DSCITVL register.	R/W
10 to 7	—	Nothing is assigned to these bits. Fix these bits to 0.	
6, 5	—	Nothing is assigned to these bits. Fix these bits to 1.	
4	—	Nothing is assigned to this bit. Fix this bit to 0.	
3	REQD	Request Direction This bit specifies whether the USB control is on the transfer source side or it is on the transfer destination side. 0: The USB control is on the transfer source side. (Default) 1: The USB control is on the transfer destination side.	R/W
2, 1	—	Nothing is assigned to these bits. Fix these bits to 0.	
0	SEL	Terminal Select This bit selects the FIFO channel to be used on the USB control side. 0: D0FIFO 1: D1FIFO	R/W

The range of values specifiable in the SDS[2:0] and DDS[2:0] bits depends on the data bus width, number of implemented buffer stages, and whether the address to be accessed is beat-aligned or not (beat-unaligned). The table below shows the range of specifiable values.

Table 33.21 Range of Sizes That Can Be Specified in SDS and DDS Bits

Transfer Address	REQD	SDS[2:0]	DDS[2:0]
Beat aligned	0	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.	8 to 32 bits (000 to 010) 128 to 512 bits (100 to 110)
	1	8 to 32 bits (000 to 010) 128 to 512 bits (100 to 110)	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.
Beat unaligned	0	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.	8 to 32 bits (000 to 010) 128 to 256 bits (100 to 101)
	1	8 to 32 bits (000 to 010) 128 to 256 bits (100 to 101)	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.

Note: When the destination is beat-unaligned with REQD = 0 or the source is beat-unaligned with REQD = 1, specify values in the range of specifiable values for beat-unaligned transfer addresses in both of the SDS[2:0] and DDS[2:0] bits.
Even if the transfer source and destination are beat-aligned when a DMA transaction starts, they might become beat-unaligned in the middle of the transaction when a skip transfer is used. If this might occur, perform settings in the first place on the assumption that the transfer source and destination are beat-unaligned.
If software cannot determine whether the transfer source and/or destination is beat-aligned, use values in the range of specifiable values for beat-unaligned transfer addresses.

33.5.4 Channel Interval Register n

For details, see section 33.9.13.6, Interval Count Function.

33.5.4.1 Channel Interval Register ch0 [CHITVL_0] <Address: 430H>
Channel Interval Register ch1 [CHITVL_1] <Address: 470H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
—															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ITVL															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 16	—	Nothing is assigned to these bits. Fix these bits to 0.	
15 to 0	ITVL	Interval These bits specify the DMA transfer interval.	R/W

33.5.5 Channel Extension Register n

33.5.5.1 Channel Extension Register ch0 [CHEXT_0] <Address: 434H>
Channel Extension Register ch1 [CHEXT_1] <Address: 474H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
—												—			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—				DPR				—				SPR			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 12	—	Nothing is assigned to these bits. Fix these bits to 0.	
11 to 8	DPR	Destination PROT These bits specify the value to be output to the MHPROT[3:0] pin in a DMA write transfer. The initial value of these bits is 0H.	R/W
7 to 4	—	Nothing is assigned to these bits. Fix these bits to 0.	R/W
3 to 0	SPR	Source PROT These bits specify the value to be output to the MHPROT[3:0] pin in a DMA read transfer. The initial value of these bits is 0H.	R/W

33.6 Link Register Set

When software sets a descriptor address in the NXLA_n register and starts the DMAC, hardware loads the value set in the NXLA_n register into the CRLA_n register. Then, the descriptor is read, and the DMAC starts a DMA transaction according to the values read from the descriptor. The value in the NXLA_n register is automatically updated to the Next Link Address value read from the descriptor, and the updated value is used as the descriptor address in the next DMA transaction.

33.6.1 Next Link Address Register n (NXLA_n)

33.6.1.1 Next Link Address Register ch0 [NXLA_0] <Address: 438H>
Next Link Address Register ch1 [NXLA_1] <Address: 478H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NXLA															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NXLA															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	NXLA	Next Link Address These bits specify the link-destination address. Upper two bits are fixed to 0, and only the address aligned with a word boundary can be set.	Bit 31 to 2: R/W Bit 1, 0: R

33.6.2 Current Link Address Register n (CRLA_n)

33.6.2.1 Current Link Address Register ch0 [CRLA_0] <Address: 43CH>
Current Link Address Register ch1 [CRLA_1] <Address: 47CH>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CRLA															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRLA															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	CRLA	Current Link Address These bits indicate the address of the descriptor being used for current transaction.	R

33.7 Skip Register Set

This register set is used to specify settings for a skip (scatter/gather) transfer.

33.7.1 Source Continuous Register n

33.7.1.1 Source Continuous Register ch0 [SCNT_0] <Address: 600H>
Source Continuous Register ch1 [SCNT_1] <Address: 620H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SCNT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCNT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	SCNT	Source Continuous These bits specify the size of the space to be accessed continuously by source address access. (Unit: byte)	R/W

Note: This register is used in pair with the SSKP_n register (see Figure 33.3). To use this mode, set the SDS[3] bits in the CHCFG_n register to 1. To perform a skip transfer on the transfer source side, the SAD bit in the CHCFG_n register must not be set to 1 (Fixed). Do not perform a skip transfer with the SCNT bits set to 0.

33.7.2 Source Skip Register n

33.7.2.1 Source Skip Register ch0 [SSKP_0] <Address: 604H>
Source Skip Register ch1 [SSKP_1] <Address: 624H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SSKP															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSKP															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	SSKP	Source Skip These bits specify the size of area to be skipped in a source address access. (Unit: byte)	R/W

Note: This register is used in pair with the SCNT_n register (see Figure 33.3). To use this mode, set the SDS[3] bits in the CHCFG_n register to 1. To perform a skip transfer on the transfer source side, the SAD bit in the CHCFG_n register must not set to 1 (Fixed).

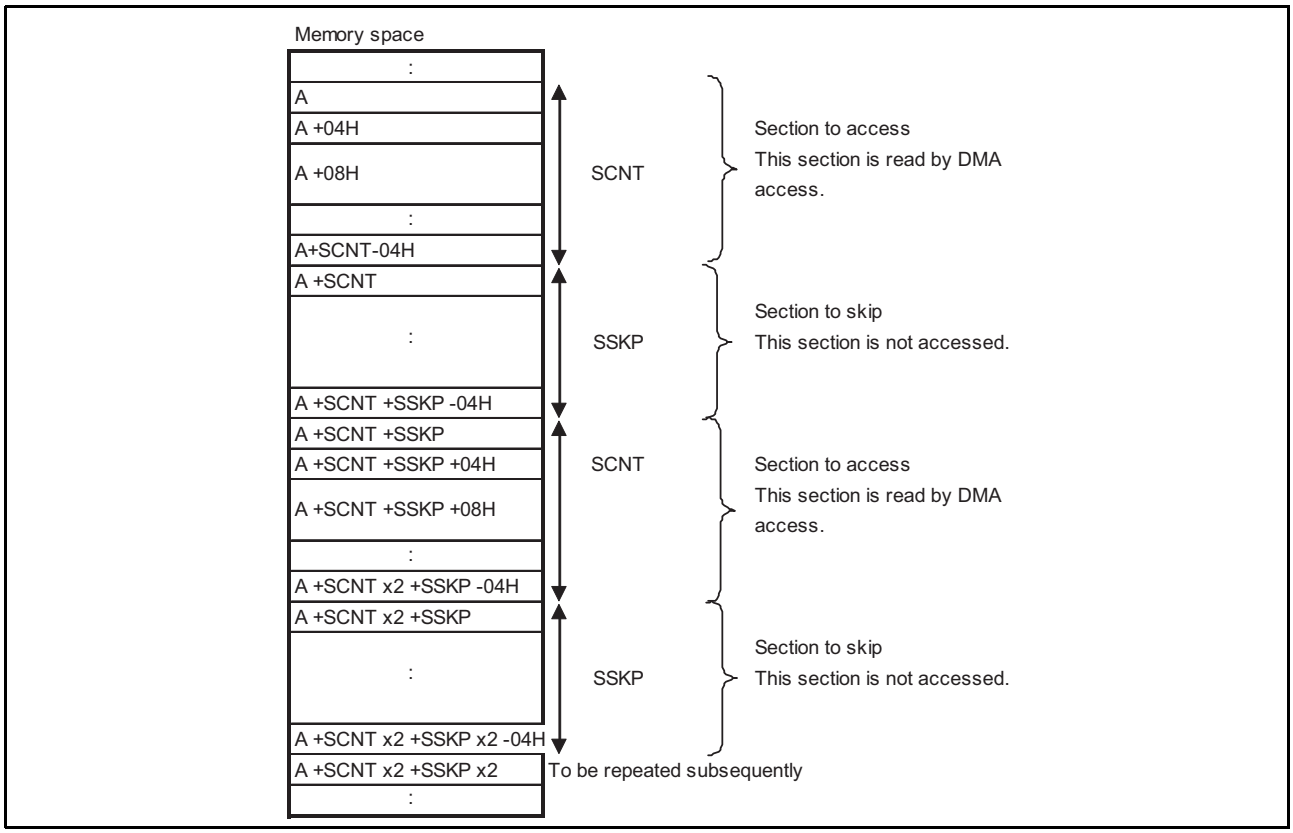


Figure 33.3 Relationship between SSKP and SCNT

You can specify values of the SCNT and SSKP bits regardless of the source address and the value of the SDS field in the CHCFG_n register. The DMAC performs access based on the size specified in the SDS field, and fetches only valid data into the buffer (see section 33.9.14.1, (1), Read access).

33.7.3 Destination Continuous Register n

33.7.3.1 Destination Continuous Register ch0 [DCNT_0] <Address: 608H>
Destination Continuous Register ch1 [DCNT_1] <Address: 628H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DCNT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCNT															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	DCNT	Destination Continuous These bits specify the size of the space to be accessed continuously by destination address access. (Unit: byte)	R/W

Note: This register is used in pair with the DSKP_n register (see Figure 33.4). To use this mode, set the DDS[3] bits in the CHCFG_n register to 1. To perform a skip transfer on the transfer destination side, the DAD bit in the CHCFG_n register must not be set to 1 (Fixed). Do not perform a skip transfer with the DCNT bits set to 0.

33.7.4 Destination Skip Register n

33.7.4.1 Destination Skip Register ch0 [DSKP_0] <Address: 60CH>
Destination Skip Register ch1 [DSKP_1] <Address: 62CH>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DSKP															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DSKP															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 0	DSKP	Destination Skip These bits specify the size of area to be skipped in a destination address access. (Unit: byte)	R/W

Note: This register is used in pair with the DCNT_n register (see Figure 33.4). To use this mode, set the DDS[3] bits in the CHCFG_n register to 1. To perform a skip transfer on the transfer destination side, the DAD bit in the CHCFG_n register must not be set to 1 (Fixed).

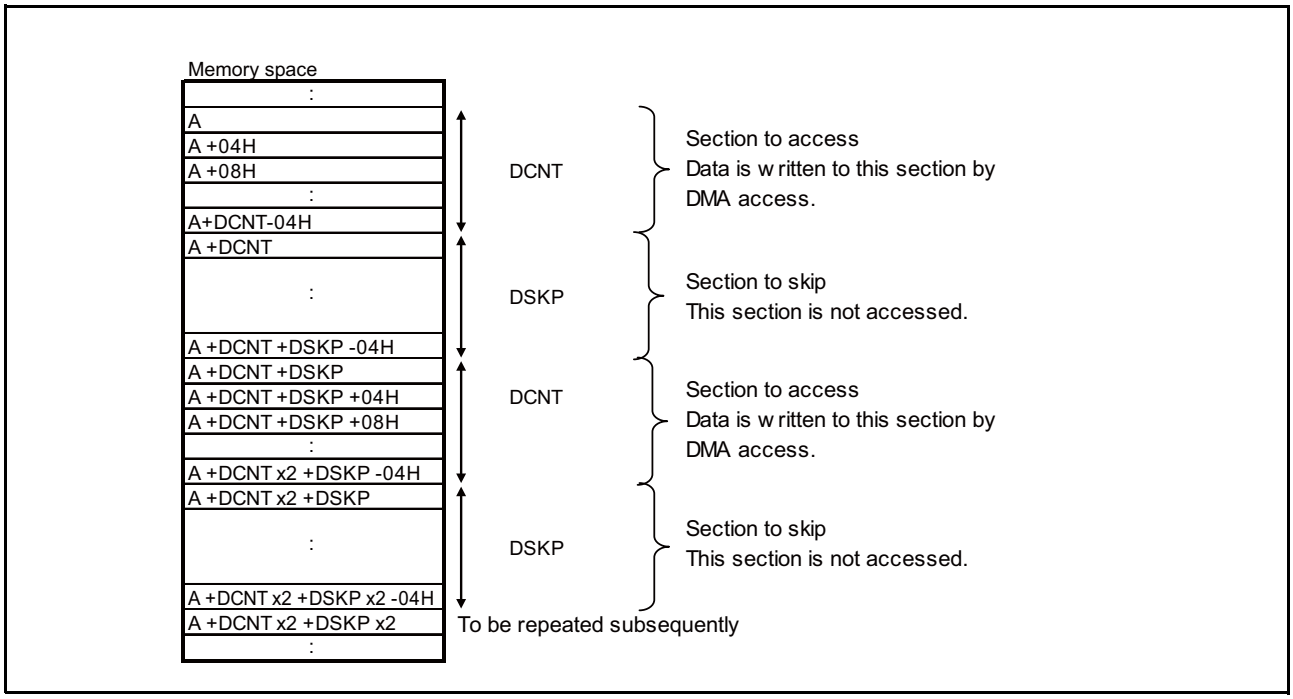


Figure 33.4 Relationship between DSKP and DCNT

You can specify values of the DCNT and DSKP bits regardless of the destination address and the value of the DDS field in the CHCFG_n register. The DMAC performs write access to only the specified space that has a combined size not more than the size specified in the DDS field (see section 33.9.14.1, (1), Read access).

33.8 DMA Register Set

The registers described below are shared by all channels.

33.8.1 DMA Control Register

33.8.1.1 DMA Control Register [DCTRL] <Address: 700H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
—				LWPR				—				LDPR			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—														LVINT	PR
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 28	—	Nothing is assigned to these bits. Fix these bits to 0.	
27 to 24	LWPR	Link WriteBack PROT These bits specify the value to be output to the MHPROT[3:0] pin at write-back to the descriptor in link mode.	R/W
23 to 20		Nothing is assigned to these bits. Fix these bits to 0.	
19 to 16	LDPR	Link Descriptor PROT These bits specify the value to be output to the MHPROT[3:0] pin at reading of the descriptor in link mode.	R/W
15 to 2	—	Nothing is assigned to these bits. Fix these bits to 0.	
1	LVINT	Level Interrupt To use this module, be sure to set this bit to 1.	R/W
0	PR	Priority This bit specifies the transfer priority control mode (see section 33.9.13.2, DMA channel priority control). 0: Fixed priority mode 1: Round-robin mode	R/W

33.8.2 Descriptor Interval Register n

33.8.2.1 Descriptor Interval Register [DSCITVL] <Address: 704H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
—															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DITVL								—							
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 16	—	Nothing is assigned to these bits. Fix these bits to 0.	
15 to 8	DITVL	Descriptor Interval These bits specify the interval of descriptor read operation. The descriptor will be re-read at intervals of "'value of DITVL x 256."	R/W
7 to 0	—	Nothing is assigned to these bits. Fix these bits to 0.	

33.8.3 DMA Control Register

33.8.3.1 DMA Status EN Register [DSTAT_EN] <Address: 710H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
—															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—														EN1	EN0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 2	—	Nothing is assigned to these bits. Fix these bits to 0.	
1	EN1	This bit indicates the state of the EN bit for DMA channel 1.	R
0	EN0	This bit indicates the state of the EN bit for DMA channel 0.	R

33.8.4 DMA Status ER Register

33.8.4.1 DMA Status ER Register [DSTAT_ER] <Address: 714H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
—															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—														ER1	ER0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 2	—	Nothing is assigned to these bits. Fix these bits to 0.	
1	ER1	This bit indicates the state of the ER bit for DMA channel 1.	R
0	ER0	This bit indicates the state of the ER bit for DMA channel 0.	R

33.8.5 DMA Status END Register

33.8.5.1 DMA Status END Register [DSTAT_END] <Address: 718H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
—															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—														END1	END0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 2	—	Nothing is assigned to these bits. Fix these bits to 0.	
1	END1	This bit indicates the state of the END bit for DMA channel 1.	R
0	END0	This bit indicates the state of the END bit for DMA channel 0.	R

33.8.6 DMA Status TC Register

33.8.6.1 DMA Status TC Register [DSTAT_TC] <Address: 71CH>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
—															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—														TC1	TC0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 2	—	Nothing is assigned to these bits. Fix these bits to 0.	
1	TC1	This bit indicates the state of the TC bit for DMA channel 1.	R
0	TC0	This bit indicates the state of the TC bit for DMA channel 0.	R

33.8.7 DMA Status SUS Register

33.8.7.1 DMA Status SUS Register [DSTAT_SUS] <Address: 720H>

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
—															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—														TC1	TC0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Name	Function	R/W
31 to 2	—	Nothing is assigned to these bits. Fix these bits to 0.	
1	SUS1	This bit indicates the state of the SUS bit for DMA channel 1.	R
0	SUS0	This bit indicates the state of the SUS bit for DMA channel 0.	R

33.9 Functions

33.9.1 System Control and Oscillation Control

This chapter describes register manipulations required to perform initial setup of this module. This chapter also describes the registers required to control power consumption.
For the parts of the sequence that are required in both host and peripheral modes, see section 32.9.1, Host/Peripheral Common Setting Sequence.

33.9.1.1 USB data bus resistor control

This module controls switchover between the pull-up resistors for the D+ signal and the pull-down resistors for the D- signal of the USBPHY. Use the DPRPU bit and DRPD bit in the SYSCFG0 register to set pull-up or pull-down of each signal.
Recognize that a connection to the USB host is established, and then set 1 for the DPRPU bit in the SYSCFG0 register to pull up the D+ signal.

After connected to the host, this module automatically switches the resistor when the state changes to reset handshake, suspend, or resume.

If 0 is set for the DPRPU bit in the SYSCFG0 register during communication with the host, the pull-up resistor (or termination resistor) for the USB data line is disabled. This can notify the host controller of disconnection of a device.

33.9.2 Interrupt Function

33.9.2.1 Overview of Interrupt Function (other than DMA Master)

The following shows a list of interrupt functions of this module. An interrupt is notified as USBFIm (m = 0, 1). Check the status register to identify the interrupt factor.

Table 33.22 List of Interrupt functions

Bit	Interrupt name	Interrupt factor	Related status
VBINT	VBUS interrupt	The status change of the VBUS input pin is detected. (Changes from L to H and from H to L are detected.)	VBSTS
RESM	Resume interrupt	In the suspended state, a change of the USB bus status is detected (from J-State to K-State or from J-State to SE0).	—
SOFR	Frame number update interrupt	If SOFRM is 0: An SOF packet with a different frame number is received. If SOFRM is 1: An SOF with µframe number 0 cannot be received due to a problem such as packet corruption.	—
DVST	Device state transition interrupt	A transition of a device state is detected. USB bus reset detected Suspended state detected Set Address request received Set Configuration request received	DVSQ
CTRT	Control transfer stage transition interrupt	A transition of a control transfer stage is detected. Setup stage completed ControlWrite transfer status stage transition ControlRead transfer status stage transition Control transfer completed Control transfer sequence error	CTSQ
BEMP	Buffer empty interrupt	All data in the buffer memory is sent and the buffer becomes empty. A packet exceeding the maximum packet size is received.	PIPEBEMP
NRDY	Buffer not ready interrupt	A token is received when the PID setting is BUF and the buffer memory is not available for sending and receiving data. A CRC error or bit stuff error occurs when data is received in isochronous transfer. An interval error occurs when data is received in isochronous transfer.	PIPENRDY
BRDY	Buffer ready interrupt	The buffer becomes ready (available for reading or writing data).	PIPEBRDY

The following shows the relationship between interrupts of this module.

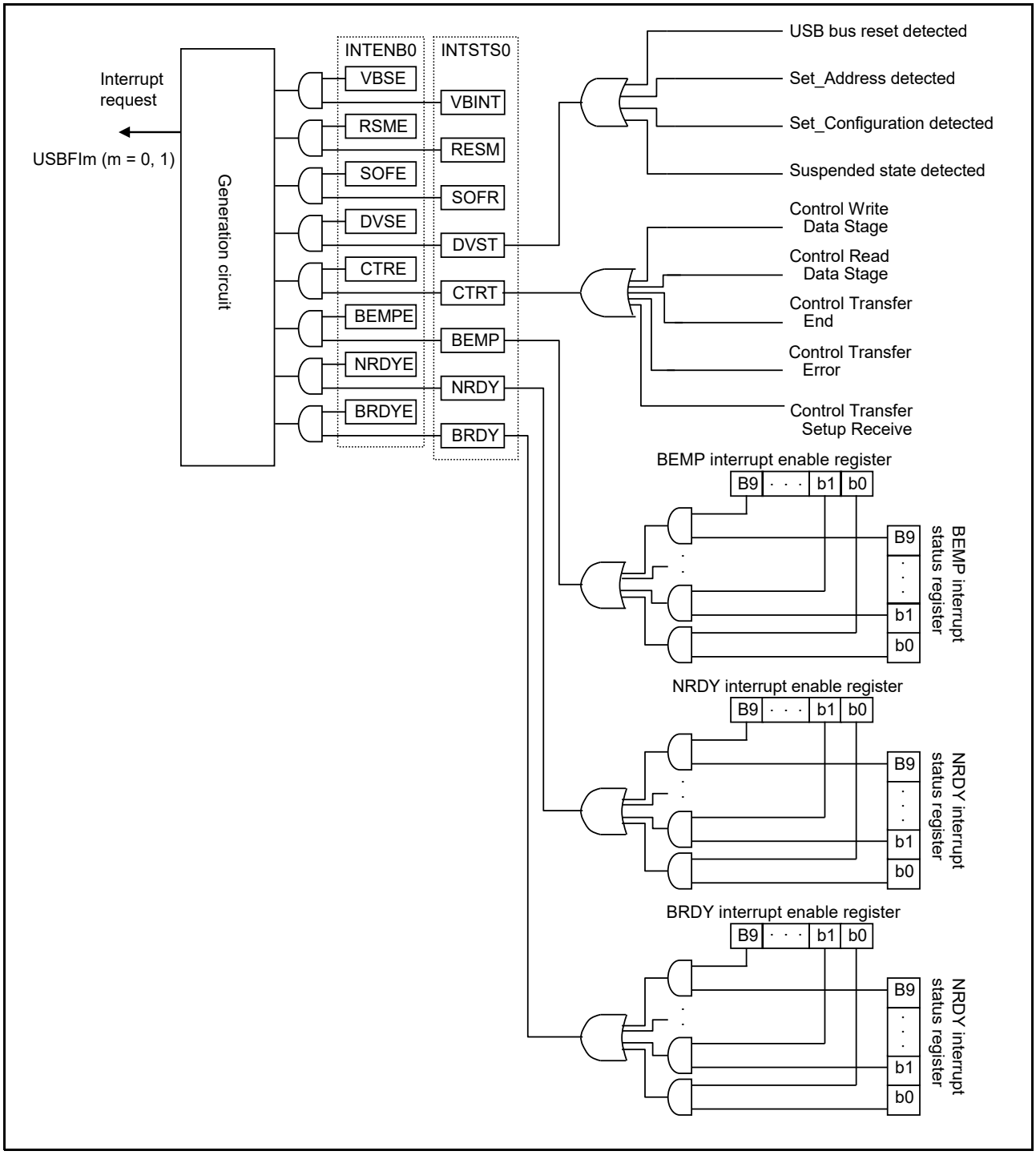


Figure 33.5 Interrupt Association Diagram

33.9.2.2 Device State Transition Interrupts

Figure 33.6 shows the device state transition diagram of this module.
This module manages device states and generates device state transition interrupts. However, resumption from the suspended state (resume signal detection) is detected by a resume interrupt.

Device state transition interrupts can be enabled and prohibited by using the INTENB0 register. A device state for which a transition has occurred can be checked in the DVSQ bits in the INTSTS0 register.
To trigger a transition to the default state, a device state transition interrupt is generated after a reset hand-shake protocol ends.

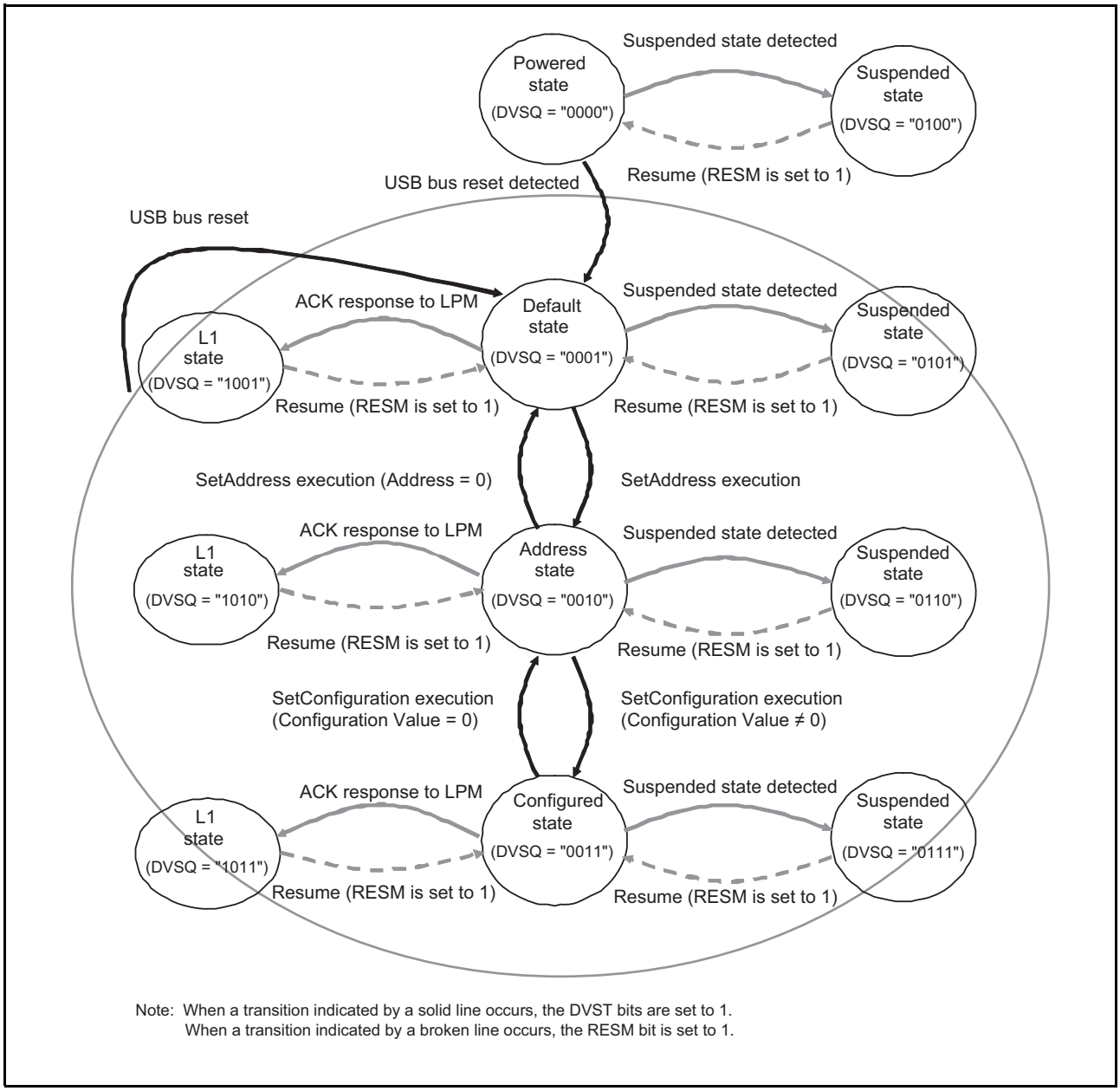


Figure 33.6 Device State Transition Diagram

33.9.2.3 Control Transfer Stage Transition Interrupts

Figure 33.7 shows the control transfer stage transition diagram of this module. This module manages a sequence of control transfers, and generates a control transfer stage transition interrupt. Control transfer stage transition interrupts can be enabled and prohibited by using the INTENB0 register. A transfer stage for which a transition has occurred can be checked in the CTSQ bits in the INTSTS0 register.

The following describes sequence errors that can occur during control transfer.
If an error occurs, the PID bits in the DCPCTR register are set to 1X (STALL).

(1) For Control Read transfer

(a) An OUT or PING token is received in a situation where data has not been transferred yet for an IN token of the data stage.
(b) An IN token is received in the status stage.
(c) A data packet "DATAPID = DATA0" is received in the status stage.

(2) For Control Write transfer

(a) An IN token is received in a situation where an ACK response has not been sent yet for an OUT token of the data stage.
(b) The first data packet ”DATAPID = DATA0” is received in the data stage.
(c) An OUT or PING token is received in the status stage.

(3) For Nodata Control transfer

(a) An OUT or PING token is received in the status stage.

If the amount of received data exceeds the value of the wLength field in a USB request in the Control Write transfer data stage, this module cannot identify this situation as a control transfer sequence error. If a packet other than a zero-length packet is received in the Control Read transfer status stage, this module sends an ACK response and then terminates processing normally.

If a CTRT interrupt is generated (SERR bit is set to 1) due to a sequence error, the CTSQ bits retain 110 until the system writes 0 to the CTRT bits to clear the interrupt status.
Therefore, as long as the CTSQ bits retain 110, even if a new USB request is received, a CTRT interrupt that reports completion of a setup stage is not generated. (Information on completion of the setup stage is retained by this module, and a CTRT interrupt is generated after the interrupt status is cleared.)

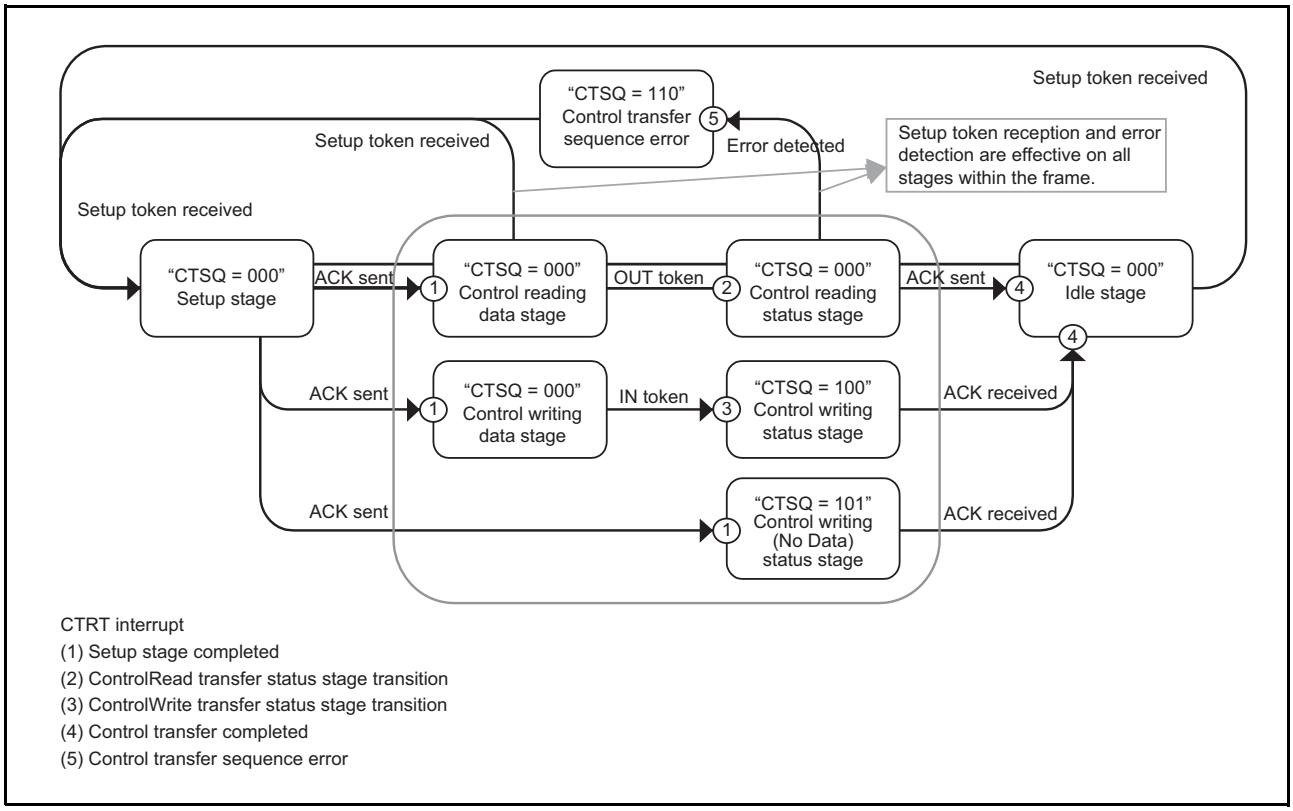


Figure 33.7 Control Transfer Stage Transition Diagram

33.9.2.4 Interrupts Relating to DMA Master

Table 33.23 List of Interrupts

Interrupt Name	Interrupt Type
USBFDMAmn (m, n = 0, 1)	A DMA transaction ends.
	An invalid descriptor is read in link mode.
USBFDMAERRm (m = 0, 1)	An error response is returned for a transfer issued by the master interface.

This module has two types of DMA interrupts USBFDMAmn and USBFDMAERRm.

- USBFDMAmn interrupt (m, n = 0, 1)

This interrupt occurs when a DMA transaction ends or when an invalid descriptor is read in link mode.
An interrupt is divided for each DMA channel. USBFDMAm0 corresponds to the interrupt for DMA 0ch, and USBFDMAm1 corresponds to the interrupt for DMA 1ch.

- USBFDMAERRm interrupt

This interrupt occurs when an error response is returned for a transfer issued by the master interface. This interrupt applies to all DMA channels.

USBFDMAmn interrupt output can be temporarily masked by setting a register.
Interrupt detection can also be masked by setting the register. If interrupt detection is masked, the status register that indicates generation of an interrupt does not change.
On the other hand, a USBFDMAERRm interrupt signal does not have the masking function.

33.9.3 Pipe Control

Table 33.24 lists the pipe settings for this module. For USB data transfer, logical pipes called endpoints are used to enable data communication. This module provides 16 pipes for data transfer. Set each pipe according to the specification of the system.

Table 33.24 PIPE Settings

Register Name	Bit Name	Setting	Comment
DCPCFG PIPECFG	TYPE	Specifies the transfer type.	Pipes 1 to 15: Settable
	BFRE	Selects BRDY interrupt mode.	Pipes 1 to 5, 11 to 15: Settable
	DBLB	Selects double-buffer configuration.	Pipes 1 to 5, 11 to 15: Settable Pipes 9, 10: Settable only in bulk transfers
	CNTMD	Selects continuous transfer or non-continuous transfer.	DCP: Settable Pipes 1, 2, 9, 10: Settable only in bulk transfers Pipes 3 to 5, 11 to 15: Settable
	DIR	Selects transfer direction.	Settable to IN or OUT
	EPNUM	Endpoint number	Pipes 1 to 15: Settable Set to a value other than "0000" when a pipe is in use.
	SHTNAK	Disables pipes when transfer is completed.	DCP: Settable Pipes 1, 2, 9, 10: Settable only in bulk transfers Pipes 3 to 5, 11 to 15: Settable
PIPEBUF	BUFSIZE	Buffer memory size	DCP: Unsettable (fixed to 64/256 (CNTMD = 1) bytes) Pipes 1 to 5, 9 to 15: Settable (up to 2 Kbytes specifiable) Pipes 6 to 8: Unsettable (fixed to 64 bytes)
	BUFNMB	Buffer memory number	DCP: Unsettable (fixed to area 0 to 3 hex) Pipes 1 to 5, 9 to 15: Settable (area 7 to 7F hex specifiable) Pipes 6 to 8: Unsettable (fixed to area 4 to 6 hex)
DCPMAXP PIPEMAXP	MXPS	Maximum packet size	Setting conforming to the USB specification.
PIPEPERI	IFIS	Buffer flush	Pipes 1, 2: Settable only in isochronous transfers Pipes 3 to 15: Unsettable
	IITV	Interval counter	Pipes 1, 2: Settable only in isochronous transfers Pipes 3 to 15: Unsettable
DCPCTR PIPExCTR	BSTS	Buffer status	DCP state switched between receive and transmit buffer by ISEL bit
	INBUFM	IN buffer monitor	Available only for pipes 3 to 5, 9 to 15
	ACLRM	Auto buffer clear	Pipes 1 to 15: Settable
	SQCLR	Sequence clear	Clears data toggle bit.
	SQSET	Sequence set	Sets data toggle bit.
	SQMON	Sequence monitor	Monitors data toggle bit.
	PBUSY	Pipe busy monitor	
	PID	Response PID	
DCPCTR PIPExCTR	ATREPM	Auto response mode	Pipes 1 to 5, 9 to 15: Settable
PIPExTRE	TRENB	Transaction count enable	Pipes 1 to 5, 9 to 15: Settable
	TRCLR	Current transaction counter clear	Pipes 1 to 5, 9 to 15: Settable
PIPExTRN	TRNCNT	Transaction counter	Pipes 1 to 5, 9 to 15: Settable

33.9.3.1 Maximum packet size setting

Use the MXPS bits in the DCPMAXP and PIPEMAXP registers to specify the maximum packet size for each pipe. The default control pipe (DCP) and pipes 1 to 5 can be set to any of the maximum packet sizes defined by the USB specification. For pipes 6 to 9, 64 bytes are the upper limit of the maximum packet size. Set the maximum packet size before starting transfer (by setting "PID = BUF").

DCP: Set 64 for high-speed operation.
DCP: Set 8, 16, 32, or 64 for full-speed operation.
Pipes 1 to 5: Set 512 for high-speed bulk transfer.
Pipes 1 to 5: Set 8, 16, 32, or 64 for full-speed bulk transfer.
Pipes 1, 2: Set a value from 1 to 1024 for high-speed isochronous transfer.
Pipes 1, 2: Set a value from 1 to 1023 for full-speed isochronous transfer.
Pipes 6 to 8: Set 64.

Pipe 9:
- For interrupt transfer, set a value from 1 to 64.
- For high-speed bulk transfer, set 512.
- For full-speed bulk transfer, set 8, 16, 32, or 64.

Pipe 10:
- For interrupt transfer, set a value from 1 to 64.
- For high-speed bulk transfer, set 512.
- For full-speed bulk transfer, set 8, 16, 32, or 64.

Pipes 11 to 15:
- For high-speed bulk transfer, set 512.
- For full-speed bulk transfer, set 8, 16, 32, or 64.

High-bandwidth transfers used for interrupt transfers and isochronous transfers are not supported.

33.9.3.2 Response PID

Set the response PID for each pipe with the PID bits in the DCPCTR and PIPExCTR registers.

- Response PID setting

The response PID specifies the response to a transaction from the host.
(a) NAK: Always sends a NAK response to a generated transaction.
(b) BUF: Responds to a transaction in accordance with the buffer memory state.
(c) STALL: Always sends a STALL response to a generated transaction.

Regardless of the value set in the PID bits, an ACK is always sent as a response to a setup transaction and a USB request is stored in corresponding registers.

This module might write data to the PID bits depending on the transaction result.
This module writes data to the PID bits in the following cases:

(a) NAK:
(A) The SETUP token is received normally (for the DCP only).
(B) If 1 is set for the SHTNAK bit in the PIPECFG register during bulk transfer, a short packet is received.
(C) If 1 is set for the SHTNAK bit during bulk transfer, the transaction counter finishes.

(b) BUF:
This module does not write "BUF".

(c) STALL:
(A) When a maximum packet size over error is detected in a received data packet
(B) When a control transfer sequence error is detected

33.9.3.3 Pipe control register switching procedure

The following bits in the pipe control registers can be modified only when USB transmission is disabled (PID = NAK). Figure 33.8 shows the procedure for changing the pipe control register state from the USB transmission enabled (PID = BUF) state.

The registered that cannot be manipulated in the USB transmission enabled (PID = BUF) state are as follows:

- All bits in the DCPCFG and DCPMAXP registers
- SQCLR and SQSET bits in the DCPCTR register
- All bits in the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers
- ATREPM, ACLRM, SQCLR, and SQSET bits in the PIPExCTR register
- All bits in the PIPExTRE and PIPExTRN registers

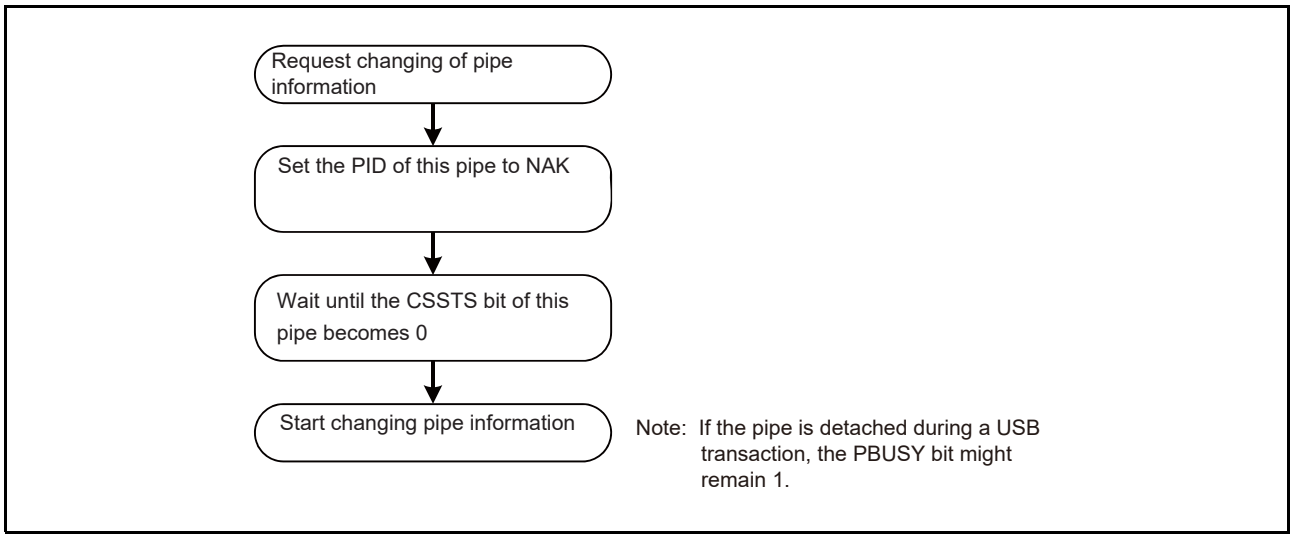


Figure 33.8 Procedure for Changing Pipe Information from USB Transmission Enabled (PID = BUF) State

In addition, for the settings of the following bits in pipe control registers, only the pipe information that is not set for the CURPIPE bit in the CPU, DMA0, or DMA1 FIFO port can be changed.

Registers that cannot be set with pipe information that is set for the CURPIPE bit in a FIFO port:

- All bits in the DCPCFG and DCPMAXP registers
- All bits in the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers
- ACLRM bit in the PIPExCTR register

If you change the pipe information, make sure that the setting of CURPIPE is different from the new pipe number. For the default control pipe (DCP), after modifying the pipe information, use the BCLR bit to clear the buffer.

33.9.3.4 Data PID sequence bit

When a normal data transfer is performed during bulk transfer or interrupt transfer, or in the data stage of control transfer, this module automatically toggles the sequence bit of a data PID. The sequence bit of the next data PID to be transferred can be confirmed with the SQMON bit in the DCPCTR or PIPExCTR registers. For data transmission, the sequence bit is switched when an ACK handshake is received. For data reception, the sequence bit is switched when an ACK handshake is sent. The data PID sequence bit can be changed by using the SQCLR and SQSET bits in the DCPCTR and PIPExCTR registers.

During control transfer, this module automatically sets the sequence bit when the stage changes. When the setup stage finishes, the data PID is set to DATA1. In the status stage, this module responds with "PID = DATA1" without referencing the sequence bit.

Note that, the data PID sequence bit must set, when a clear feature request is sent or received.

Also note that for isochronous transfer setting pipes, you cannot use the SQSET bit to manipulate the sequence bit.

33.9.4 FIFO Buffer

This section describes the processing related to the FIFO buffer of this module.

33.9.4.1 FIFO buffer allocation

Figure 33.9 shows an example of FIFO buffer memory mapping of this module. The FIFO buffer area is shared by the CPU and this module. The FIFO buffers can be accessed by either the system (CPU) or this module (SIE).

An independent FIFO buffer area is allocated for each pipe. The memory area is made up of memory blocks (1 block = 64 bytes) and is specified by the starting block number and the number of blocks (specified by the BUFNMB and BUFSIZE bits in the PIPEBUF register). When the CNTMD bit in the PIPExCFG register is used to set "continuous transfer mode", the value set with the BUFSIZE bit must be an integral multiple of the maximum packet size. If the double-buffer configuration is selected by the DBLB bit in the PIPExCFG register, two memory areas the size of which is specified by the BUFSIZE bit in the PIPEBUF register are allocated to a single pipe.

FIFO ports are used to access the FIFO buffer (data read/write). The pipe number of a pipe to be assigned to a FIFO port is specified by the CURPIPE bits in the CFIFOSEL/DxFIFOSEL register.

The FIFO buffer status of each pipe can be confirmed by using the BSTS and INBUFM bits in the DCPCTR and PIPExCTR registers. The access right of a FIFO port can be confirmed by using the FRDY bit in the CFIFOCTR/ DxFIFOCTR register.

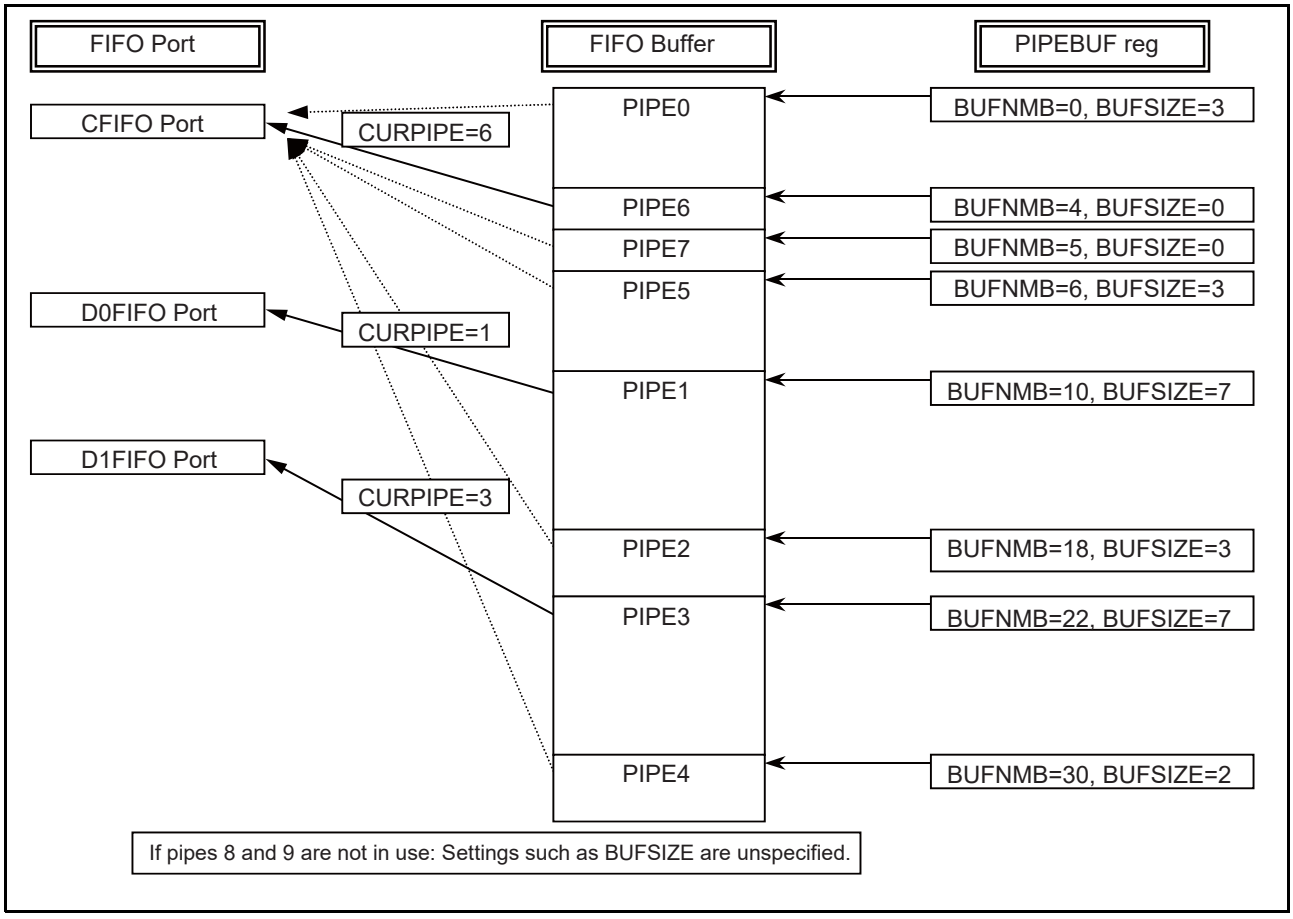


Figure 33.9 Example of FIFO Buffer Memory Mapping

33.9.4.2 Clearing FIFO buffers

Table 33.25 lists the modes in which this module can clear the FIFO buffer. Clearing of the FIFO buffer is controlled by the following bits.

Table 33.25 List of FIFO Buffer Clearing Modes

Bit Name	BCLR	DCLRM	ACLRM
Register	CFIFOCTR register DxFIFOCTR register	DxFIFOSEL register	PIPExCTR register
Function	The FIFO buffer on the CPU side is cleared.	The FIFO buffer is automatically cleared after the data is read from the specified pipe.	The buffer is automatically cleared to discard all the received packets.
Clearing method	Write "1" to clear.	1: Mode enabled 0: Mode disabled	1: Mode enabled 0: Mode disabled

33.9.5 FIFO Port Functions

This section describes FIFO port functions. Table 33.26 shows the FIFO port function settings for this module.
If data write access is enabled and data is written up to buffer full state (in non-continuous transfer: maximum packet size), the port automatically goes to the USB bus transmittable state. To enable transmission of data smaller than the buffer full (in non-continuous transfer: less than the maximum packet size), the BVAL bit in the CFIFOCTR/DxFIFOCTR register must be used to set write end.
To send a zero-length packet, the BCLR bit of that register must be used to clear the buffer before the BVAL bit is used to write end.
When all the data is read in a read access, the port automatically enters the state in which new packets can be received. However, when a zero-length packet is received (DTLN = 0), no data can be read, and therefore the buffer must be cleared by using the BCLR bit.
The receive data length is confirmed with the DTLN bits in the CFIFOCTR/DxFIFOCTR register.

Table 33.26 FIFO Port Function Settings

Register Name	Bit Name	Function	Note
C/DxFIFOSEL	RCNT	Selects DTLN read mode	
	REW	Rewinds buffer memory (re-read, re-write)	
	DCLRM	Reads received data of the specified pipe, and then automatically clears the received data	DxFIFO only
	DREQE	Asserts DREQ signal	DxFIFO only
	MBW	Specifies FIFO port access bit width	
	BIGEND	Selects FIFO port endian	CFIFO only
	ISEL	Specifies FIFO port access direction	DCP only
	CURPIPE	Selects current pipe	
C/DxFIFOCTR	BVAL	Finishes buffer memory write	
	BCLR	Clears CPU-side buffer memory	
	FRDY	Monitors FIFO port ready	
	DTLN	Confirms received data length	

33.9.5.1 FIFO port selection

Table 33.27 lists pipes that can be selected for each FIFO port.
Use the CURPIPE bits in the C/DxFIFOSEL register to select the pipe to be accessed. After selecting the pipe, confirm that the value written to the CURPIPE bits can be read correctly (if the previous pipe number is read out, this module is currently changing the pipe), confirm that FRDY = 1, and then access the FIFO port.

Figure 33.10 shows the procedure for switching the pipe when accessing a FIFO port.

In addition, use the MBW bit to select the bus width with which to access the FIFO port. If the target pipe is the default control pile (DCP), the ISEL bit determines the buffer memory access direction. If the target pipe is not the DCP, the DIR bit in the PIPExCFG register determines the buffer memory access direction.

Table 33.27 FIFO Port Access for Each Pipe

Pipe	Access Method	Usable Port
DCP	CPU access	CFIFO port register
Pipes 1 to 15	CPU access	CFIFO port register
	DMA access	DxFIFO port register

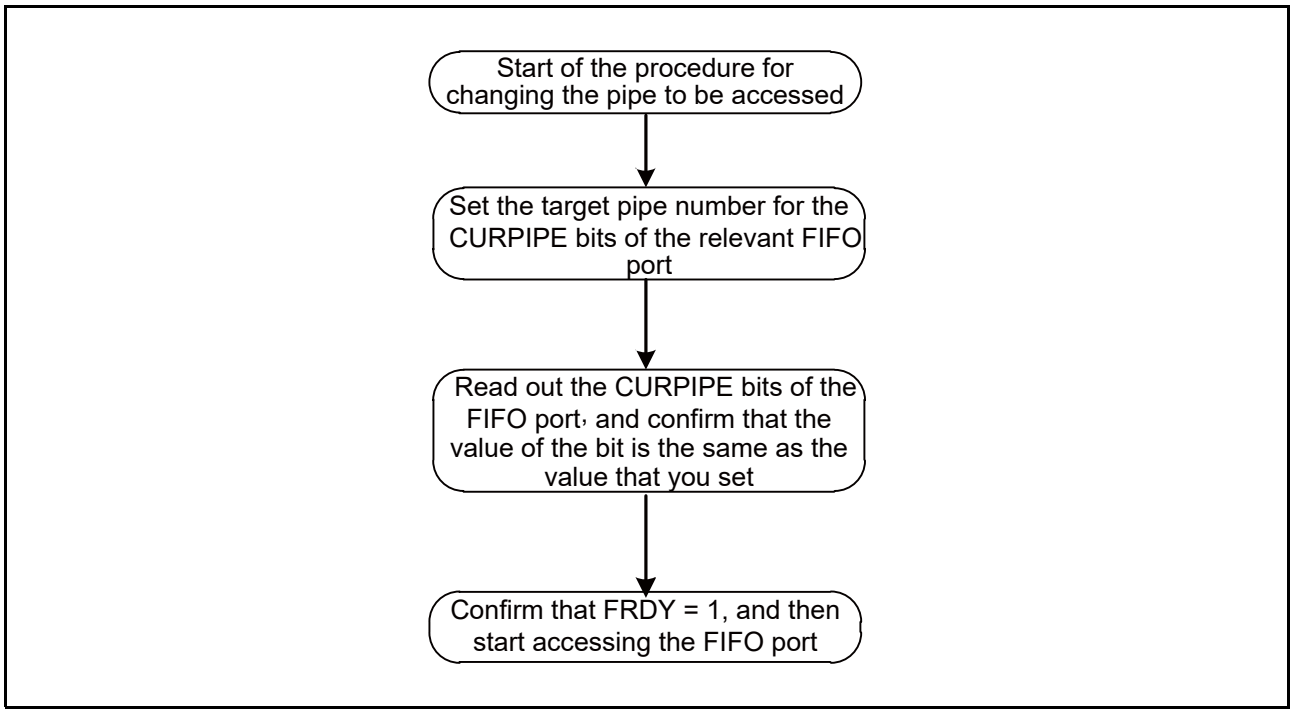


Figure 33.10 Pipe Switching Procedure for FIFO Port Access

33.9.5.2 DxFIFO automatic clear mode (DxFIFO port read direction)

This module automatically clears the buffer memory for a pipe when data is read out from the buffer memory if the DCLRM bit in the DxFIFOSEL register is set to 1.

Table 33.28 shows the correspondence between packet reception and buffer memory clear processing by the software for each setting.
As shown in Table 33.28, the buffer clearing conditions vary with the value that is set for the BFRE bit. However, using the DCLRM bit eliminates the need for buffer clear by the software even in states where clearing is required, which enables DMA transfers without using the software.

Note that for this function, only the buffer memory read direction can be set.

Table 33.28 Relationship between Packet Reception and Buffer Memory Clear Processing by the Software

	Register Setting			
	DCLRM = 0		DCLRM = 1	
Buffer State during Packet Reception	BFRE = 0	BFRE = 1	BFRE = 0	BFRE = 1
Buffer full	Clearing unnecessary		Clearing unnecessary	
Zero-length packet received	Clearing necessary			
Normal short packet received	Clearing unnecessary	Clearing necessary		
Transaction count end				

33.9.5.3 BRDY interrupt timing selection function

The BFRE bit in the PIPECFG register can be set so that a BRDY interrupt is not generated when a data packet of maximum packet size is received.

When using a DMA transfer, this function enables an interrupt to be generated only when the last data is received. The "last data" here indicates either a short packet reception or the transaction count end. If the BFRE bit is set to 1, a BRDY interrupt is generated after the received data is read. By reading the DTLN bits in the DxFIFOCTR register, the receive data length of last data packet received just before the BRDY interrupt is generated can be confirmed.

Table 33.29 shows when this module generates a BRDY interrupt.

Table 33.29 BRDY Interrupt Generation Timing

	Register Setting	
Buffer State during Packet Reception	BFRE = 0	BFRE = 1
Buffer full (normal packet received)	When packet is received	No interrupt generated
Zero-length packet received	When packet is received	When packet is received
Normal short packet received	When packet is received	When received data has been read from the buffer memory
Transaction count end	When packet is received	When received data has been read from the buffer memory

The BFRE bit function is valid only in the reading direction from the buffer memory. For the writing direction, fix the BFRE bit to 0.

33.9.6 Control Transfer (DCP)

In the data stage of control transfer, the default control pipe (DCP) is used to transfer data. For the DCP, a single 64-byte buffer is allocated as a fixed area that is used for both control reading and writing (in continuous transfer mode (CNTMD = 1), the size of this area is fixed to 256 bytes). The buffer memory can be accessed through the CFIFO port only.

33.9.6.1 Control transfer

(1) Setup stage

This module always responds with ACK to any normal setup packet for this module. The following shows the behavior of this module in the setup stage:

1. When this module receives a new setup packet, this module sets the following bits:
   (a) VALID bit in the INTSTS0 register: 1
   (b) PID bit in the DCPCTR register: NAK
   (c) CCPL bit in the DCPCTR register: 0

2. When this module receives a data packet after receiving a setup packet, this module stores USB request parameters in the USBREQ, USBVAL, USBINDX, and USBLENG registers.

Response processing for control transfer must be performed after VALID is set to 0. While VALID is 1, PID cannot be set to BUF, and therefore the data stage cannot end.
By using a function of the VALID bit, when this module receives a new USB request during control transfer, this module can respond to the newest request, canceling the request that is being processed.

This module also automatically recognizes the type of transfer (ControlRead, ControlWrite, or NodataControl) from the direction bit (bit 8 of bmRequestType) and request data length (wLength) of the received USB request to manage stage transition. For an incorrect sequence, a control transfer stage transition interrupt occurs to report a sequence error to software. For details about stage management by this module, see Figure 33.7.

(2) Data stage

Use the DCP to transfer data in response to the received USB request. Before the DCP buffer memory is accessed, use the ISEL bit in the CFIFOSEL register to specify the access direction.
A transaction is executed by setting the PID bit in the DCPCTR register to BUF.
The end of data transfer is detected with a BRDY or BEMP interrupt. For ControlWrite transfer, use a BRDY interrupt. For ControlRead transfer, use a BEMP interrupt.
For ControlWrite transfer in high-speed operation mode, an NYET handshake response is performed in accordance with the buffer memory state.

(3) Status stage

If the PID bit in the DCPCTR register is BUF, setting the CCPL bit to 1 terminates control transfer.
After control transfer is terminated by the above setting, this module automatically executes the status stage according to the data transfer direction determined in the setup stage, as shown below:

(a) For ControlRead transfer

Upon receiving a zero-length packet from the USB Host Controller, this module sends an ACK response.

(b) For ControlWrite or NodataControl transfer

This module sends a zero-length packet, and then receives an ACK response from the USB Host Controller.

(4) Control transfer automatic response function

This module automatically responds to any normal SET_ADDRESS request. However, if a SET_ADDRESS request has any of the following errors, software must respond, instead of this module:

(a) bmRequestType ≠ "0x00"
(b) wIndex ≠ "0x00"
(c) wLength ≠ "0x00"
(d) wValue > "0x7F"
(e) DVSQ = "011 (Configured)"

Software must respond to all requests other than SET_ADDRESS.

33.9.7 Bulk Transfer (Pipes 1 to 5 and 9 to 15)

The user can select the buffer memory usage method (single/double buffer, continuous/non-continuous transfer mode) for bulk transfers. The buffer memory size can be set up to 2 KB. The controller manages the buffer memory state and automatically responds to PING packets and NYET handshakes.

33.9.7.1 NYET handshake control

Table 33.30 lists responses to tokens received in a bulk or control transfer.
When an OUT token is received in a bulk or control transfer and there is only an open space for one packet in the buffer memory, this module sends a NYET response. However, when a short packet is received, this module sends an ACK response instead of a NYET response even under these conditions.

Table 33.30 List of Responses to Received Tokens

PID bit value	Buffer memory state*1	Received token	Response	Note
NAK/STALL	-	SETUP	ACK	-
	-	IN/OUT/PING	NAK/STALL	-
BUF	-	SETUP	ACK	-
	RCV-BRDY	OUT/PING	ACK	Receives data packet when OUT token is received *1
	RCV-BRDY	OUT	NYET	Receives data packet *2
	RCV-BRDY	OUT (Short)	ACK	Receives data packet *2
	RCV-BRDY	PING	ACK	*2
	RCV-NRDY	OUT/PING	NAK	
	TRN-BRDY	IN	DATA0/1	Sends data packet
	TRN-NRDY	IN	NAK	

Note: Details are described below.
RCV-BRDY*1: Buffer memory has a space for 2 packets or more when an OUT or PING token is received.
RCV-BRDY*2: Buffer memory has only a space for one packet when an OUT token is received.
RCV-NRDY: Buffer memory has no space for any packet when a PING token is received.
TRN-BRDY: Buffer memory has send data when an IN token is received.
TRN-NRDY: Buffer memory has no send data when an IN token is received.

33.9.8 Interrupt Transfer (Pipes 6 to 9 and 10)

This module performs an interrupt transfer in accordance with the period managed by the host controller. This module ignores (no response) PING packets in interrupt transfers. In addition, this module does not send a NYET handshake, but sends an ACK, NAK or STALL response.
Note that this module does not support high-bandwidth interrupt transfers.

33.9.9 Isochronous Transfer (Pipes 1 and 2)

This module is provided with the following functions for isochronous transfers:

1. Notification of error information about isochronous transfers
2. Interval counter (IITV bit)
3. Data setup control for isochronous IN transfers (IDLY function)
4. Buffer flush function for isochronous IN transfers (IFIS bit)
5. SOF pulse output function

This module does not support high-bandwidth Isochronous transfers.

33.9.9.1 Isochronous transfer error detection

This module has the following error information detection functions for the software to manage errors that occur during isochronous transfer.
Table 33.31 and Table Table 33.32 describe the error checking procedure and interrupts that are generated.

1. PID error
   The PID of the received packet is invalid.
2. CRC error and bit stuffing error
   The received packet has a CRC error or invalid bit stuffing.
3. Max packet size over
   The data size of the received packet is larger than the preset maximum packet size.
4. Overrun error and underrun error
   (a) The buffer memory has no data when an IN token is received during IN-direction (send) transfer.
   (b) The buffer memory has no space although an OUT token is received during OUT-direction (receive) transfer.
5. Interval error
   An interval error occurs in the following cases:
   (a) An IN token cannot be received within the interval frame during isochronous IN transfer.
   (b) An OUT token cannot be received within the interval frame during isochronous OUT transfer.

Table 33.31 Error Detection during Transmission/Reception of Token

Error Detection Priority	Error Type	Interrupt Generated at Error Detection and Status
1	PID error	No interrupt is generated (ignored as a corrupted packet).
2	CRC error, bit stuffing error	No interrupt is generated (ignored as a corrupted packet).
3	Overrun error, underrun error	An NRDY interrupt is generated and the OVRN bit is set. A zero-length packet is sent in response to an IN token. A data packet is not received in response to an OUT token.
4	Interval error	An NRDY interrupt is not generated.

Table 33.32 Error Detection during Reception of Data Packet

Error Detection Priority	Error Type	Interrupt Generated at Error Detection and Status
1	PID error	No interrupt is generated (ignored as a corrupted packet).
2	CRC error, bit stuffing error	An NRDY interrupt is generated and the CRCE bit is set.
3	Packet size error (too large packet)	A BEMP interrupt is generated and the PID bit is set to STALL.

33.9.9.2 DATA-PID

This module does not support high-bandwidth transfers.
The following shows actions that can be taken in response to a received PID.

1. IN direction:

(a) DATA0: Used to send packets.
(b) DATA1: Not used to send packets.
(c) DATA2: Not used to send packets.
(d) mDATA: Not used to send packets.

2. OUT direction (in full-speed operation):

(a) DATA0: Packets are received normally.
(b) DATA1: Packets are received normally.
(c) DATA2: Packets are ignored.
(d) mDATA: Packets are ignored.

3. OUT direction (in high-speed operation):

(a) DATA0: Packets are received normally.
(b) DATA1: Packets are received normally.
(c) DATA2: Packets are received normally.
(d) mDATA: Packets are received normally.

33.9.9.3 Interval counter

(1) Outline of operation

The IITV bit in the PIPEPERI register can be used to set the interval of isochronous transfer.
The interval counter enables the functions listed in Table 33.33.

Table 33.33 Functions of the Interval Counter

Transfer Direction	Function	Detecting Condition
IN	Transmit buffer flush function	An IN token cannot successfully be received within the interval frame during isochronous IN transfer.
OUT	Notification of unreceived token	An OUT token cannot successfully be received within the interval frame during isochronous OUT transfer.

Counting of intervals is based on received SOF packets or interpolated SOFs. Therefore, even if SOF packets are damaged, the isochronism can still be maintained. Frame intervals are set as 2^n (micro) frames, where n is the value of the IITV bit.

(2) Interval counter initialization

This module initializes the interval counter under the following conditions:

(a) Power on reset
The IITV bit is initialized.

(b) Clearing of the buffer memory by the ACLRM bit
The IITV bit is not initialized but the counter is initialized.

(c) USB bus reset

After the interval counter is initialized and a packet is successfully transferred, counting of intervals starts under the following conditions:
1) An SOF packet is received after data is sent in response to an IN token when PID = BUF.
2) An SOF packet is received after data is received in response to an OUT token when PID = BUF.

Note that the interval counter is not initialized in the following conditions:

(a) The PID bit is set to NAK or STALL.
The interval timer is not stopped at this interval. The transaction will be attempted at the next interval.

(b) USB bus reset or USB suspension
The IITV bit is not initialized. When an SOF packet is received, counting starts from the value existing before reception.

33.9.9.4 Send data setup for isochronous transfer

This module becomes able to send data packets by isochronous transfer from the next frame after data is written to the buffer memory and then an SOF packet is detected. This is called "send data setup for isochronous transfer".
This function can identify the frame with which data sending started.

If the buffer memory is in a double-buffer configuration and writing to both buffers has been completed, only the buffer to which writing finished earlier can transfer data. Therefore, even when several IN tokens are received within the same frame, only one packet of data is sent from the buffer memory.

When an IN token is received, if the buffer memory is ready for sending data, the data is transferred and a normal response is returned. However, if the buffer memory is not ready for sending data, a zero-length packet is sent and an underrun error occurs.

Figure 33.11 shows examples of sending using the send data setup function for isochronous transfer by setting "IITV = 0" (each frame) in this module.

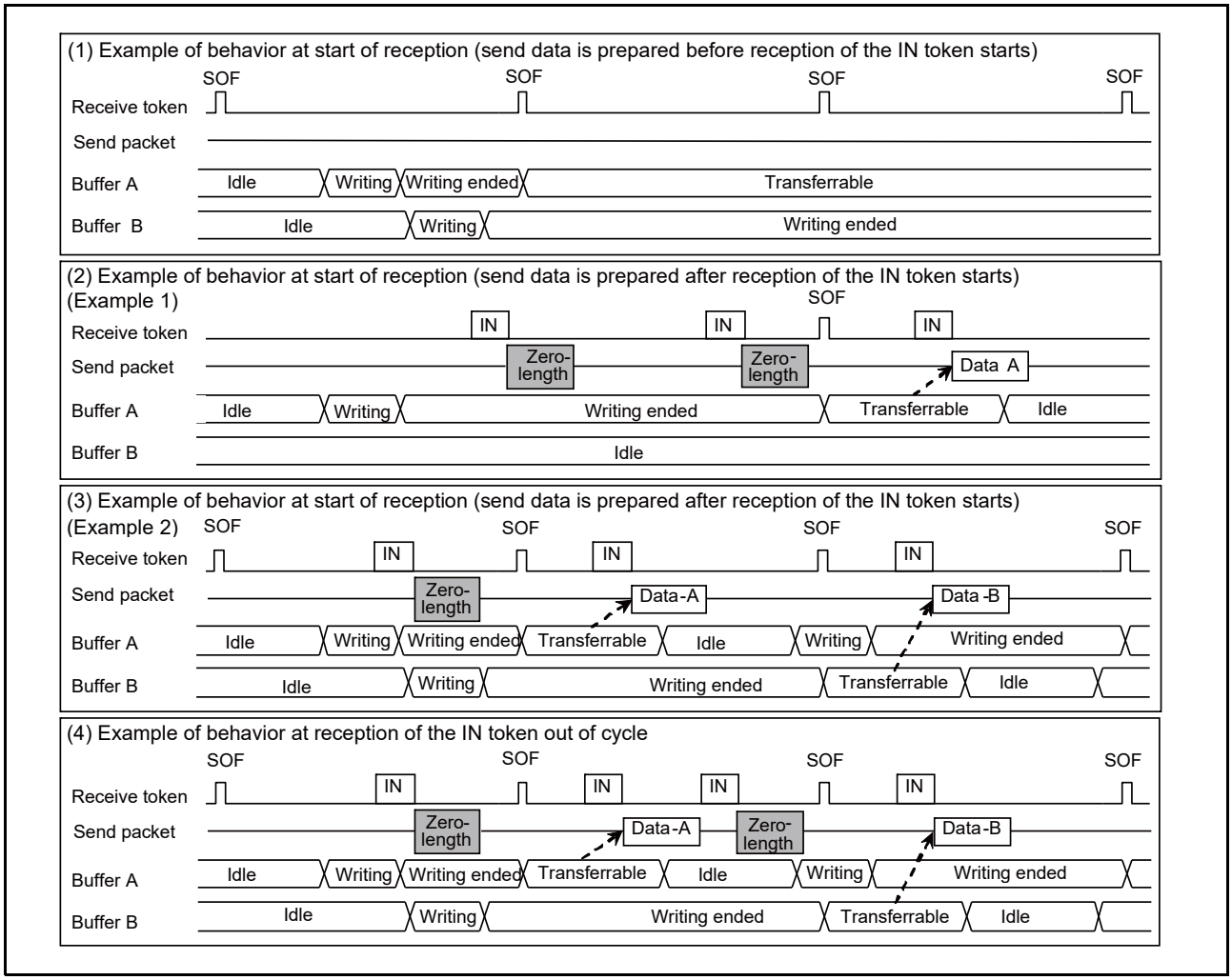


Figure 33.11 Examples of Data Setup Function Behavior

33.9.9.5 Transmit buffer flush for isochronous transfer

If this module does not receive an IN token in an interval frame and receives a (micro) SOF packet in the next frame during isochronous data transfer, this module handles the IN token as a corrupted token and clears the buffer that can send data to make the buffer writable.
At this time, if the buffer memory is in a double-buffer configuration and writing to both buffers has been completed, this module assumes the discarded buffer memory to be sent within the same interval frame. As a result, the buffer memory that is not discarded by reception of a (micro) SOF packet becomes to be able to transfer data.

The operation start timing of the buffer flush function varies with the value of the IITV bit.

1. If IITV is 0
   Buffer flush operation is performed from the first frame after the pipe is enabled.
2. If IITV is not 0
   Buffer flush operation is performed after the first successful transaction.

Figure 33.12 shows an example of how the buffer flush function of this module behaves. For a token outside the set interval (token prior to the interval frame), however, this module sends the written data or a zero-length packet as an underrun error according to the data setup state.

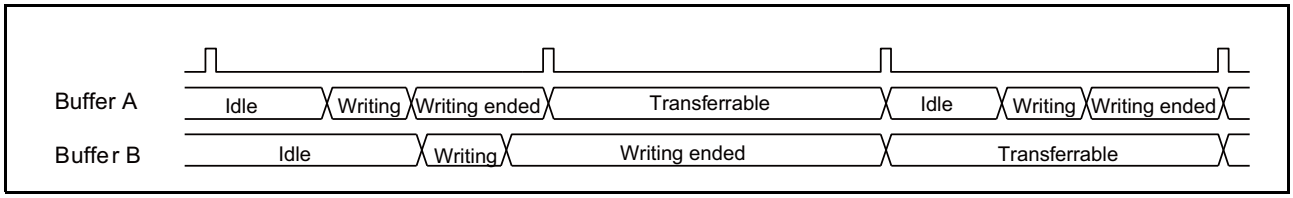


Figure 33.12 Example of Buffer Flush Function Behavior

Figure 33.13 shows an example of an interval error that occurs in this module.

There are five types of interval errors, as listed below. At timing (1) in the figure, an interval error occurs and the buffer flush function operates.
If an interval error occurs during IN transfer, the buffer flush function starts. If an interval error occurs during OUT transfer, an NRDY interrupt occurs.
Use the OVRN bit to determine whether an error is an NRDY interrupt (such as a receive packet error) or an overrun error.
In the figure, responses to tokens indicated as shaded boxes are made in accordance with the buffer memory state.

1. IN direction:
(a) If the buffer is ready to transfer data, data is transferred and a normal response is returned.
(b) If the buffer is not ready to transfer data, a zero-length packet is sent and an underrun error occurs.

2. OUT direction:
(a) If the buffer is ready to receive data, data is received and a normal response is returned.
(b) If the buffer is not ready to receive data, data is discarded and an overrun error occurs.

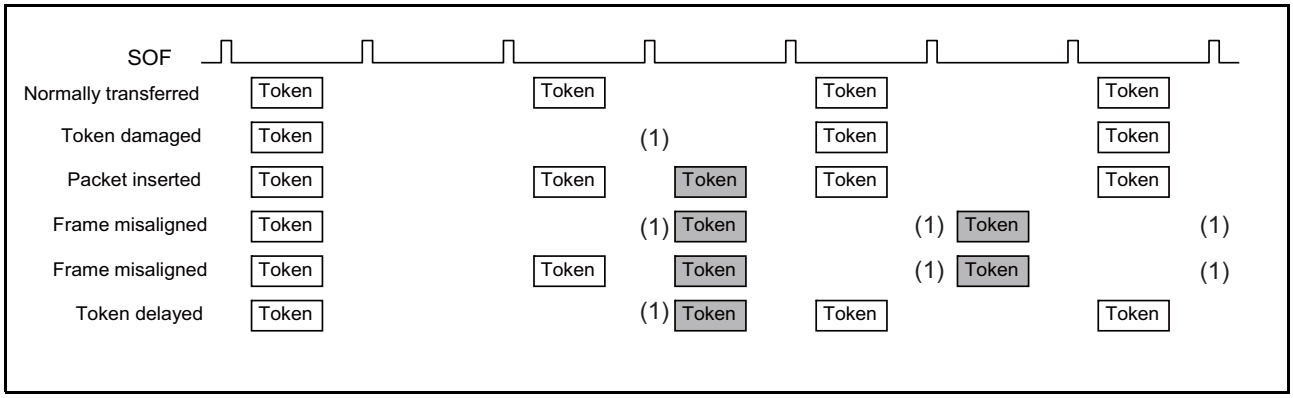


Figure 33.13 Example of Occurrence of Interval Error (when IITV is 1)

33.9.10 SOF Interpolation Function

If the controller cannot receive data at intervals of 1 ms (in full-speed operation) or 125 µs (in high-speed operation) due to corruption or missing of an SOF packet, the controller internally interpolates the SOF. The controller starts SOF interpolation upon receiving an SOF packet when both the USBE bit and SUSPM bit are set to 1.

The interpolation function is initialized under the following conditions:
(1) Power on reset
(2) USB bus reset
(3) Suspended state detected

The SOF interpolation operates according to the following specifications:
(1) Frame interval (125 µs or 1 ms) is based on the results of the reset handshake protocol.
(2) The interpolation function does not operate until an SOF packet is received.
(3) After receiving the first SOF packet, this module interpolates the SOF by using the 60-MHz internal clock to measure 125 µs or 1 ms.
(4) After receiving the second or a subsequent SOF packet, this module interpolates the SOF by using the previous reception interval.
(5) Interpolation is not performed in the suspended state or while a USB bus reset is being received.
(If this module enters the suspended state in high-speed operation, interpolation continues for 3 ms after receiving the last packet.)

The SOF interpolation function operates with the following functions:
(1) Updating of frame number or micro-frame number
(2) SOFR interrupt and micro SOF lock
(3) SOF pulse output
(4) Counting of isochronous transfer intervals

If an SOF packet is lost in full-speed operation, the FRNM bit in the FRMNUM register is not updated.
If a micro SOF packet is lost in high-speed operation, the UFRNM bit in the uFRMNUM register is updated.
However, if a micro SOF packet for which "micro-FRNM = 000" is set is lost, the FRNM bit is not updated. In this case, the FRNM bit is not updated even if subsequent micro SOF packets for which "micro-FRNM = 000" is not set are successfully received.

33.9.11 Link Power Management Processing

According to the Link Power Management specification, the existing suspend state is redefined as the L2 state and a new L1 state is defined as a state where transition and resumption at a lower latency than L2 (suspend) are possible.
The table below compares the features of the L2 (suspend) state and L1 state.

Table 33.34 Comparison Between Suspend (L2) State and L1 State

Item	L1	Suspend (L2)
Transition	LPM Transaction	3-ms idle period
Host-activated resumption	(Host) Minimum drive period specifiable by the host. Specified between 75 µs to 1.175 ms (Device) 10-µs K drive	(Host) Min. 20-ms K drive (Device) 10-ms K drive
Device-activated resumption	(Device) 50-µs K drive (Host) 60- to 990-µs K drive (Device) 10-µs K drive	(Device) 1-ms to 15-ms K drive (Host) Min. 20-ms K drive (Device) 10-ms K drive
Signaling	Low and Full Speed Idle	Low and Full Speed Idle

The following describes the processing for transition to and resumption from the L1 state.

33.9.11.1 Descriptor

This module must return its own descriptor when receiving the GetDescriptor command.
Whether the contents of the descriptor to be returned need to be modified is dependent on whether this module responds to the transition to and resumption from the L1 state with an LPM transaction. The details are summarized in the table below.

Table 33.35 Relationship between LPM Response and Descriptor

LPM Response	bcdUSB	Presence of USB 2.0 Ex. Desc	USB 2.0 Ex. Desc LPM	Response When LPM Is Received	Remarks
Not respond	0200	Not present	--	Not Respond	Standard action in the case where this module does not respond to LPM
	0201	Present	LPM = 0	STALL	This is when rejection of response to LPM is explicitly declared. In this case, it is necessary to send a STALL response instead of making no response.
Respond	0201	Present	LPM = 1	ACK or NYET	Standard action in the case where this module responds to LPM

Whether to respond to transition to and resumption from L1 is declared by the LPM bit of the USB 2.0 extension descriptor. To provide this module with the USB 2.0 extension descriptor, it is necessary to set the bcdUSB field of the device descriptor to 0201 or greater.
When not responding to LPM, set the bcdUSB value to 0200 without providing this module with the USB 2.0 extension descriptor. In this case, it is necessary to ignore any LPM token received.
When not responding to LPM, it is also possible to set bcdUSB to 0201 and set the LPM bit of the USB 2.0 extension descriptor to 0 (noncompliant). In this case, however, it is not allowed to ignore LPM and is necessary to send a STALL response.
When responding to LPM, set bcdUSB to 0201 and the LPM bit of the USB 2.0 extension descriptor to 1 (compliant). This grants this module to send an NYET or ACK in response to an LPM token.

33.9.11.2 Basic processing

This module needs to execute the following processing.

1. Responds to the LPM token received from the host with "No response", "ACK", "NYET", or "STALL" according to this module's own state.
2. Transitions to the L1 state if it fails to detect the retransmission of an LPM token for 8 µs after making an ACK response.
3. Detects a K drive of the host and performs resume processing to the idle state.
4. Performs resume processing to the idle state based on the Remote Wake signal.

For 1, the software specifies the response method according to the values of the L1RESPEN, L1RESPMD, and L1NEGOMD bits in the PL1CTRL register. The hardware makes the response that is designated by the software upon receiving an LPM token.
For 2, both retransmission control and transition to the L1 state are processed by the hardware.
Transition to the L1 state can be identified through a DVST interrupt.
For 3, a RESM interrupt occurs on detection of host K in the L1 state.
For 4, starting the Remote Wake processing can be instructed to the hardware by setting the WKUP bit by the software. The specification stipulates that the software clears this bit on resumption from the L2 state. On the other hand, the hardware clears this bit on resumption from the L1 state.

33.9.11.3 HIRD value negotiation

The HIRD value contained in the LPM token is the K period of the host on resumption from the L1 state.

This module can respond with ACK if the received HIRD value falls within the desired range as specified by the L1NEGOMD and HIRDTHR bits in the L1CTRL register; otherwise, this module can respond with NYET and request the host to modify the HIRD value.

Note: This HIRD value negotiation function must also be supported on the host side.

33.9.12 DMA Mode

33.9.12.1 Register mode/link mode

The DMS bit in the CHCFG_n register can be used to switch between register mode and link mode.

Table 33.36 DMA mode settings

DMS (CHCFG)	Mode	Description
0	Register mode	Performs DMA transfer based on the values set by Next Register Set.
1	Link mode	Accesses the descriptor area, and executes DMA transfer based on the values set by descriptors. This module repeats descriptor reading and DMA transfer unless the descriptor settings are changed or the control register is used to stop the processing.

(1) Register mode

In register mode, this module executes DMA transfer based on the values set in internal registers.
Two sets of the transfer-source address, transfer-destination address, and number of bytes to be transferred can be held (in Next0 Register Set and Next1 Register Set registers). One of these Next registers can be used to execute transfer, and both Next registers can be used to execute continuous transfer.

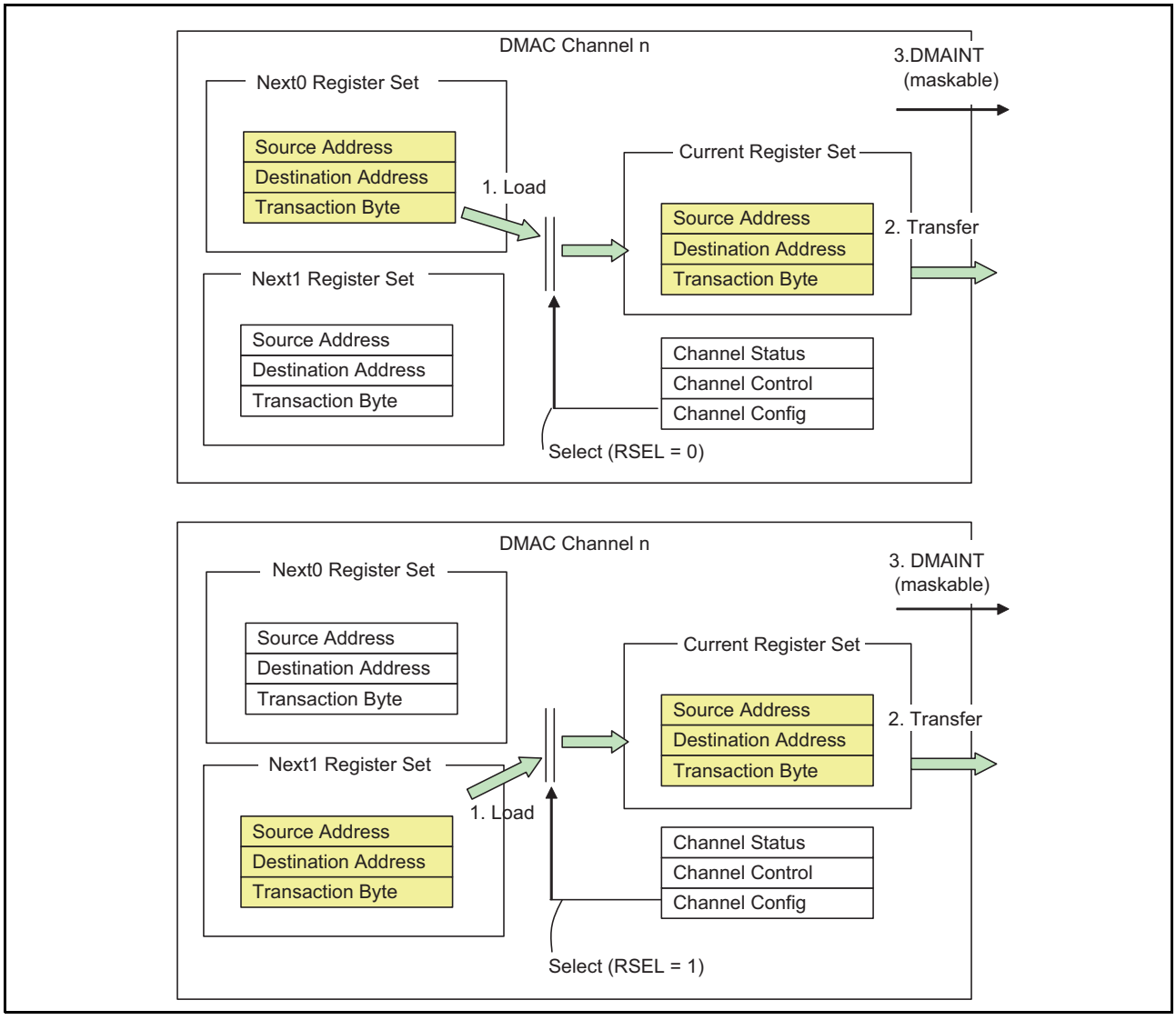


Figure 33.14 Overview of Normal Behavior of Register Mode

The upper part of the above figure indicates a case when Next0 Register Set is processed. The lower part of the above figure indicates a case when Next1 Register Set is processed.

(a) Operation flow of register mode

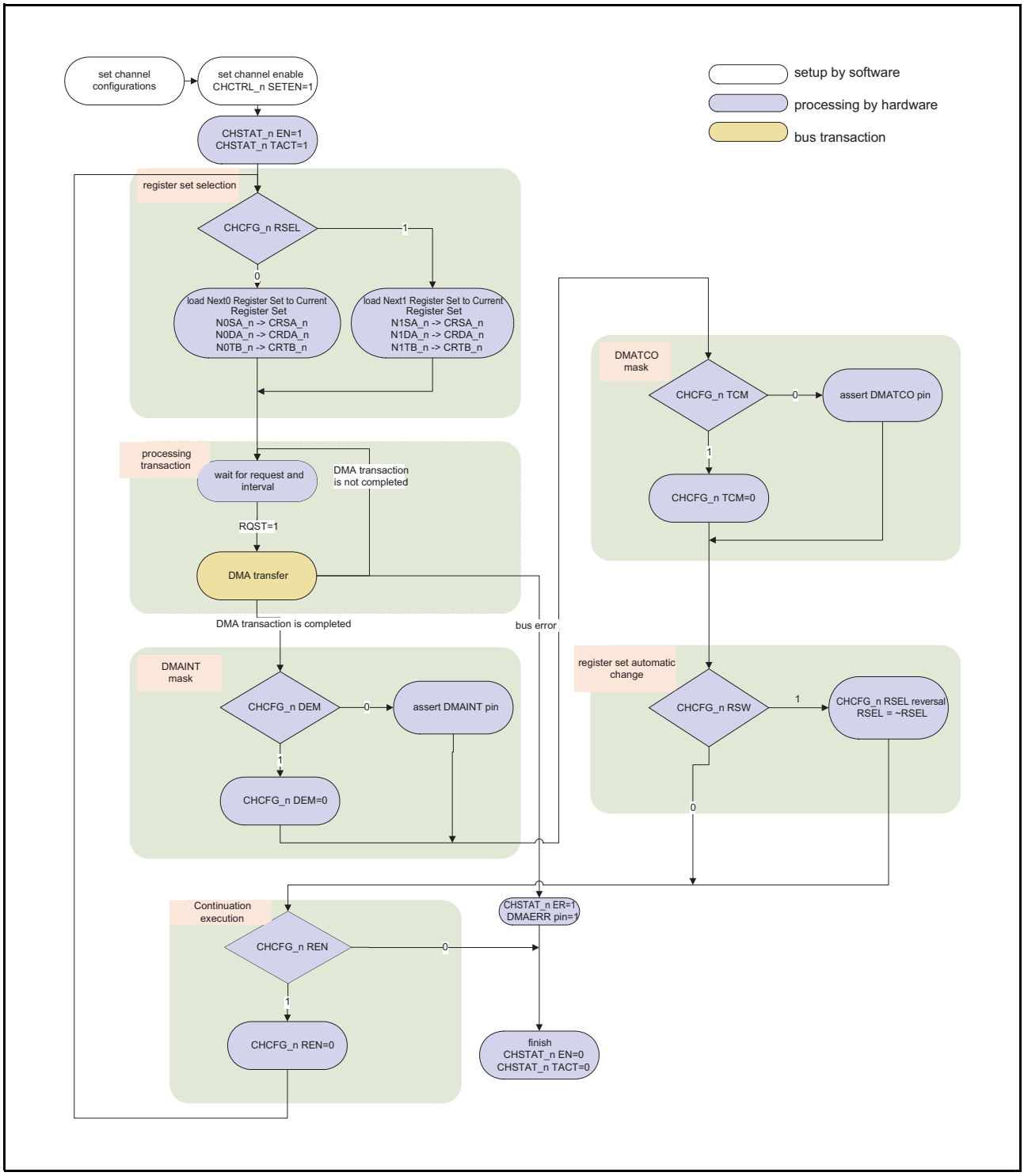


Figure 33.15 Register Mode Operation Flow

Description of the register mode operation flow:

1. Channel setting

The Next0 Register Set or Next1 Register Set register is set (transfer-destination address, transfer-source address, and total number of bytes to be transferred). In addition, the FIFO channel, volume of transfer data, and other items are set for USB control that is used for the Channel Register Set register.
No software settings are required in OUT transfer (reception in peripheral mode) or in IN transfer (transmission in peripheral mode) because the hardware sets the transfer source or destination address according to the setting of the CURPIPE bits in the DxFIFOSEL register.
(see section 33, DMA Transfer).

2. Register set selection

When 1 is written to the SETEN bit in the CHCTRL_n register, the EN and TACT bits in the CHSTAT_n register are set to 1. As a result, the values set by the Next Register Set register selected by the RSEL bit in the CHCFG_n register are loaded to the Current Register Set register.

3. DMA transaction

A DMA transaction is executed based on the values that are set. For details about transfer, section 33.9.13, DMA Transfer.

4. The USBFDMAmn masking

The USBFDMAmn interrupt is masked depending on the value of the DEM bit in the CHCFG_n register. If DEM = 1, the USBFDMAmn interrupt is masked, and the DEM bit is automatically cleared to 0.

5. DMATC masking

DMATC from DMAC control to USB control is masked depending on the value of the TCM bit in the CHCFG_n register. If TCM = 1, DMATC is masked, and the TCM bit is automatically cleared to 0.

6. Automatic register set switchover

Whether the current Next register set is to be switched to the other Next register set is determined by the value of the RSW bit in the CHCFG_n register.

7. Continuation of execution

Whether to continue DMA transfer is determined by the value of the REN bit in the CHCFG_n register. If REN = 0, the EN and TACT bits in the CHSTAT_n register are cleared to 0, and DMAC operation stops. If REN = 1, DMAC operation continues, and the REN bit is automatically cleared to 0.

(b) Register mode setting

- Register mode setting

The register set to be processed is selected.

Table 33.37 Register Mode Setting

DMS (CHCFG_n)	RSEL (CHCFG_n)	Description
0	0	Processes Next0 Register Set.
	1	Processes Next1 Register Set.

- USBFDMAmn masking

The USBFDMAmn interrupt can be masked.

Table 33.38 USBFDMAmn Mask Setting

DEM (CHCFG_n)	Description
0	Asserts the USBFDMAmn interrupt when the DMA transaction is completed.
1	Does not assert the USBFDMAmn interrupt even when the DMA transaction is completed. After the DMA transaction is completed, the DEM bit is cleared to 0.

- DMATC mask setting

DMATC from DMAC control to USB control can be masked.

Table 33.39 DMATC Mask Setting

DEM (CHCFG_n)	Description
0	Asserts DMATC when the DMA transaction is completed.
1	Does not assert DMATC even when the DMA transaction is completed. After the DMA transaction is completed, the TCM bit is cleared to 0.

- Automatic transaction execution for a register set

After a DMA transaction finishes, another DMA transaction can be executed.

Table 33.40 Automatic Execution Setting for a Register Set

REN (CHCFG_n)	Behavior	Remarks
0	The EN bit is cleared and DMA operation is terminated when the DMA transaction for the register set that is set by RSEL finishes.	Use this setting to execute a DMA transaction only once.
1	After a DMA transaction finishes, DMA transfer of the contents of the next register set continues. The REN bit is cleared to 0 when continuous transfer is successful.	Use this setting to continue processing of register set contents.

- Automatic register set switchover setting

After a DMA transaction finishes, the next register set to be processed can be switched.

Table 33.41 Automatic Execution Setting for a Register Set

RSW (CHCFG_n)	Behavior	Remarks
0	The register set is not switched when a DMA transaction finishes.	Use this setting to use only one register set.
1	When REN = 1 and a DMA transaction finishes, the RSEL setting is automatically reversed to select the other register set.	Use this setting to switch the register set.

(c) Example of setting the register mode

- Example of setting the register mode when using only the Next0 register set

Table 33.42 Register Mode Setting Example

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	TCM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0 (register mode)	0 (Next0)	0 (not masked)	0 (not masked)	0 (not switched)	0 (continuous execution disabled)

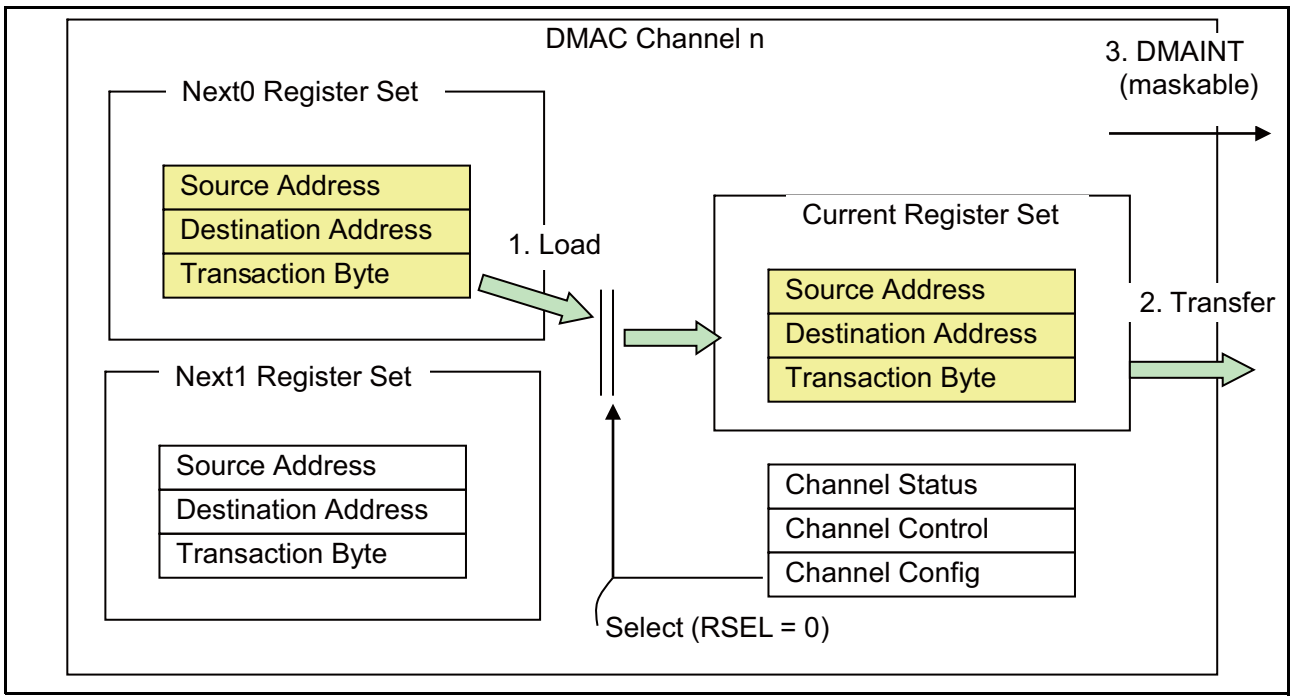


Figure 33.16 Register Mode Setting Example 1

1. By writing 1 to the SETEN bit in the CHCTRL_n register, the EN bit in the CHSTAT_n register is set to 1. As a result, the contents of Next0 Register Set are loaded to Current Register Set.
2. A DMA transaction is executed based on the values of Current Register Set and Channel Register Set.
3. Because the DEM bit in the CHCFG_n register is 0, the USBFDMAmn interrupt is asserted after the DMA transaction finishes.
4. Because the TCM bit in the CHCFG_n register is 0, DMATC is asserted after the DMA transaction finishes.
5. Because the REN bit in the CHCFG_n register is 0, the EN bit in the CHSTAT_n register is cleared to 0, and the operation ends.

- Example of setting the register mode when using two register sets continuously

Table 33.43 Automatic Register Set processing Setting

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	TCM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0 (register mode)	0 (Next0)	1 (masked)	0 (not masked)	1 (switched)	1 (continuous execution enabled)

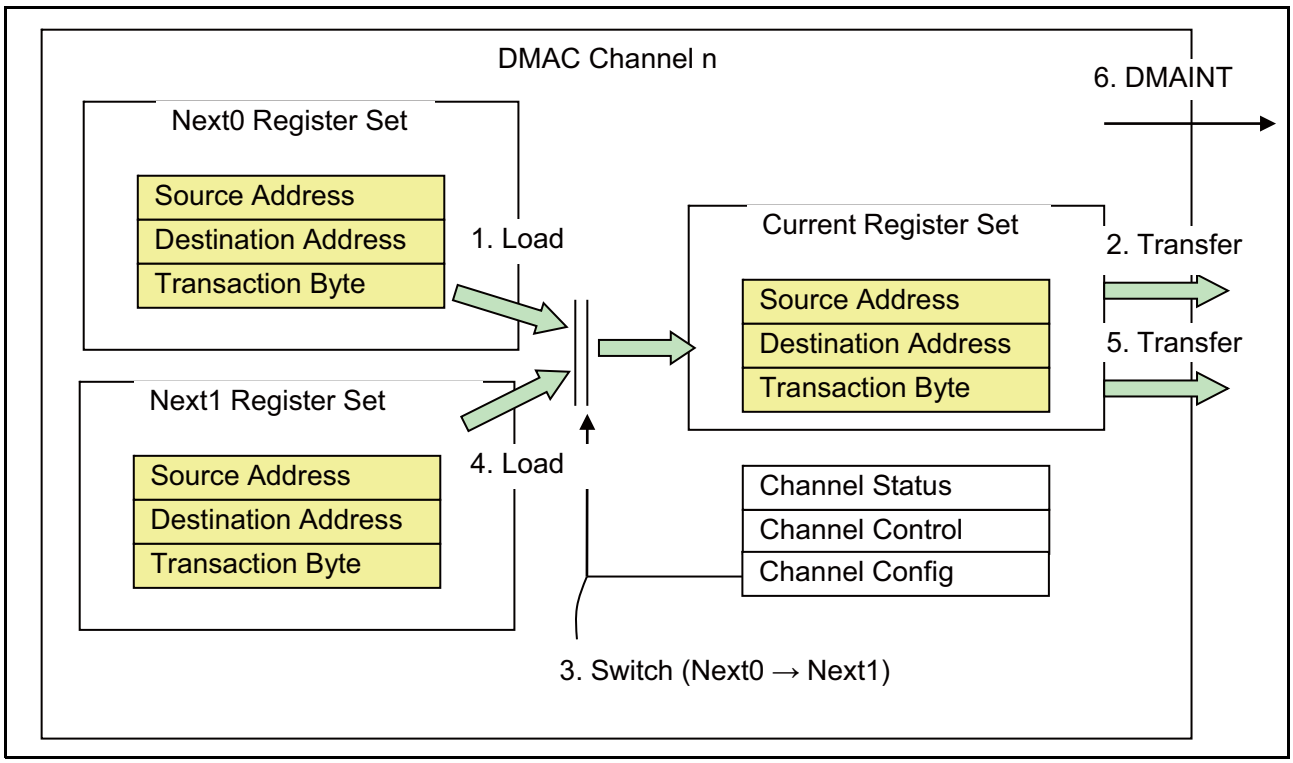


Figure 33.17 Register Mode Setting Example 2

1. By writing 1 to the SETEN bit in the CHCTRL_n register, the EN bit in the CHSTAT_n register is set to 1. As a result, the contents of Next0 Register Set are loaded to Current Register Set.
2. A DMA transaction is executed based on the values of Current Register Set and Channel Register Set.
3. Because the DEM bit in the CHCFG_n register is 1, the USBFDMAmn interrupt is not asserted after the DMA transaction finishes.
4. Because the REN bit in the CHCFG_n register is 1, operation continues. The REN bit is automatically cleared to 0.
5. Because the RSW bit in the CHCFG_n register is 1, the next register set to be processed is switched (RSEL = 0 to 1).
6. The contents of Next1 Register Set are loaded to Current Register Set.
7. A DMA transaction is executed based on the values of Current Register Set and Channel Register Set.
8. Because the DEM bit in the CHCFG_n register is 0, the USBFDMAmn interrupt is asserted after the DMA transaction finishes.
9. Because the TCM bit in the CHCFG_n register is 0, DMATC is asserted after the DMA transaction finishes.
10. Because the REN bit in the CHCFG_n register is 0, the EN bit in the CHSTAT_n register is cleared to 0, and the operation ends.

(2) Link mode

In link mode, this module reads the value set in a descriptor placed in an external storage area to execute a DMA transaction. In DMAC, there are Next Link Address (NXLA_n) and Current Link Address (CRLA_n) registers for each channel. The Next Link Address (NXLA_n) register is used to set the address of the descriptor to be read the next time. The Current Link Address (CRLA_n) register is used to display the descriptor address for the current DMA transaction.

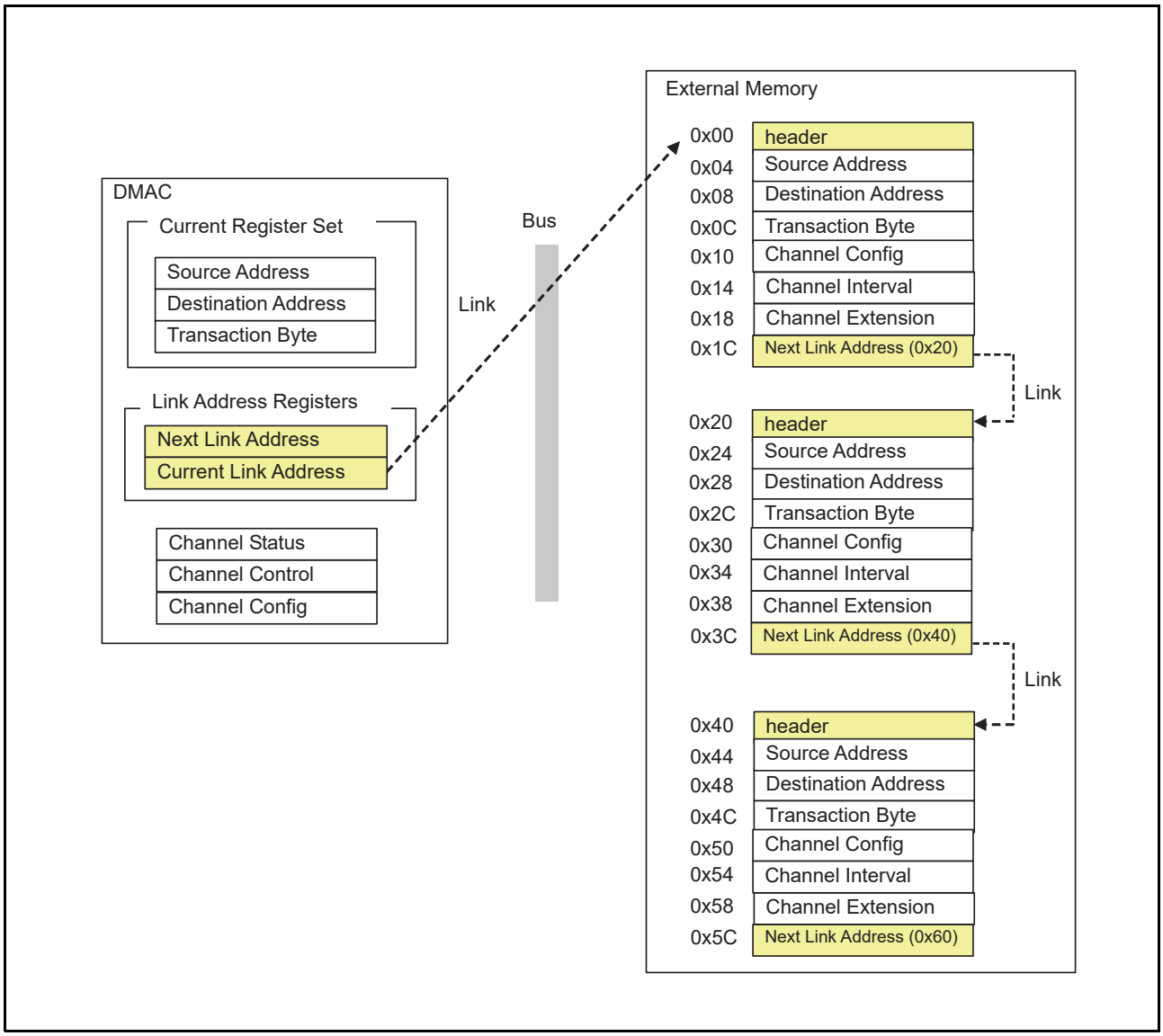


Figure 33.18 Overview of Link Mode

(a) Operation flow of link mode

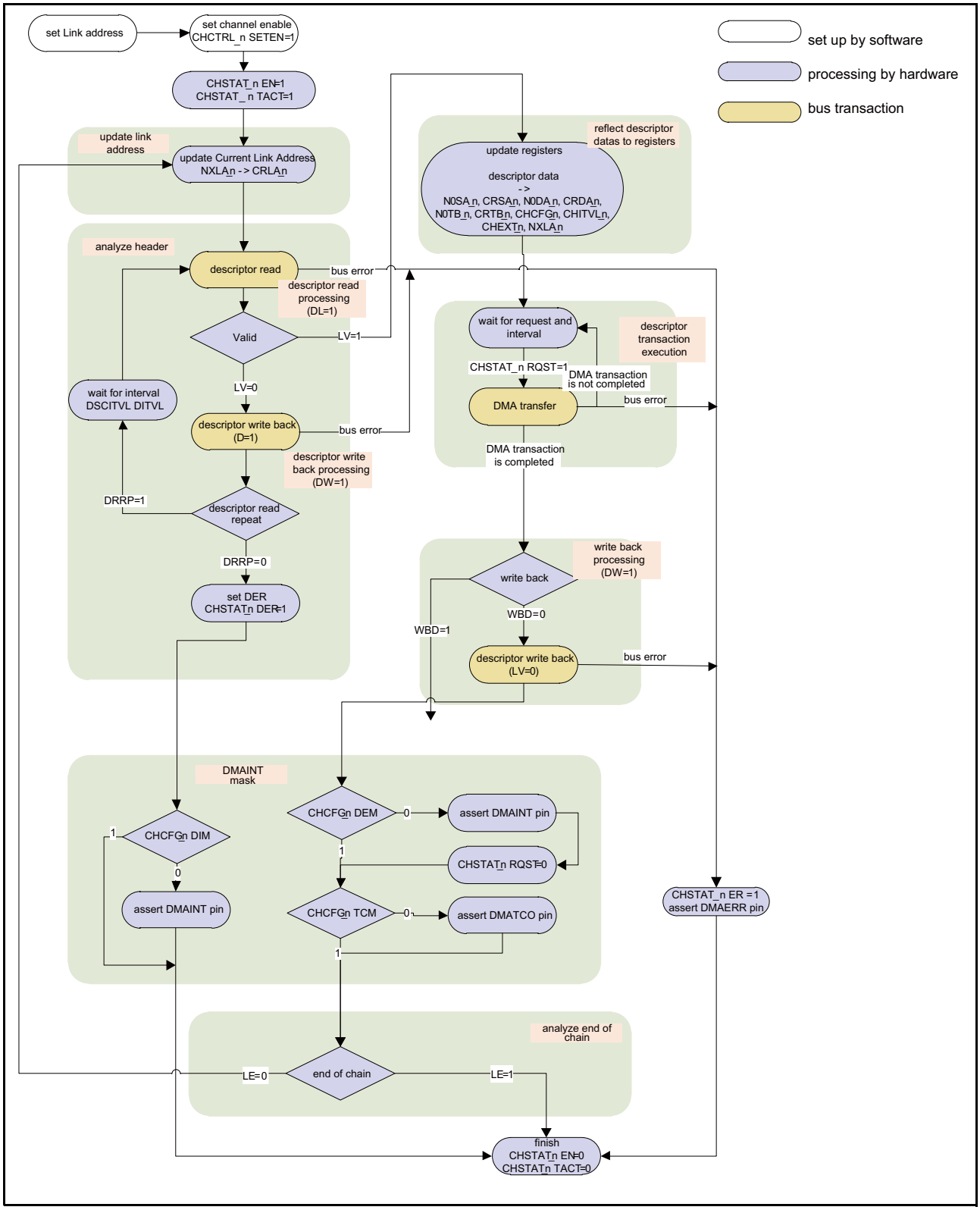


Figure 33.19 Link Mode Operation Flow

Description of the link mode operation flow:

1. Channel setting
   The beginning address of the link destination is set in the NXLA_n register.

2. Link address updating
   If 1 is written to the SETEN bit in the CHCTRL_n register, the EN and TACT bits of the CHSTAT_n register are set to 1. As a result, the address set in the NXLA_n register is loaded to the CRLA_n register.

3. Descriptor reading and header judgment
   A read of the descriptor starts, and DMAC checks the contents of header. If LV = 0, this module writes 1 back to the D bit of header. After that, if the DRRP bit in the CHCFG_n register is 1, this module waits for the time intervals set by the DSCITVL register, and then reads the same descriptor again. If DRRP = 0, the DER bit in the CHSTAT_n register is set to 1, and this module is placed in the end state (both EN and TACT bits in the CHSTAT_n register are 0). At this time, if the DIM bit in the CHCFG_n register is 0, this module asserts the USBFDMAmn interrupt.

4. Descriptor setting
   If LV = 1, the data read from the descriptor is loaded to Current Register Set and Channel Register Set. In addition, the next link target is loaded to the NXLA_n register.

5. DMA transaction
   A DMA transaction is executed based on the values that are set. For details about transfer, see section 33.9.13, DMA Transfer.

6. Header write-back
   If WBD of header is 0, DMAC writes LV = 0 to the header area.

7. USBFDMAmn masking
   If the DEM bit in the CHCFG_n register is 0, this module asserts the USBFDMAmn interrupt.

8. DMATC masking
   If the TCM bit in the CHCFG_n register is 0, this module asserts DMATC.

9. Link end judgment
   If LE of header is 1, the EN and TACT bits in the CHSTAT_n register are cleared to 0, and DMAC terminates operation. If LE is 0, this module updates Current Register Set, and then restarts reading the next descriptor.

(b) Register setting

- Link mode setting

To use link mode, set the DMS bit in the CHCFG_n register to 1.

Table 33.44 Link Mode Setting

DMS (CHCFG_n)	Description
1	This module operates in link mode. The setting of this bit cannot be changed by using a descriptor.

- Link address setting

The Next Link Address (NXLA_n) and Current Link Address (CRLA_n) registers are used to indicate a link target.
Before starting link mode, set the link target in the NXLA_n register.
After reading a descriptor, this module updates the NXLA_n register to the next link. Note that the CRLA_n register indicates the address of the link target that is being executed.

Table 33.45 Link Address Register Set

Register	Description
Next Link Address Register (NXLA_n)	This register is used to set and display the next link target. Before starting link mode, set the address of the link target in this register.
Current Link Address Register (CRLA_n)	This register is used to display the link target that is being executed. This is a read-only register.

(c) Descriptor setting

DMAC supports multiple descriptor formats.
A switchover between formats is specified by using the DSCFM field of bits [31:28] of the 1st word (header) of the descriptor.
The following table shows the relationship between DSCFM values and descriptor formats.

Table 33.46 Descriptor Formats

DSCFM	Descriptor Size	Next Link Address	Channel Extension	Channel Interval	Channel Config	Transaction Size	Destination Address	Source Address	header
3	4 words	√	— (reload)	— (reload)	— (reload)	— (header)	√	√	√ (with STS)
1	8 words	√	√	√	√	√	√	√	√ (no STS)
Other than the above	If DSCFM is set to a value that is not 1 or 3, operation cannot be guaranteed. Make sure that DSCFM is set to 1 or 3.								

Table 33.47 Explanation of the Marks in Table 33.46

Field	Mark	Description	Remarks
Header	√ (with STS)	The STS field of bits [15:0] in the header is valid. The value set in the STS field is used as the total number of transfer bytes (Transaction Size).	—
	√ (no STS)	The STS field of bits [15:0] in the header is invalid. The value of "Transaction Size" in the descriptor is used as the total number of bytes.	
Source Address	√	Specify the source address.	—
Destination Address	√	Specify the destination address.	—
Transaction Size	√	Specify the transaction size.	—
	— (header)	Omit the transaction size. The value set in the STS field is used as the total number of transfer bytes (Transaction Size)	Because the STS field is of 16 bits, a maximum of 65,535 bytes can be set.
Channel Config Channel Interval Channel Extension	√	Specify Channel Config, Channel Interval, and Channel Extension.	—
	— (reload)	Omit Channel Config, Channel Interval, and Channel Extension. The previous settings (the values of the CHCFG_n, CHITVL_n, and CHEXT_n registers of the last time) are inherited.	—
Next Link Address	√	Specify the next descriptor address (Next Link Address) to be read after DMA transfer of the descriptor.	—

DMAC sequentially interprets data read from descriptors. If the number of words specified for DSCFM is less than 8, place the data of descriptors that are indicated by " √ " in Table 33.46 on memory.
No software settings are required in OUT transfer (reception in peripheral mode) or in IN transfer (transmission in peripheral mode) because the hardware sets the transfer source or destination address according to the setting of the CURPIPE bits in the DxFIFOSEL register.

Table 33.48 Example of Placing Descriptors

	Address							
DSCFM	Link Address + 1CH	Link Address + 18H	Link Address + 14H	Link Address + 10H	Link Address + 0CH	Link Address + 08H	Link Address + 04H	Link Address + 00H
3H	—	—	—	—	Next Link Address	Destination Address	Source Address	header
1H	Next Link Address	Extension	Interval	Config	Transaction Byte	Destination Address	Source Address	header

- header

The header area provides the descriptor status and other information as shown below.

DMAC reads this area before DMA transfer in link mode starts. After a DMA transaction terminates, DMAC writes the transfer status back to this area.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DSCFM				—	WBD	LE	LV	D	—						
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
If the WBD bit is 0 when DMAC reads the header, DMAC writes data back to this 1-byte area after the DMA transaction in accordance with the descriptor ends. At this time, DMAC writes 0 back to the LV bit. For other bits, DMAC writes the values read from the header.								If the LV bit in the header is 0 when DMAC reads the header, DMAC writes 80H back to this 1-byte area.							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STS															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 33.20 Header Area

Table 33.49 Header Area

Bit position	Bit name	Meaning
31 to 28	DSCFM	Descriptor Format Specifies the descriptor format (length and combination of descriptors). For details see Table 33.46
27	—	Reserved area. Set 0.
26	WBD	Wrazite Back Disable Masks a write-back operation for the LV bit. If this bit is 1, DMAC does not perform a write-back operation. 0: Writes 0 back to the LV bit. 1: Does not perform a write-back operation for the LV bit.
25	LE	Link End Indicates that the link will end with the DMA transaction for this descriptor. Set this bit to 1 to indicate the end of link. 0: The link continues. 1: The link ends.
24	LV	Link Valid Indicates that this descriptor is valid. If WBD = 0, after DMAC executes the DMA transaction written in the descriptor, DMAC writes 0 to this bit. When header is set, set 1 to this bit. 0: This descriptor is invalid. 1: This descriptor is enable.
23	D	Descriptor Error Indicates a descriptor access error. If LV is 0 when the descriptor is read, DMAC writes 1 back to this bit. 0: A descriptor error has not occurred. 1: LV was 0 when the descriptor was read.
22 to 16	—	Reserved area. Set 0.
15 to 0	STS	Short Transaction Size If DSCFM is 3, the transaction size is set (in bytes). The maximum number of transfer bytes that can be set is 65,535. If DSCFM is 3, Do not set 0 for STS. If 0 is set, operation cannot be guaranteed.

If descriptors are added sequentially while DMAC is operating, the access that the CPU sets 0 to the LV bit and the access that DMAC writes 1 back to the D bit might contend with each other. If this contention occurs, prior-written data is overwritten by latter-written data.

To prevent this problem from occurring, DMAC performs a write-back operation for the D bit in a byte-write manner. Therefore, the CPU must also set LV to 1 in a byte-write manner. Because the byte lanes for the D and LV bits are

different, by writing data to different areas, occurrence of this problem can be prevented.

- Settings of descriptors other than header

The specifications of data of the descriptors other than header are the same as the specifications of internal registers.

- Settings specified when descriptors are accessed

The MHPROT pin output can be set for the LWPR and LDPR fields of the DCTRL register when descriptors are accessed. Set it according to the access target in which descriptors are deployed.

- Descriptor areas and DMA transfer areas

The following provides an overview of the descriptor areas and DMA transfer areas that are accessed by DMAC.

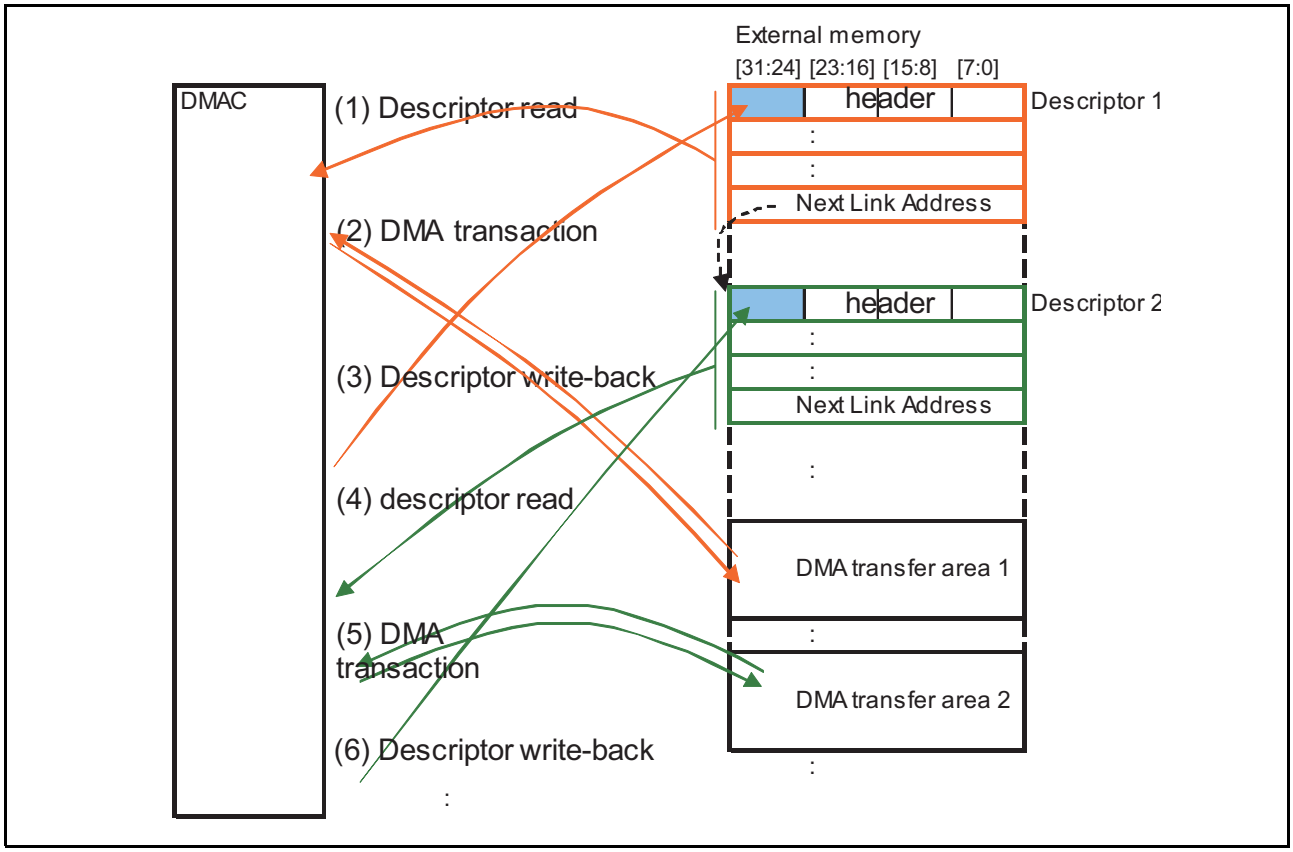


Figure 33.21 Overview of Descriptor Areas and DMA Transfer Areas

(1) Descriptor read
DMAC loads a value from the internal NXLA_n register to the CRLA_n register, and then reads a descriptor from the external memory space indicated by the CRLA_n register (descriptor 1).

(2) DMA transfer
If the LV bit of the header descriptor is 1, DMAC performs a DMA transfer in accordance with the descriptor information.

(3) Descriptor write-back
After performing a DMA transfer of data by the number of bytes that are set, if the WBD bit in header is 0, DMAC performs write-back for bits [31:24] in header of Descriptor 1. For the LV field, 0 is written back. For other fields, the values read in (1) are written back on a byte-size basis.

(4) Descriptor read
If the value of the LE bit in the header descriptor that was read previously (in (1)) is 0, DMAC reads the next descriptor from the address (descriptor2) indicated by Next Link Address in the current descriptor.

(5) DMA transfer
If the LV bit in the header descriptor is 1, DMAC performs a DMA transfer in accordance with the descriptor information.

(6) Descriptor write-back
After performing a DMA transfer of data by the number of bytes that are set, if the WBD bit in header is 0, DMAC performs write-back for bits [31:24] in header of Descriptor 2. For the LV field, 0 is written back. For other fields, the values read in (4) are written back on a byte-size basis.

(Steps (4) to (6) are repeated.)

If LE = 1 and WBD = 0 in header, DMAC performs a DMA transfer with the descriptor settings, writes 0 back to the LV bit in header, and then terminates processing.
If LE = 1 and WBD = 1 in header, DMAC performs a DMA transfer with the descriptor settings, and then terminates processing (without performing write-back).
If LV = 0 in header, DMAC writes 1 back to the D bit in header, and then, if the DRRP bit in the CHCFG_n register is 1, DMAC waits for the number of intervals specified by the DITVL field of the DSCITVL_n register, and then reads the descriptor again. If DRRP = 0, DMAC terminates processing.

- Notes on descriptors

  - In link mode, settings can be changed by reading descriptors. However, it is impossible to synchronize the setting change times and hardware requests. Therefore, to use hardware requests, before setting the SETEN bit in the CHCTRL_n register, set the AM, LVL, HIEN, LOEN, and SEL bits in the CHCFG_n register. Note that these setting bits must not be changed in descriptors.
  - Descriptors cannot be used to change the DMS field in the CHCFG_n register (DMAC is always placed in link mode). Although descriptors can be used to change the settings of the REN, RSW, and RSEL fields in the CHCFG_n register, changes of those fields do not affect operation.
  - The descriptor can be initialized by overwriting the memory area corresponding to the descriptor you intend to initialize while the DMAC is not operating. The DMAC determines whether or not the descriptor is valid by referring to the DSCFM field and LV bit in the header. Accordingly, set the areas in memory corresponding to the DSCFM field and LV bit to 1 or 3, and to 1, respectively, before enabling DMAC operation.
  - To set the next descriptor on memory while DMAC is operating, make sure that 1 is written to the LV bit after the descriptors subsequent to header (Source Address, Destination Address, ..., Next Link Address) are set. If this is not performed and descriptor setting by the CPU and descriptor reading by DMAC contend, DMAC performs a DMA transfer using the previous values of those descriptors (Source Address, Destination Address, ..., Next Link Address).
  - To leave the write-back information for the D bit of header, make sure that 1 is written to the LV bit of header in a byte-access manner.

(d) Link configuration example

In link mode, descriptors can be configured as shown below.

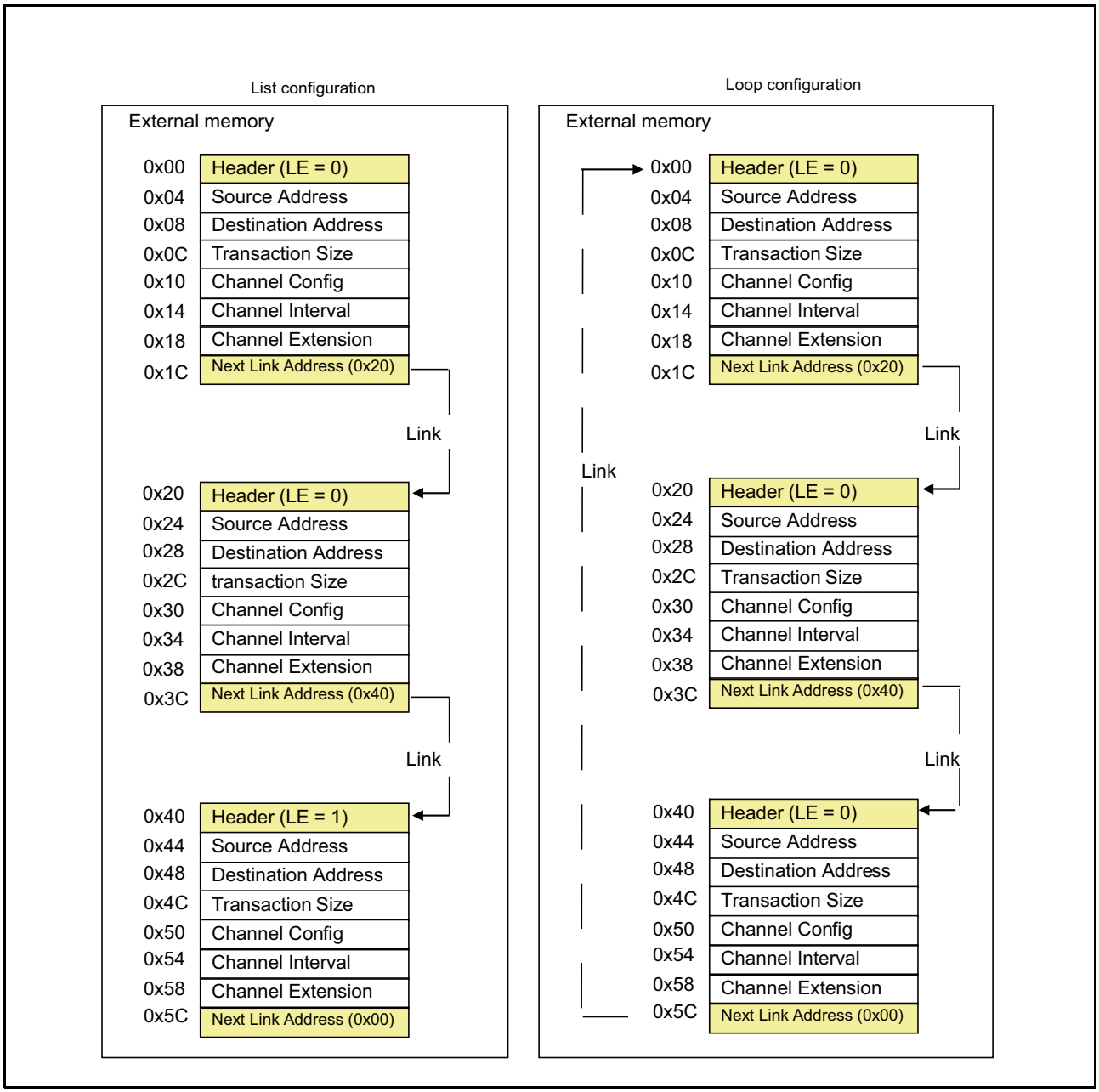


Figure 33.22 Link Mode Configuration Example

- List configuration

The link ends by setting 1 for the LE bit in the header of the last descriptor.

- Loop configuration

A loop of descriptors can be created by setting the link target of the last descriptor to the address of the first descriptor. To end the loop, change the value of the LE bit of header to 1 or use the transfer interrupt procedure.

33.9.12.2 Write only mode

Write-only mode is enabled by setting 1 for the WONLY bit in the CHCFG_n register.

Table 33.50 Write-Only Mode Setting

WONLY (CHCFG)	Mode	Description
0	Normal mode	A DMA transfer is performed using the values set in Next Register Set.
1	Write-only mode	A DMA write transfer is performed without performing a DMA read transfer.

In write-only mode, no DMA read transfer is performed (note that descriptors are read in the same way as in normal mode). In register mode, the values set in the NxSA_n register (if RSEL = 0, x = 0; if RSEL = 1, x = 1) are used as write data. In link mode, the values of the SA fields of descriptors are used as write data.
Use this mode to, for example, initialize the memory area.

33.9.13 DMA Transfer

This chapter describes the basic operation of DMA transfer.

33.9.13.1 Transfer modes

DMAC supports only single transfer mode.
Upon receiving a DMA transfer request from USB control, DMAC executes a single DMA transfer on the side (source or destination) indicated by the REQD bit in the CHCFG_n register. DMAC then asserts internal DMA permission from internal USB control to DMAC control. DMAC performs a single transfer each time a transfer is received. DMAC continues this operation by the transfer size loaded to the CRTB_n register (arbitration between channels is performed for each DMA transfer).
The timing of internal DMA permission from internal USB control to DMAC control differs depending on the setting of the REQD bit in the CHCFG_n register and the setting of the transfer size (DDS[2:0] and SDS[2:0] in the CHCFG_n register). For details, see 33.9.13.7, Operational difference depending on the transfer size.

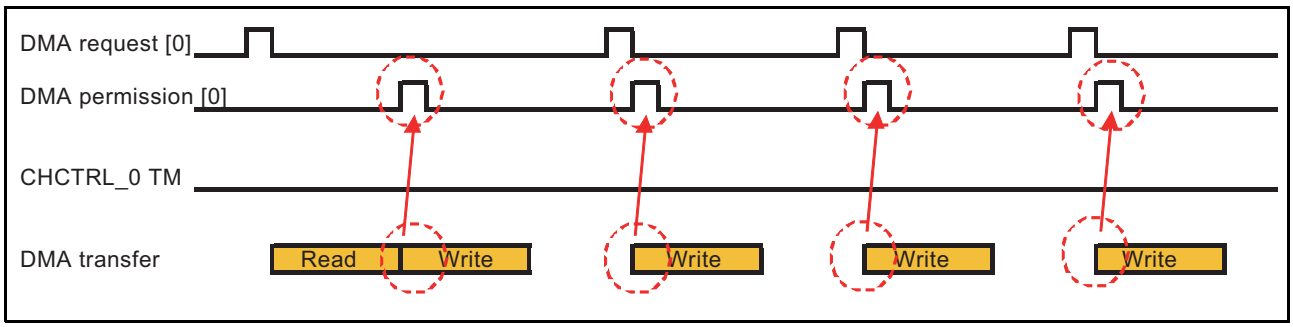


Figure 33.23 Single Transfer Mode (REQD = 1, SDS > DDS)

33.9.13.2 DMA channel priority control

DMAC supports fixed-priority mode and round-robin mode as methods of arbitration between channels. The mode is selected by using the PR bit in the DCTRL register. If the PR bit is 0, fixed-priority mode is selected. If the PR bit is 1, round-robin mode is selected.

Table 33.51 Priority Control Setting

Mode	PR (DCTRL)	Description	Remarks
Fixed-priority	0	Controls requests based on fixed priority (CH0 > CH1 > ...).	Use this mode if channels have priority.
Round-robin	1	Controls requests in a round-robin manner.	Use this mode to execute requests equally.

(1) Fixed-priority mode

In fixed-priority mode, a fixed priority level is assigned to each channel as shown below.

(High) CH0 > CH1 (Low)

If DMA transfer requests simultaneously occur over multiple channels, the DMA transfer over the channel whose number is smallest is performed first.

The transfer over channel 0 is performed first. However, while a transfer switches to another transfer over channel 0, a transfer over the channel with the next highest priority level is performed in order to increase the bus usage rate.

(2) Round-Robin Mode

In round-robin mode, each time a transfer over a channel is received, the priority level of the channel that was used for the previous transfer is changed to the lowest level.
In the status immediately after the mode is reset, channels are assigned priority levels in the same way as fixed-priority mode as shown below.

(High) CH0 > CH1 (Low)

In this status, if a transfer request for DMA channel 0 does not occur and a transfer request for DMA channel 1 occurs, the transfer over DMA channel 1 is performed. When the transfer finishes, the channel priority is changed as follows.

(High) CH1 > CH0 (Low)

The following shows an example of DMA transfer in round-robin mode.

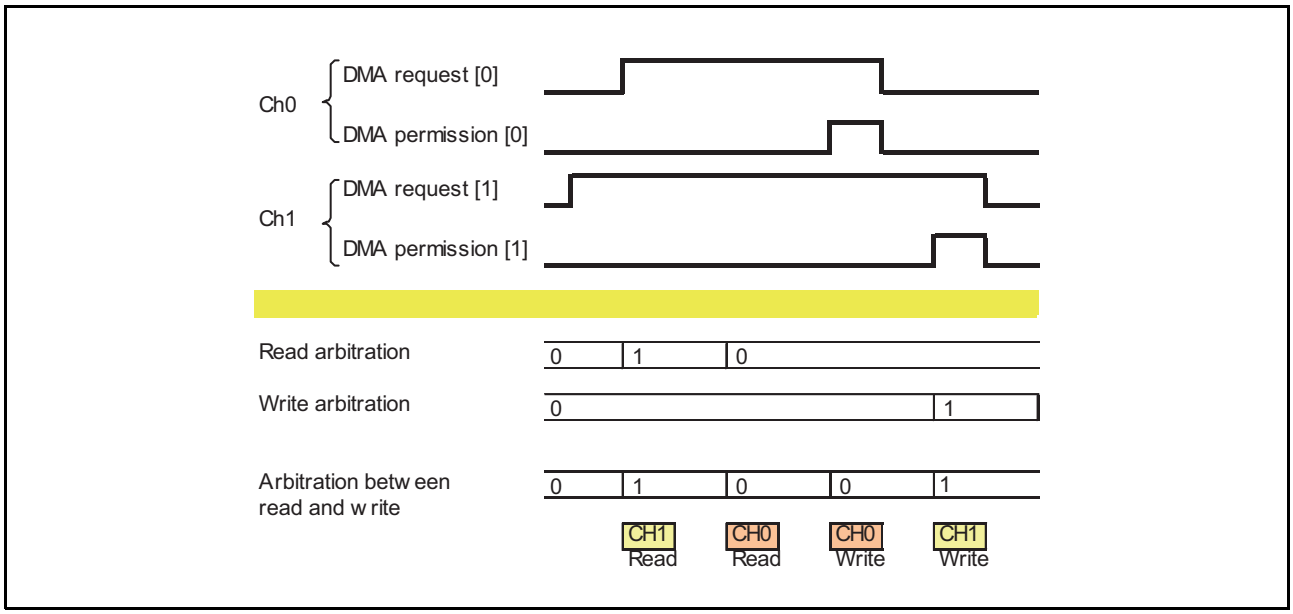


Figure 33.24 Example of Operation in Round-Robin Mode (with 4 channels, REQD = 1)

DMAC internally performs arbitration between read channels and arbitration between write channels, further performs arbitration between the arbitration results, and then issues bus access.

33.9.13.3 Forced sweeping request

When a forced sweeping request is entered, DMAC transfers the data that is left untransferred in the buffer to the destination address. After the sweep processing finishes, DMAC continues DMA transfer.

The following provides notes on forced sweeping requests:

- If a forced sweeping request and a transfer request from USB control contend, DMAC performs forced sweeping first, and then performs a DMA transfer.
- If the USB-control-side system is the destination (REQD bit in the CHCFG_n register is 1) register REQD = 1, buffer overflow or another error might occur on the destination unit because data is transferred although no DMA transfer request is made on the USB control side. Therefore, the specifications physically prohibit DMAC from using forced sweeping if REQD = 1.
- The difference from ordinary sweeping mode described in section 33.9.13.8 (3) (b), Transfer suspension (buffer sweeping enabled: SBE = 1) (EN is cleared by setting 1 for the SBE bit in the CHCFG_n register) is as follows: DMAC stops operation after writing data in the buffer in ordinary sweeping mode, whereas DMAC can continue DMA transfer after sweeping the buffer in forced sweeping mode.

(1) Forced software sweeping request

The SETSSWPRQ bit in the CHCTRL_n register determines whether software-based forced sweeping requests can be used. To perform a forced sweeping request, write 1 to the SETSSWPRQ bit. DMAC then outputs the data in the buffer to the destination.

33.9.13.4 DMA transfer completion interrupt (USBFDMAmn)

USBFDMAmn (m, n = 0, 1) is an interrupt signal that indicates termination of a DMA transaction.

If a transfer for the total number of transfer bytes loaded to the CRTB_n register is completed by an OKAY response, the END bit in the CHSTAT_n register is set to 1. At this time, if the DEM bit in the CHCFG_n register is 0, DMAC generates a USBFDMAmn (m, n = 0, 1) interrupt.
(If a write-back operation is performed in link mode, the interrupt is generated after the write-back operation finishes.)

In link mode, when the DRRP bit in the CHCFG_n register is 0, if the LV bit of the header of the descriptor that is read is 0, the DER bit in the CHSTAT_n register is set to 1. At this time, if the DIM bit in the CHCFG_n register is 0, DMAC generates a USBFDMAmn interrupt.

Use this signal to detect a transfer completion interrupt performed by the interrupt controller.

Table 33.52 USBFDMAmn Assertion Conditions

Cause	Condition	INT_DMA[n] mask signal
DMA transaction ended	A transfer of data by the number of transfer bytes loaded to the CRTB_n register is completed by an OKAY response (if a write-back operation is performed in link mode, after the operation finishes)	DEM bit in the CHCFG_n register
Descriptor was invalid	LV of header of the descriptor that is read is 0 when DRRP and DIM in the CHCFG_n register are both 0 in link mode	DIM bit in the CHCFG_n register

33.9.13.5 DMA error interrupt (USBFDMAERRm)

If an error response is received for DMA transfer or descriptor access, this module stops transfer, assuming that an error occurred. When an error response is received, the EN bit in the CHSTAT_n register for channel n that is being used for transfer is cleared to 0, and the ER bit is set to 1 (n = 1, 0). Also, the USBFDMAERRm interrupt is asserted.

The USBFDMAERRm signal cannot be masked.

For a sequence of transfers for which an error occurred, data integrity cannot be guaranteed. Always use the following procedure to restart the transfer sequence from the beginning.

1. Set the SWRST bit in the CHCTRL_n register to 1.
2. Reset each register.

33.9.13.6 Interval Count Function

The execution interval of a DMA transfer can be adjusted by using the ITVL field in the CHITVL_n register. This function prevents DMAC from continuously occupying the bus. If this function is enabled, DMAC does not perform a DMA transfer for the next request until the counter value becomes 0.
The following shows an operation example.

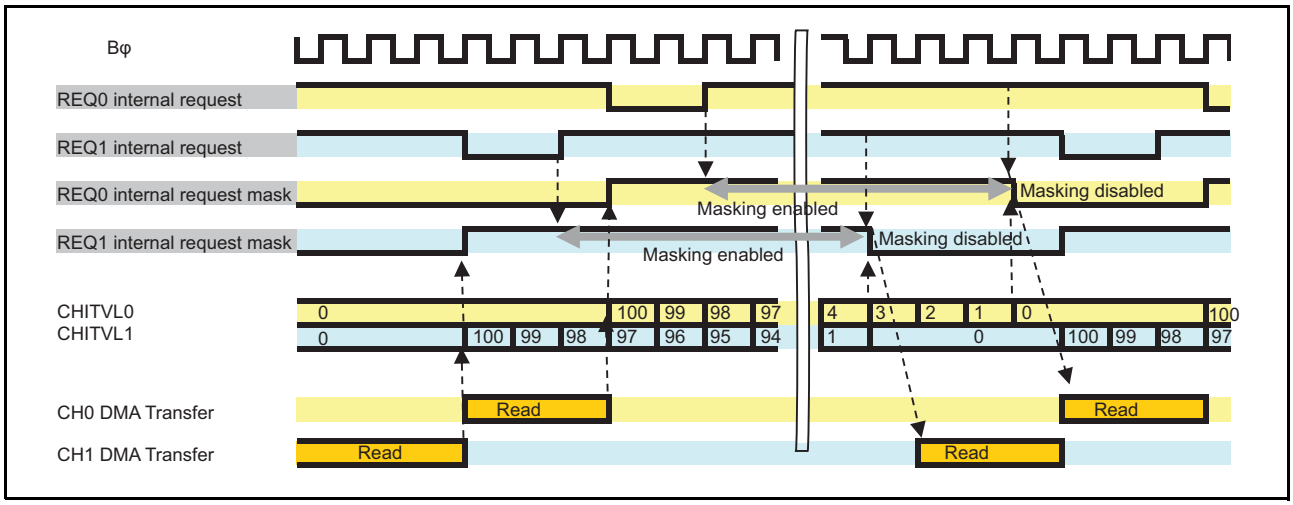


Figure 33.25 Example of Counting Intervals (REQD = 0, SDS < DDS)

An interval is inserted after a transfer is performed on the side specified by the REQD bit in the CHCFG_n register. The following shows how the REQD, SDS, and DDS values of the CHCFG_n register are related with the interval.

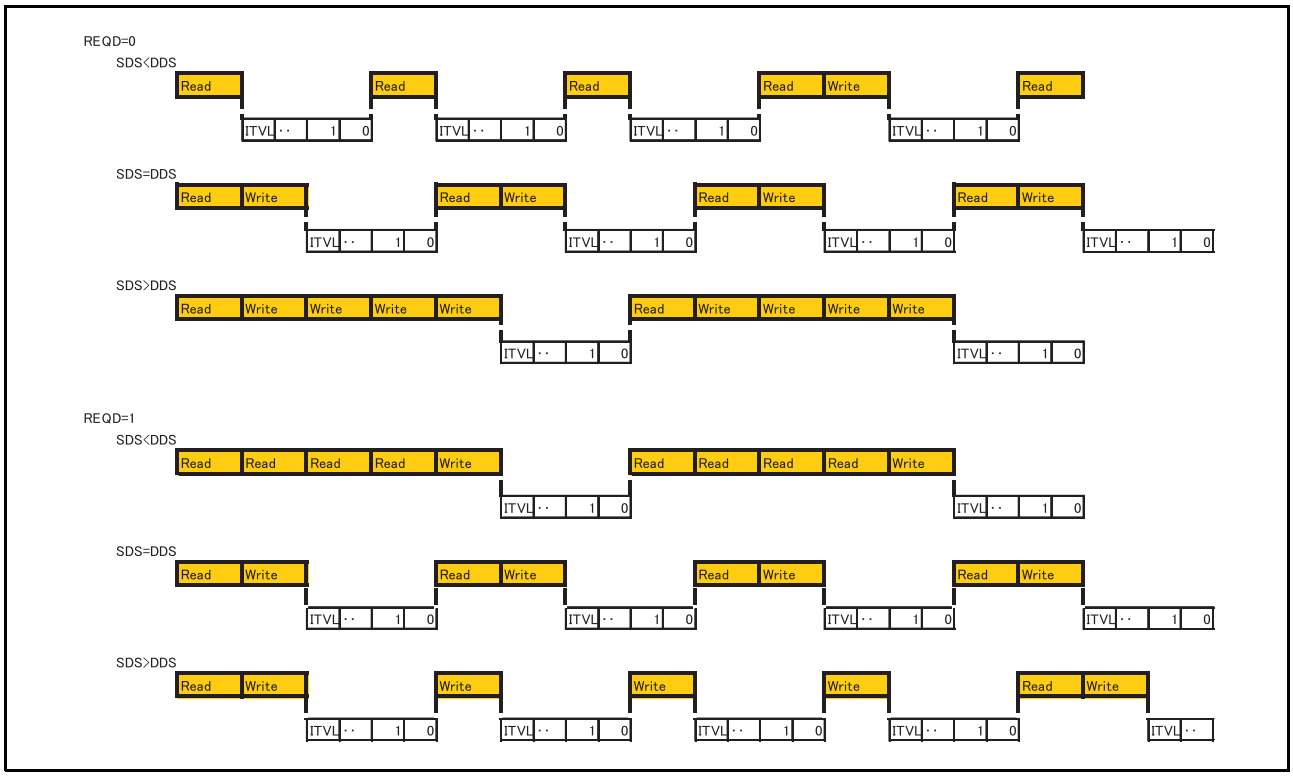


Figure 33.26 DMA Transfer Settings and Interval Count

33.9.13.7 Operational difference depending on the transfer size

(1) If the transfer size on the source side is smaller

When reading of as much data as the destination data size finishes, a write to the destination starts.
The following figure is an example of the timing chart in the case where the source is an 8-bit field and the destination is a 32-bit field (SDS = 0 and DDS = 2 in the CHCFG_n register) (when the rising edge is detected).

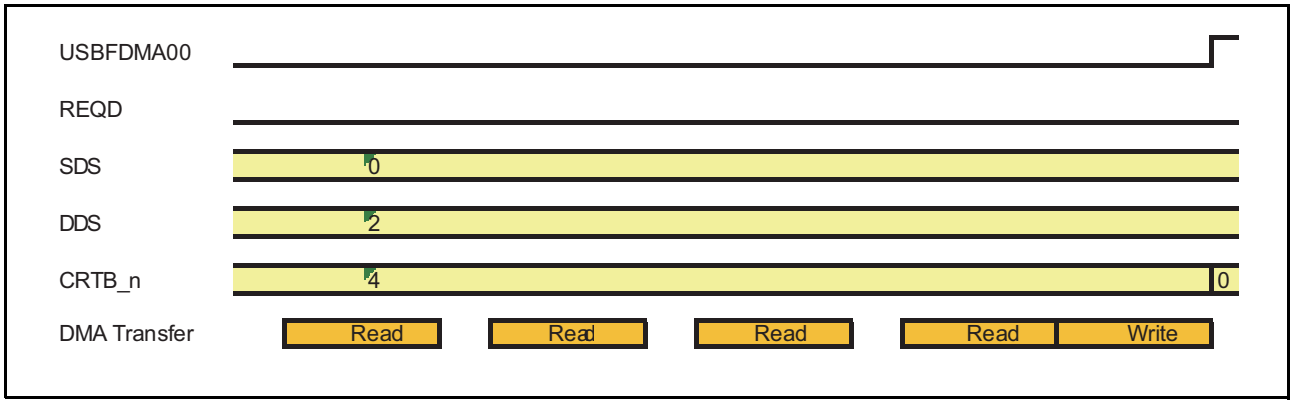


Figure 33.27 Example of Timing Chart in the Case Where the Source is Smaller
(LVL = 0, HIEN = 1, REQD = 0, and SDS < DDS)

(2) If the transfer size on the destination side is smaller

Because the source side is larger than the destination side, two or more destination write operations occur for one source read operation. The following is an example of the timing chart in the case where the source is a 64-bit field and the destination is a 16-bit field (SDS = 3 and DDS = 1 in the CHCFG_n register) (when the rising edge is detected).

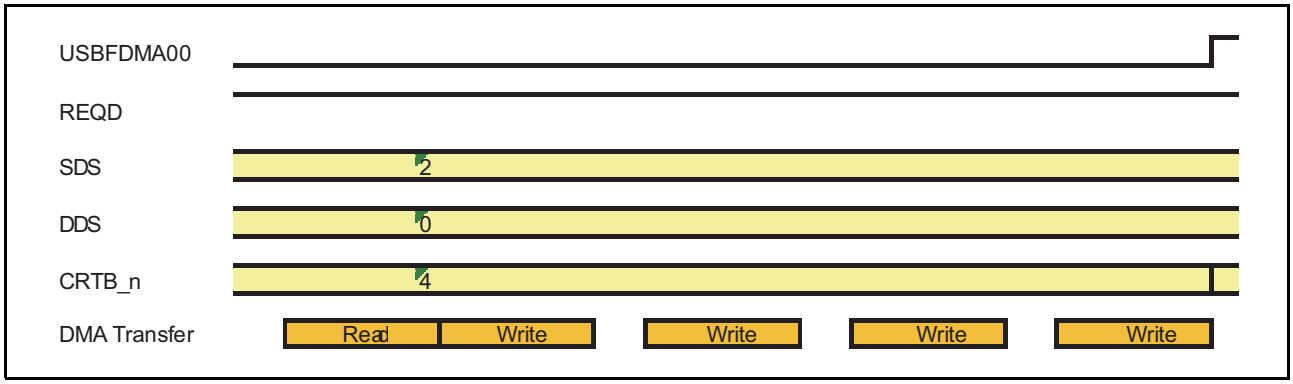


Figure 33.28 Example of Timing Chart in the Case Where the Destination Is Smaller
(LVL = 0, HIEN = 1, REQD = 1, and SDS > DDS in the CHCFG_n Register)

(3) If the Source and Destination Transfer Sizes Are the Same

Each time a DMA transfer request is detected, a source read operation and a destination write operation occur.
The following is an example of the timing chart in the case where the source and destination are 8-bit fields (SDS = 0 and DDS = 0 in the CHCFG_n register) (when the rising edge is detected).

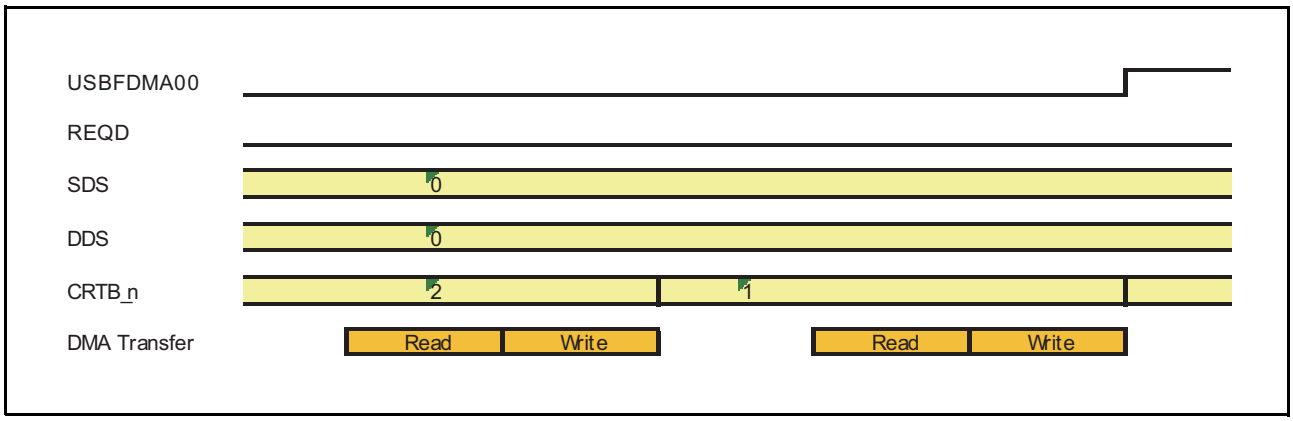


Figure 33.29 Example of Timing Chart in the Case where the Source and Destination Sizes are the Same (LVL = 0, HIEN = 1, REQD = 0, and SDS = DDS in the CHCFG_n Register)

33.9.13.8 Transfer state

The CHSTAT_n register indicates the transfer state of a channel.

(1) Transfer state

The TACT bit in the CHSTAT_n register indicates that channel n is operating. When 1 is written to the SETEN bit in the CHCTRL_n register, the TACT bit is set to 1. The TACT bit continues to be 1 while DMAC is accessing a descriptor or waiting for a DMA request.
The TACT bit is cleared when the EN bit in the CHSTAT_n register is cleared (for details about the conditions in which the EN bit is cleared, see 33.4.1, Current Source Address Register) and the DMA transfer ends.
If the EN bit is not cleared when the DMA transaction ends (for example, when the REN bit of the CHCFG_n register is 1 in register mode or when DMAC accesses the next descriptor in link mode), the TACT bit is not cleared.

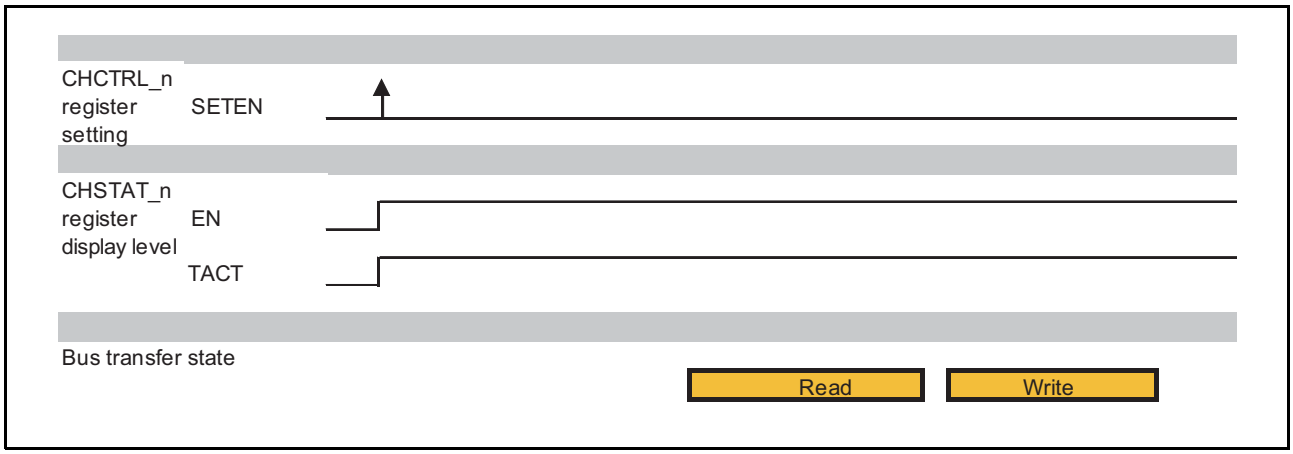


Figure 33.30 DMAC State Example 1 (Hardware Request)

(2) Suspension

The SETSUS bit in the CHCTRL_n register can be used to suspend a DMA transfer. If suspension of a DMA transfer is attempted when a bus cycle is running, the DMA transfer is suspended after the bus cycle finishes. The suspended transfer can be resumed by writing 1 to the CLRSUS bit in the CHCTRL_n register.

To check whether a DMA transfer is suspended, after setting the SETSUS bit in the CHCTRL_n register, check the SUS bit in the CHSTAT_n register or the SUS bit for the relevant channel in the DSTAT_SUS register. If The SUS bit is 1, the DMA transfer is currently suspended.

(3) Transfer suspension

By writing 1 to the CLREN bit in the CHCTRL_n register during a DMA transaction, the DMA transaction of the channel can be suspended. As the post-processing for suspension, the SBE bit in the CHCFG_n register can be used to select whether to sweep the data remaining in the buffer when the transaction is suspended. The default is SBE = 0 (do not sweep remaining data).
If the sweep mode is enabled and a transfer is suspended by setting 1 for the CLREN bit in the CHCTRL_n register, DMAC sweeps any data remaining in the buffer and stops operation.

(a) Transfer suspension (buffer sweeping disabled: SBE = 0)

If 1 is written to the CLREN bit in the CHCTRL_n register during DMA transfer, DMAC suspends DMA transfer and then stops. The timing of stoppage depends on the value set for the REQD bit. After DMAC stops, write 1 to the SWRST bit in the CHCTRL_n register to clear the internal status of DMAC, and then specify the next transfer settings.
Complete deactivation of the channel can be confirmed when the value of the TACT bit in the CHSTAT_n register changes from 1 to 0.
If DMA transfer is suspended before it is completed, the USBFDMAmn interrupt is not asserted.
If the REQD bit in the CHCFG_n register is 0, DMAC stops when the next read operation is completed. However, if data that can be written exists in the buffer, DMAC writes the data and then stops.
If the REQD bit in the CHCFG_n register is 1, DMAC stops when the next read operation is completed.

(b) Transfer suspension (buffer sweeping enabled: SBE = 1)

If 1 is written to the CLREN bit in the CHCTRL_n register during DMA transfer, DMAC suspends DMA transfer. If the REQD bit in the CHCFG_n register is 0, DMAC sweeps (writes) the already read data, and then stops DMA transfer. If the REQD bit is 1, sweep mode cannot be used physically.
After DMAC stops, set the SWRST bit in the CHCTRL_n register to clear the internal status of DMAC, and then specify the next transfer settings.
Complete deactivation of the channel can be confirmed when the value of the TACT bit in the CHSTAT_n register changes from 1 to 0.

(c) How to confirm deactivation of the channel

Even when the EN bit of the CHSTAT_n register is cleared to 0 by writing 1 to the CLREN bit in the CHCTRL_n register, if a transfer has already been executed over the bus, DMAC cannot immediately stop. To check whether DMAC has stopped completely, check the EN and TACT bits in the CHSTAT_n register. If both bits are 0, DMAC has stopped completely.

(d) Procedure for suspending transfer

To suspend transfer:

1. Write 1 to the CLREN bit in the CHCTRL_n register.
2. If the SBE bit in the CHCFG_n register is 0, DMAC stops according to the value of the REQD bit in the CHCFG_n register. If the SBE bit is 1, DMAC is placed in sweep mode.
3. Read the CHSTAT_n register to check whether the TACT bit is 0. If the TACT bit is 0, DMAC has stopped completely. If the TACT bit is 1, continue polling until the bit changes to 0.
4. To perform the next DMA transfer after it is suspended, make sure that the SWRST (software reset) bit in the CHCTRL_n register is turned on before the next DMA transfer starts.

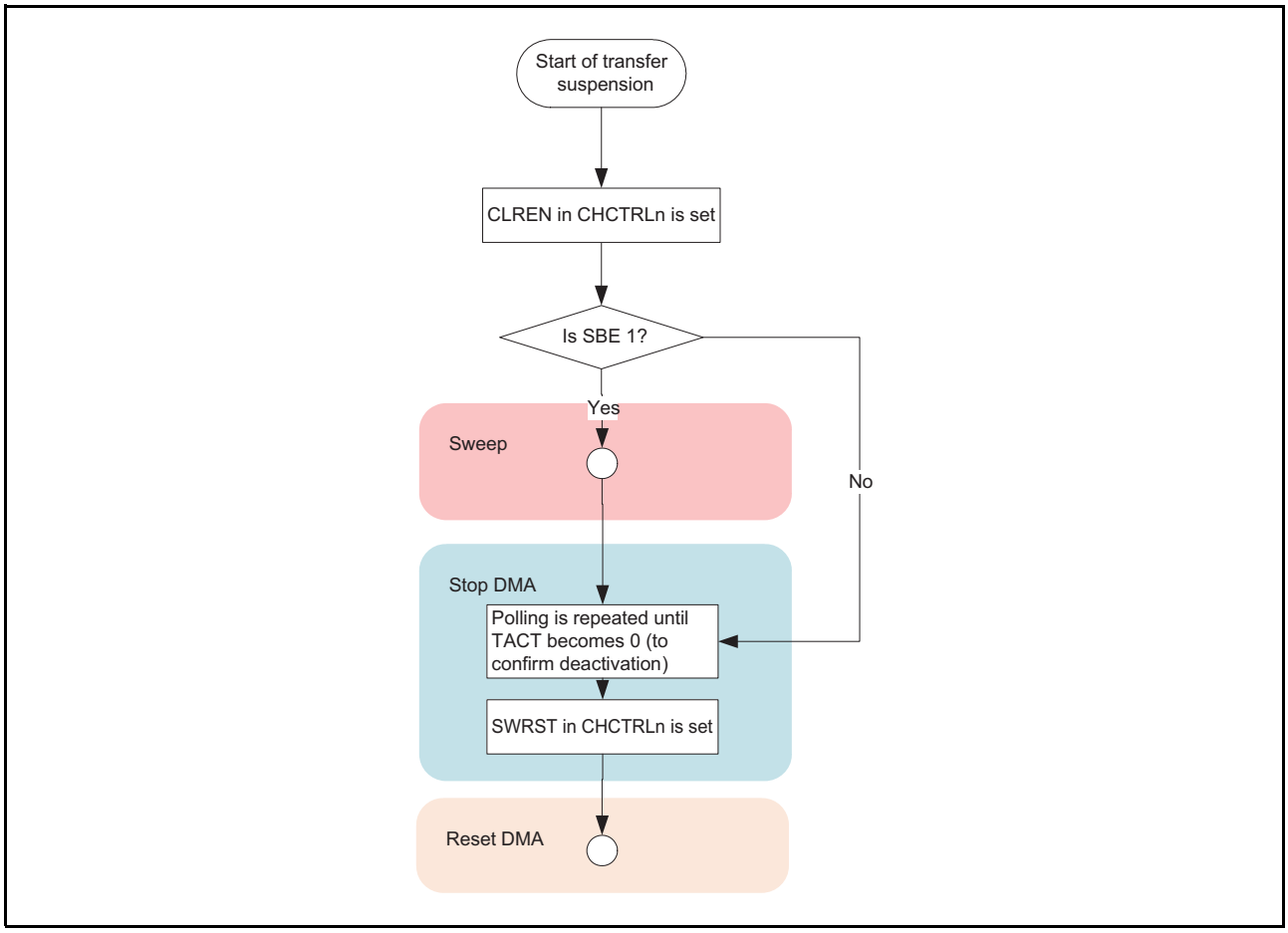


Figure 33.31 Transfer Suspension Flowchart

33.9.14 Access Type

33.9.14.1 DMA master transfer combination list

(1) Read access

The following describes characteristics of the issuance type for DMA read access.

- An access is performed for a beat align space whose size is set by SDS[2:0] in the CHCFG_n register, including the source address indicated in the CRSA_n register. A beat unaligned transfer for the bus is not performed. An excess area is sometimes read depending on CRSA_n or SKIP settings. In this case, the necessary data is imported into the buffer from the read data.

- The size and burst length are determined based on the value set in the SDS[2:0] field.
  If the value set in the SDS[2:0] field is equal to or less than the bus width
  Size: Value set in SDS[2:0]
  Burst type: SINGLE
  If the value set in the SDS[2:0] field is larger than the bus width
  Size: Bus width
  Burst type: Fixed-length burst (burst length = SDS[2:0] value/bus width)

The following indicates the access types for the bus.

Table 33.53 DMA Read Transfer Combination List

		AHB transfer					
		First transfer			Second transfer		
SDS	Source Address	Address	Data Size	Burst Type	Address	Data Size	Burst Type
0 (8 bits)	-	addr	8	SINGLE			
1 (16 bits)	2 byte align	{addr[31:1], 0B}	16	SINGLE			
	2 byte unalign				{addr[31:1], 0B} + 2H	16	SINGLE
2 (32 bits)	4 byte align	{addr[31:2], 00B}	32	SINGLE			
	4 byte unalign				{addr[31:2], 00B} + 4H	32	SINGLE
4 (128 bits)	16 byte align	{addr[31:4], 0H}	32	INCR4			
	16 byte unalign				{addr[31:4], 0H} + 10H	32	INCR4
5 (256 bits)	32 byte align	{addr[31:5], 00H}	32	INCR8			
	32 byte unalign				{addr[31:5], 00H} + 20H	32	INCR8
6 (512 bits)	64 byte align	{addr[31:6], 00H}	32	INCR16			
	64 byte unalign				{addr[31:6], 00H} + 40H	32	INCR16

Note: If EBT is detected in the middle of burst, 32-bit INCR burst is used to transfer the remaining data.

(2) Write access

The following describes characteristics of the issuance type for DMA write access.

- An access is performed from the destination address indicated by the CRDA_n register to the beat align boundary whose size is set by DDS[2:0] in the CHCFG_n register.

- The size and burst length are determined based on the value set in the DDS[2:0] field.
  If the value set in the DDS[2:0] field is equal to or less than the bus width
  Size: Value set in DDS[2:0]
  Burst type: SINGLE
  If the value set in the DDS[2:0] field is greater than the bus width
  Size: Bus width
  Burst type: Fixed-length burst (burst length = DDS[2:0] value/bus width)

- In write access, only the specified space is accessed. In the following cases, the combination of values smaller than the value set in the DDS[2:0] field is used for access.

  - The destination address is beat-unaligned for the value set in the DDS[2:0] field.
  - An access specified in the DDS field will be across the SKIP boundary.
  - The size specified in the DDS[2:0] field is too large for the number of remaining bytes to be transferred.

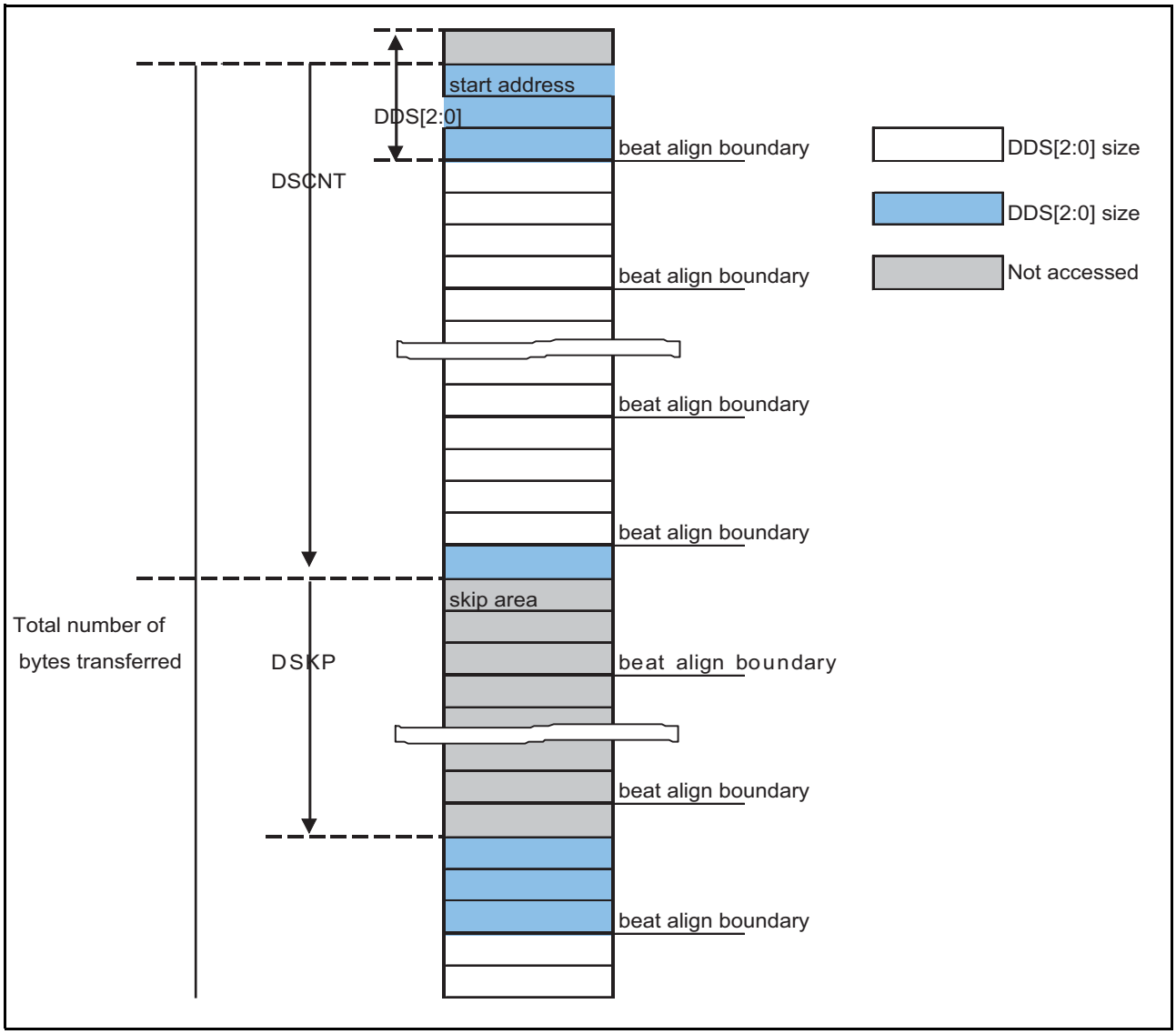


Figure 33.32 Example of DMA Write Access Space and Access Types

The following shows the access types for the bus in the case of beat-aligned.

Table 33.54 DMA Write Transfer Combination List

	AHB transfer		
	First transfer		
DDS[2:0]	Address	Data Size	Burst Type
0 (8 bits)	addr	8	SINGLE
1 (16 bits)	{addr[31:1], 0B}	16	SINGLE
2 (32 bits)	{addr[31:2], 00B}	32	SINGLE
4 (128 bits)	{addr[31:4], 0H}	32	INCR4
5 (256 bits)	{addr[31:5], 00H}	32	INCR8
6 (512 bits)	{addr[31:6], 00H}	32	INCR16

33.9.14.2 DMA master descriptor combination list

(1) Read access

The following describes characteristics of a descriptor access.

- Accesses are performed for 8-word beat align space including the LINK address (the address indicated in the CRLA_n register). A beat unaligned transfer is not performed.
- The size and burst length set by INCR8 are used.
- The descriptor format (DSCFM) of the read header is analyzed, and then the descriptor data is set in an internal register.
- If the descriptor extends over the 8-word boundary, an additional read is performed on the succeeding eight words.

The following indicates the types of access to the bus.

Table 33.55 Descriptor Read Transfer Combination List

Descriptor Format	Address	AHB Transfer					
		First Transfer			Second Transfer		
	addr[4:0]	Address	Size	Burst	Address	Size	Burst
4 words	00H	{addr[31:4], 4'b0}	32	INCR8			
	04H						
	08H						
	0CH						
	10H						
	14H				{addr[31:4], 4'b0} +20H	32	INCR8
	18H						
	1CH						
8 words	00H						
	04H				{addr[31:4], 4'b0} +20H	32	INCR8
	08H						
	0CH						
	10H						
	14H						
	18H						
	1CH						

Note: If EBT is detected in the middle of burst, 32-bit INCR burst is used to transfer the remaining data.

(2) Write access

Use single transfer as the issuance type for writing data back to a descriptor. The following indicates the types of access to the bus.

Table 33.56 Descriptor Write Transfer Combination List

Type	AHB transfer		
	Address	Size	Burst
Write-back in normal mode	{addr[31:2], 2'b0} + 3H	8	SINGLE
Write-back in the case of an error	{addr[31:2], 2'b0} + 2H	8	SINGLE

33.9.15 Arbitration between DMACs

Arbitration between internal DMACs is performed in round-robin mode.
In round-robin mode, the highest priority is given to the DMAC whose DMAC number is the DMC number being used for transfer + 1. Immediately after a reset, DMAC0 has the highest priority.

Table 33.57 Priority of a transfer request for DMACs that are performing transfer

	Next DMAC	
Current DMAC	DMAC0	DMAC1
DMAC0	2	1
DMAC1	1	2

Priority: 1 (high), 2 (low)

33.9.16 Flowcharts of Transitions to Deep Standby and Resumption from Deep Standby

33.9.16.1 Setting at transition to deep standby

The following is the flowchart for transitions to deep standby. After having completed the procedure in the following flowchart, proceed with that in the flowchart in section 52.3.4 (1), Transition to Deep Standby Mode.

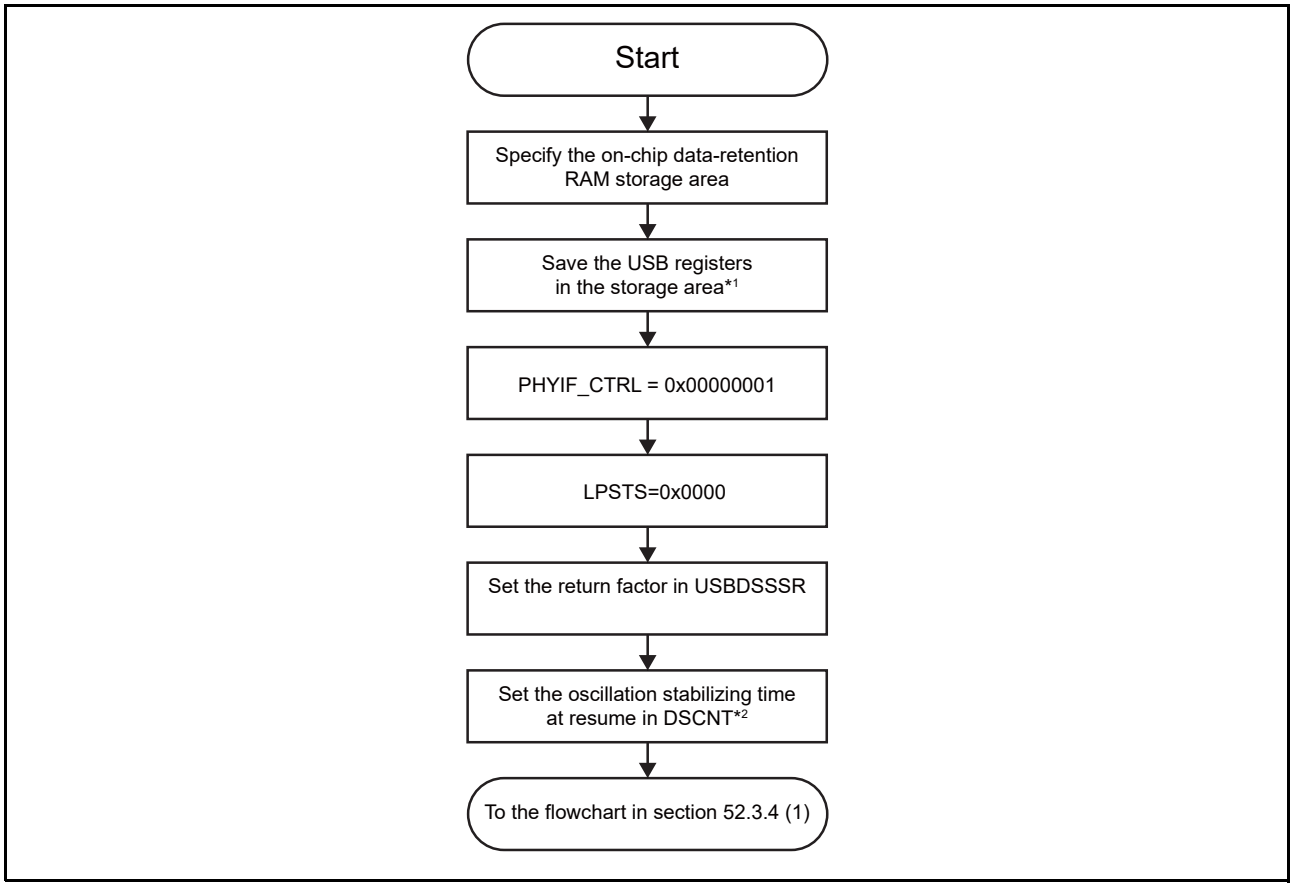


Figure 33.33 Flowchart of Transition to Deep Standby

Note 1. The registers listed in Table 33.58 must be saved.
Add registers to be saved as required even if they are not listed in the table.

Table 33.58 List of Registers to be Saved

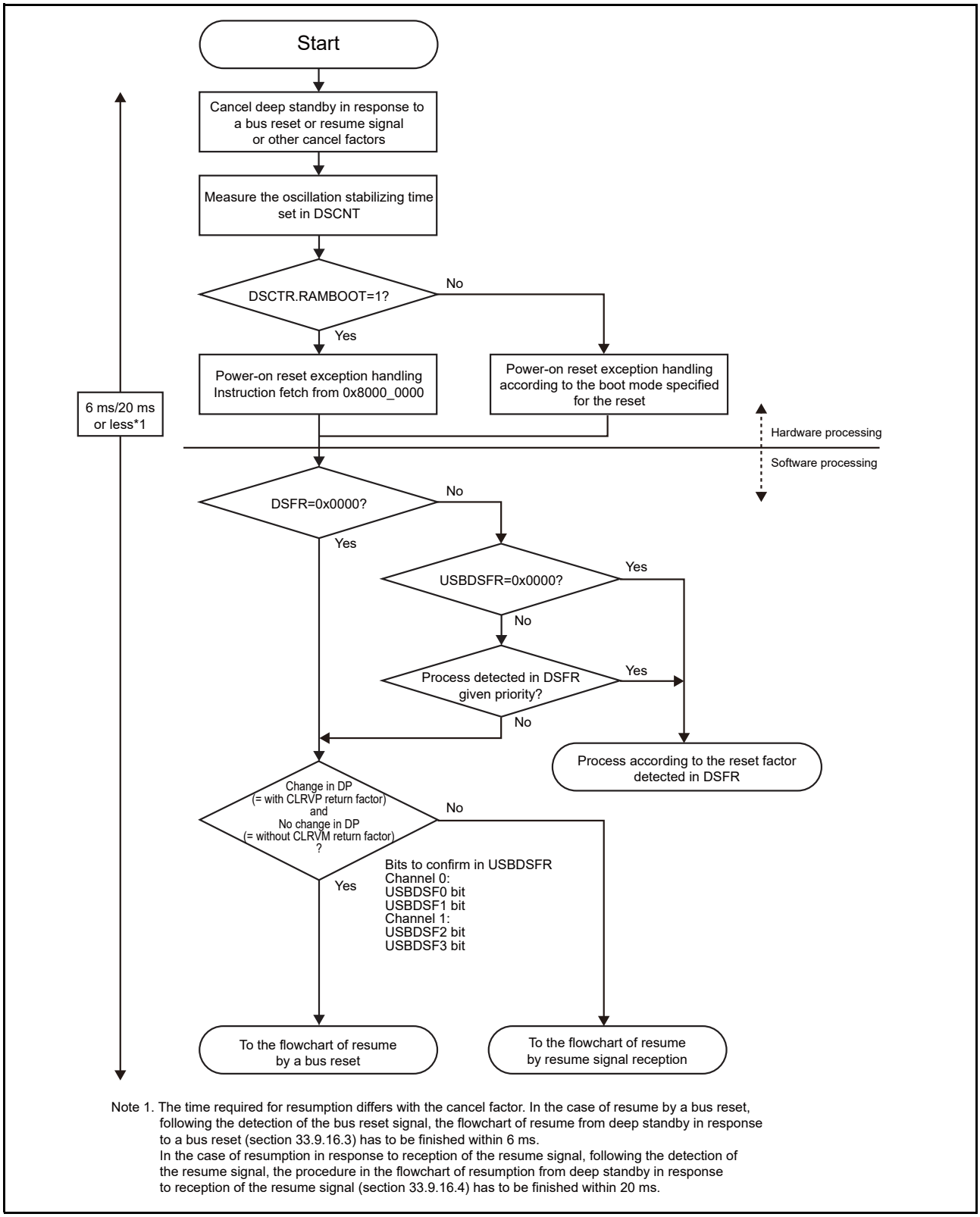
Register Name	Abbreviation	Register Name	Abbreviation
Common Control Register	COMMCTRL	DCP control register	DCPCTR
CFIFO port select register	CFIFOSEL	Pipe configuration register *	PIPECFG
Interrupt enable register 0	INTENB0	Pipe maximum packet size register *	PIPEMAXP
BRDY interrupt enable register	BRDYENB	Pipe cycle control register*	PIPEPERI
NRDY interrupt enable register	NRDYENB	Pipe buffer setting register*	PIPEBUF
BEMP interrupt enable register	BEMPENB	System configuration control register 0	SYSCFG0
Interrupt status register 0	INTSTS0	System configuration control register 1	SYSCFG1
USB address register	USBADDR	Pipe control registers (x = 1 to F)	PIPExCTR
DCP max. packet size register	DCPMAXP		

Note: * After the pipe numbers have been specified in the PIPESEL register, save the number of pipes used and their values.

Note 2. For details, see section 52.2.34, Deep standby cancel Oscillation stability count register (DSCNT).

33.9.16.2 Flowchart of determining the cancel factor of deep standby

The following shows the flowchart of determining the cancel factor, which should be proceeded after deep standby is canceled.



Note 1. The time required for resumption differs with the cancel factor. In the case of resume by a bus reset, following the detection of the bus reset signal, the flowchart of resume from deep standby in response to a bus reset (section 33.9.16.3) has to be finished within 6 ms.
In the case of resumption in response to reception of the resume signal, following the detection of the resume signal, the procedure in the flowchart of resumption from deep standby in response to reception of the resume signal (section 33.9.16.4) has to be finished within 20 ms.

Figure 33.34 Flowchart of Determining the Cancel Factor of Deep Standby

33.9.16.3 Flowchart of resumption from deep standby in response to a bus reset

The following shows the flowchart of resumption from deep standby in response a bus reset.

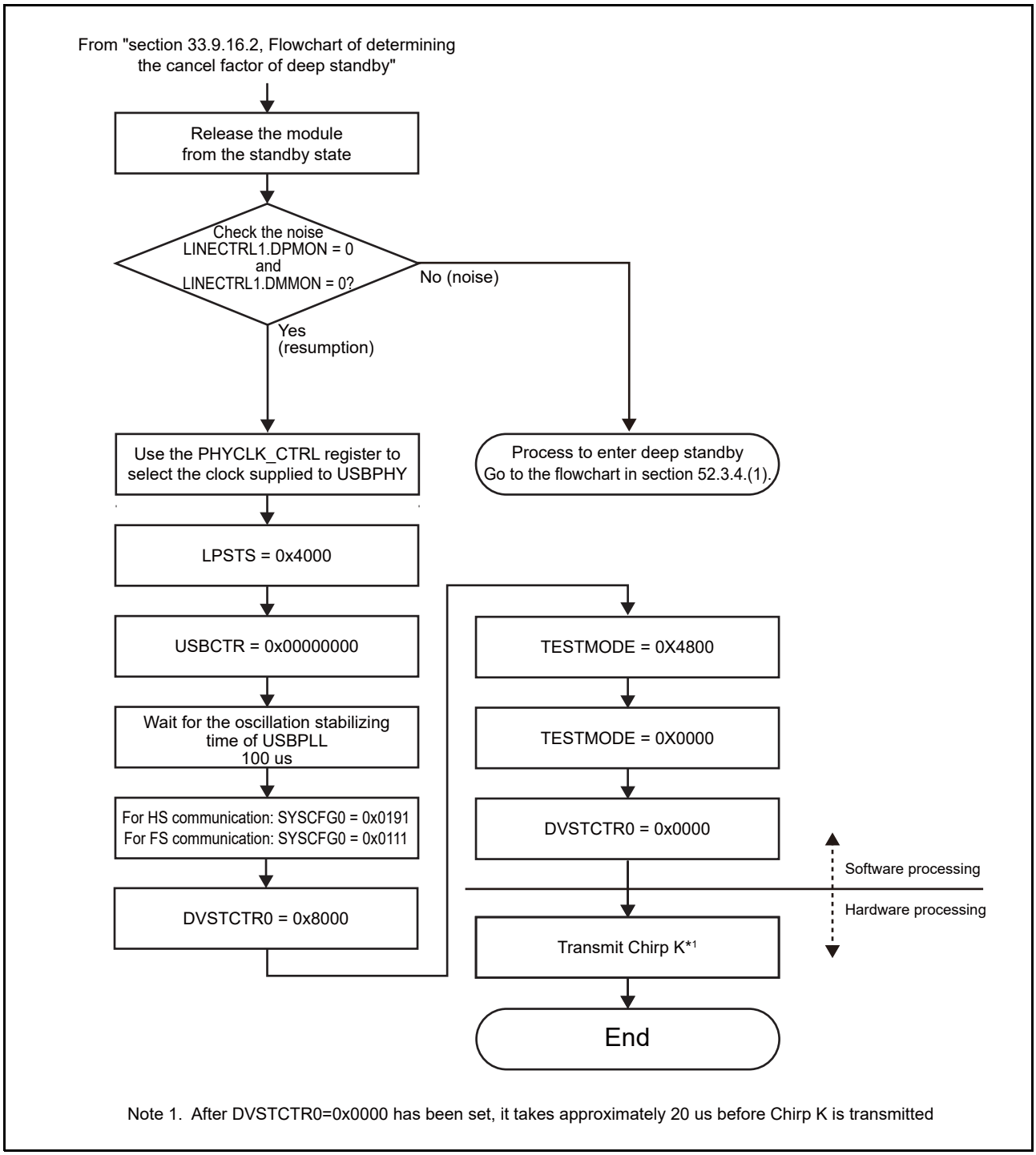


Figure 33.35 Flowchart of Resume from Deep Standby in Response to a Bus Reset

33.9.16.4 Flowchart of resumption from deep standby in response to reception of the resume signal

The following shows the flowchart of resumption from deep standby in response to reception of the resume signal.

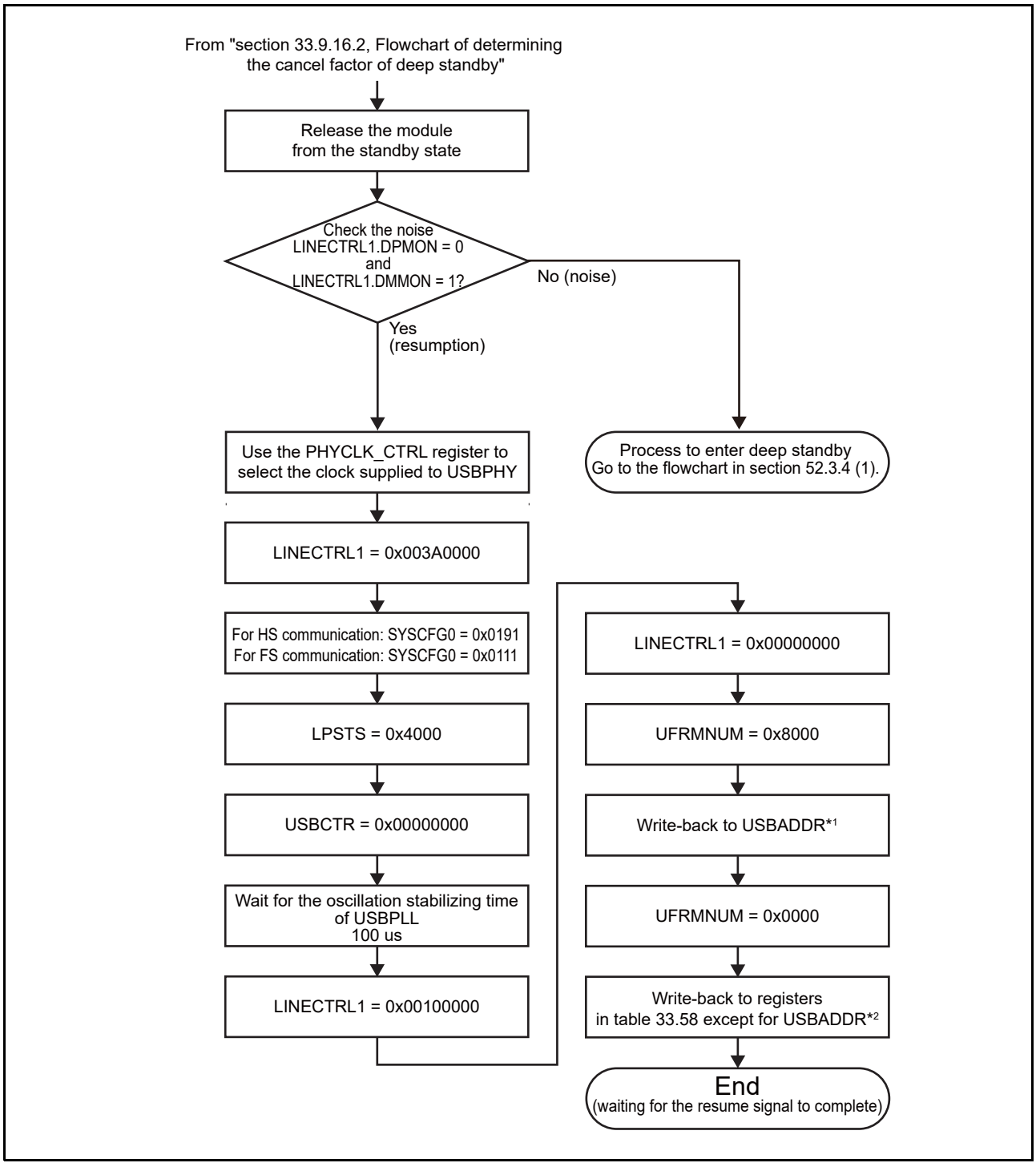


Figure 33.36 Flowchart of Resumption from Deep Standby in Response to Reception of the Resume Signal

Note 1. Writing back to the USBADDR register
Write back the USBADDR.USBADDR[6:0] value that was saved in the storage RAM area before entering deep standby to the USBADDR.USBADDR[6:0] bits.
Write back the INTSTS0.DVSQ[5:4] value that was saved in the storage RAM area before entering deep standby to the STSRECOV0[9:8] bits.
Write back 0 to the STSRECOV0[10] bit in FS operation and 1 in HS operation.

Note 2. Writing back to registers in Table 33.58 other than the USBADDR register
For the method of writing-back to the PIPECFG, PIPEMAXP, PIPEPERI, and PIPEBUF registers, see section 33.2.15, Pipe Configuration Registers (PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI).
The value to be written back to PIPExCTR depends on the value of PIPExCTR.SQMON.
When SQMON is 0, write back 1 to the SQCLR bit and write back the values that were saved in the storage RAM area before entering deep standby to the bits other than SQCLR.
When SQMON is 1, write back 1 to the SQSET bit and write back the values that were saved in the storage RAM area before entering deep standby to the bits other than SQSET.

33.9.17 Notes

33.9.17.1 Access

During read access, a beat align area is accessed by one transfer. Therefore, if beat unaligned is set, the beat align area including the specified area is accessed.

For example, if the Source Address is 0000_1038H and SDS is 5 (256-bit), read starts from area 0000_1020H, not from address 0000_1038H. At this time, if the area from 0000_1020H to 0000_1037H contains a register whose value changes by read, the operation might be disrupted.

To prevent problems, use the beat align setting to access a register whose value changes by read access or access to an area adjacent to FIFO.

33.9.17.2 Level Interrupt bit

This is a DMA interrupt output enable bit. Always set this bit to 1 irrespective of whether peripheral module interrupt USBFDMAmn (m, n = 0, 1) or USBFDMAERRm (m = 0, 1) is used.
To use peripheral module interrupt USBFDMAmn or USBFDMAERRm, set this bit to 1, and then set up the interrupt controller.

34. Video Display Controller 6 (1): Overview

34.1 Features

The video display controller 6 consists of the following six blocks. For the image synthesis, one plane of video image + two graphics planes, or three graphics planes can be selected.

1. Input controller: Input video image selection, sync signal adjustment, horizontal noise reduction, and brightness adjustment, gain adjustment, and YCbCr ↔ GBR conversion using a color matrix
2. Scaler: Scale up, scale down, and rotation of input video images using the frame buffer, and repeated recording of the specified number of fields in the frame buffer
3. Image quality improver: Black stretch, LTI/sharpness, and YCbCr ↔ GBR conversion using a color matrix
4. Image synthesizer: Synthesis of one plane of video image + two graphics planes, or three graphics planes
5. Output controller: Brightness/contrast adjustment, gamma correction, dither processing, output format conversion, control signal output for TFT-LCD panel
6. System controller: Interrupt control, panel clock control, CLUT table select signal status flag output

The functions of video display controller 6 are listed in Table 34.1.

Table 34.1 Features of video display controller 6

Item	Function
Operating frequency	Video input clock: 87 MHz or less (for RGB/YCbCr video image) Panel clock: 87 MHz or less (depends on the panel specifications)
Input video image specification	• 8-bit input conforming to ITU-R BT.656 standard (27 MHz, interlace signal) • 8-bit input conforming to ITU-R BT.656 extended standard (54 MHz, progressive signal) *1 • 8-bit input conforming to ITU-R BT.601 extended standard (27 MHz, interlace signal) *1 • 8-bit input conforming to ITU-R BT.601 extended standard (54 MHz, progressive signal) *1 • 16-bit input conforming to ITU-R BT.601 extended standard (13.5 MHz, interlace signal) *1 • Digital pin input: YCbCr422, YCbCr444, RGB888, RGB666, and RGB565 video image • Digital pin input size: Maximum input video image size to be set *2: 1920 pixels × 1080 lines (horizontal × vertical) Note 1. The ITU-R BT.656 and 601 standards do not include the description regarding the progressive signal. The ITU-R BT.601 standard does not include the description regarding the connection interface. Note 2. Depends on the AC characteristics of the connected device. • Examples of input video image size: XGA (1024 × 768), SVGA (800 × 600), WVGA (800 × 480), VGA (640 × 480), WQVGA (480 × 240), QVGA in landscape (320 × 240), QVGA in portrait (240 × 320)
Video image recording function	• Storing the video image in the YCbCr422/YCbCr444/RGB565/RGB888 format at a rate of 1/1, 1/2, 1/4, or 1/8 field. • Maximum video image size to be stored: 1440 pixels × 1024 lines (horizontal × vertical)
Video image quality adjustment function	Contrast adjustment, brightness adjustment, horizontal noise reduction, black stretch, LTI/sharpness
Video image scaling processing	Vertical: ×1/8 to ×8, linear/hold interpolation Horizontal: ×1/8 to ×8, linear/hold interpolation IP conversion can be performed by adjusting the initial phase.
Video image rotation function	• 0/90/180/270 degree rotation in the YCbCr422/RGB565 format • Horizontal mirroring in the YCbCr422/YCbCr444/RGB565/RGB888 format

Table 34.1 Features of video display controller 6

Item	Function
Graphics	• Number of graphic planes: Three planes (graphics 0, graphics 2, and graphics 3) • Supported pixel formats: - RGB565 progressive format (α: none, R: 5 bits, G: 6 bits, B: 5 bits; 16 bits in total) - RGB888 progressive format (α: none, R: 8 bits, G: 8 bits, B: 8 bits; 24 bits in total) - α RGB1555 progressive format (α: 1 bit, R: 5 bits, G: 5 bits, B: 5 bits; 16 bits in total) - α RGB4444 progressive format (α: 4 bits, R: 4 bits, G: 4 bits, B: 4 bits; 16 bits in total) - α RGB8888 progressive format (α: 8 bits, R: 8 bits, G: 8 bits, B: 8 bits; 32 bits in total) - RGBα5551 progressive format (R: 5 bits, G: 5 bits, B: 5 bits, α: 1 bit; 16 bits in total) - RGBα8888 progressive format (R: 8 bits, G: 8 bits, B: 8 bits, α: 8 bits; 32 bits in total) - CLUT8 progressive format (CLUT: 8 bits) - CLUT4 progressive format (CLUT: 4 bits) - CLUT1 progressive format (CLUT: 1 bits) - YCbCr422 progressive format (Y: 8 bits, Cb/Cr: 8 bits; 16 bits in total) (only for graphics 0) - YCbCr444 progressive format (Y: 8 bits, Cb/Cr: 8 bits; 16 bits in total) (only for graphics 0) • Maximum image size to be read: 1440 pixels × 1440 lines (horizontal × vertical)
Graphics function	Alpha blending in rectangular area: Mixes images according to transparency rate α in the specified area (fade-in and fade-out functions are available.) Chroma-key: Mixes images using the specified RGB color and CLUT value according to transparency rate α. Alpha blending in one pixel units: Mixes images according to transparency rate α when the target graphics image is in the αRGB1555, αRGB4444, αRGB8888, RGBα5551, RGBα8888, or CLUT8/4/1 format.
	For each dot, the priority among the α values of the above functions is as follows: Alpha blending in rectangular area > Chroma-key > Alpha blending in one pixel units
Output video image size	Maximum output video image size to be set*: 1999 pixels × 2035 lines (horizontal × vertical) Note: * Depends on the AC characteristics of the display panel. Examples of output video image size: • XGA (1024 × 768) • SVGA (800 × 600), WVGA (800 × 480), • VGA (640 × 480), WQVGA (480 × 240), • QVGA size in landscape (320 × 240) • QVGA size in portrait (240 × 320)
Output video image format	• RGB888 progressive video output (24-bit parallel output) • RGB666 progressive video output (18-bit parallel output) • RGB565 progressive video output (16-bit parallel output) • RGB888 progressive video output (8-bit serial output)
Panel output adjustment	Panel brightness/contrast adjustment, RGB gamma correction, dither processing, output format conversion
Sync signal output	Control signal output for the TFT-LCD panel
Interrupt output	• Vsync signal for video image input/output • Line interrupt output (can be output on a desired line.) • Erroneous Vsync cycle detection signal for video input • Field write completion signal • Overflow/underflow signal for the internal buffer

34.2 Block Diagram

Figure 34.1 to Figure 34.3 show the entire block diagram of this module.

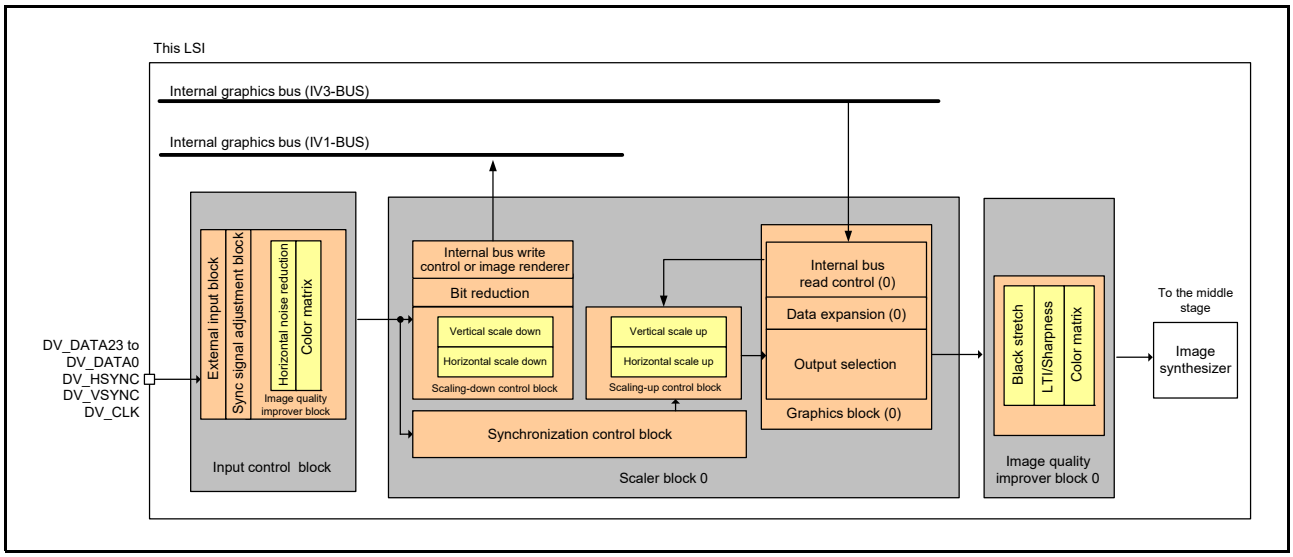


Figure 34.1 Video Display Controller 6 Former Stage Block Diagram

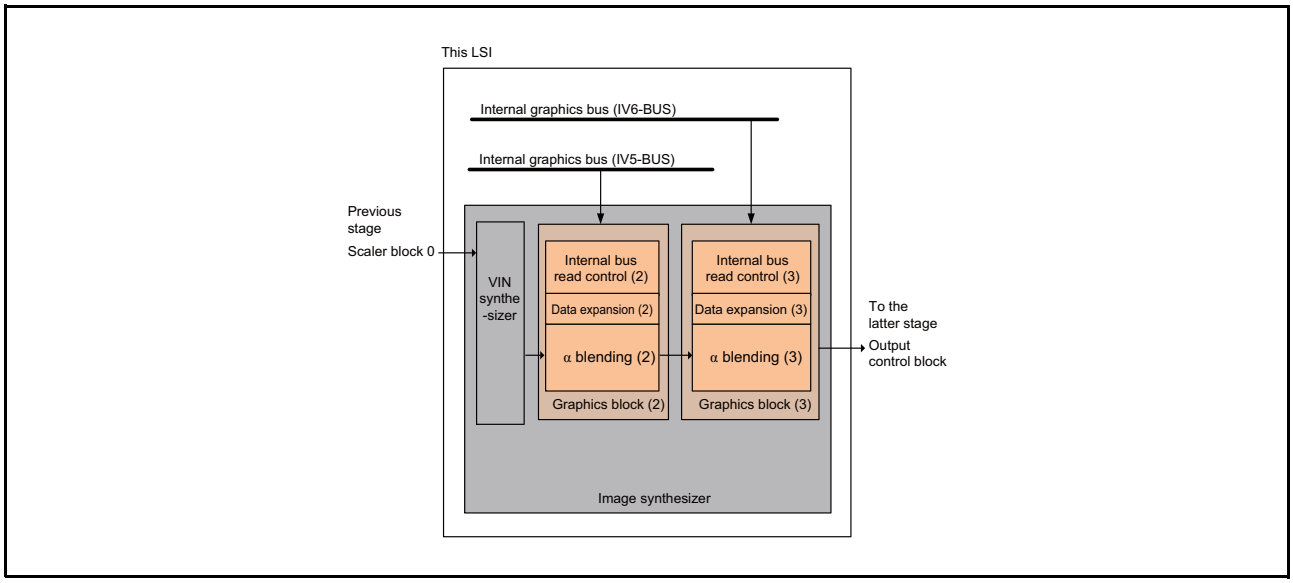


Figure 34.2 Video Display Controller 6 Middle Stage Block Diagram

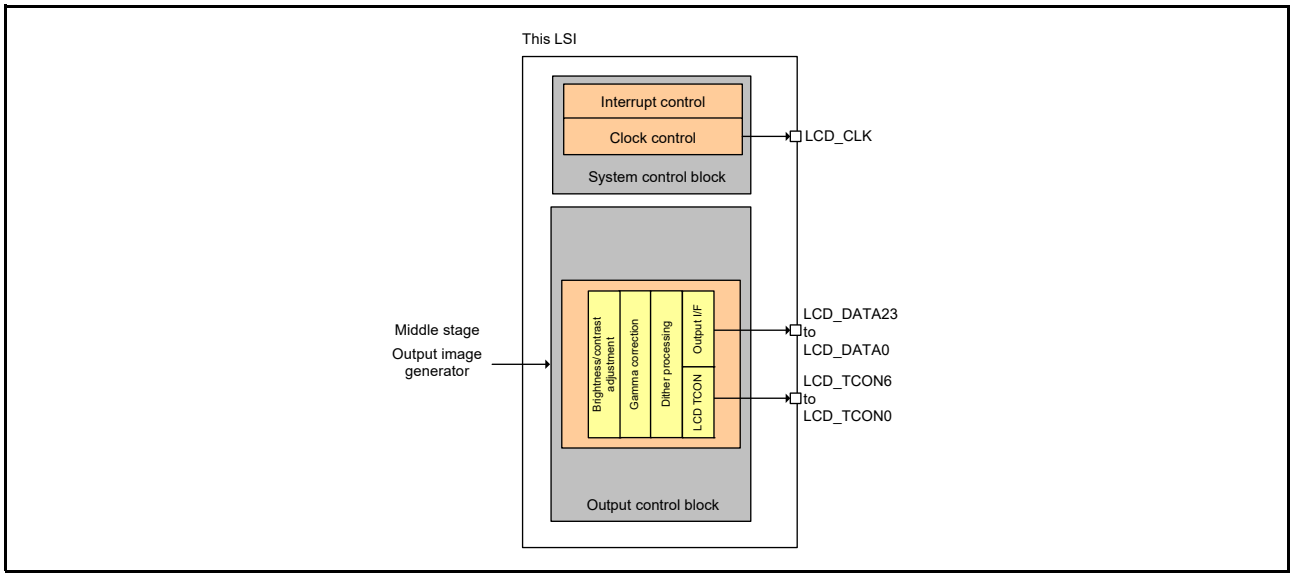


Figure 34.3 Video Display Controller 6 Latter Stage Block Diagram

34.3 Input/Output Pins

Table 34.2 shows the pin configuration.

Table 34.2 Input/Output Pins

Symbol	I/O	Pin Name	Function
DV0_CLK	Input	External input clock 0	External input 0 clock pin
DV0_VSYNC	Input	External input Vsync 0	External input 0 Vsync signal pin
DV0_HSYNC	Input	External input Hsync 0	External input 0 Hsync signal pin
DV0_DATA23 to DV0_DATA0	Input	External input video image data 0	External input 0 video image data pin
LCD0_CLK	Output	Panel clock 0	Panel output 0 clock pin
LCD0_DATA23 to LCD0_DATA0	Output	Video image data 0 for panel	Panel output 0 video image data pin
LCD0_TCON6 to LCD0_TCON0	Output	Control signal 0 for panel	Panel output 0 timing control pin
LCD0_EXTCLK	Input	Panel clock source 0	Panel clock source 0 input pin

34.4 Clocks

There are two clocks to be mainly used by the video display controller 6: the video image clock and pixel clock.

The video image clock is used while the video image is processed in the input controller, passed to the scale-down control block in the scaler, and then written to the buffer (internal bus write control). The DV_CLK clock is used as the video image clock.

The pixel clock is used in graphics read-out processing by the scaler (internal bus read controller) through output controller processing. When the parallel RGB output is selected in the output controller, the frequencies of the pixel clock and panel clock (LCD_CLK) are the same. The panel clock can be selected from the video clock, LCD0_EXTCLK peripheral bus clock 1 (P1φ), and LVDS PLL output clock*1 with SYSCNT_PANEL_CLK.PANEL_ICKSEL[1:0] and SYSCNT_PANEL_CLK.PANEL_OCKSEL[1:0] of the system controller. When the serial RGB output (3/4 speed mode) is selected in the output controller, the pixel clock frequency is 1/3 or 1/4 the panel clock (LCD_CLK) frequency.

Note 1. For details of LVDS PLL , refer to section 41, LVDS Output Interface.

34.5 Hsync and Vsync Signals

Hsync and Vsync signals to be used in the logic stage following the scale-up control block of the scaler are generated by the synchronization control block of the scaler. Since the Hsync and Vsync signals are used as the reference signals for the LCD TCON, which generates various panel driving timings, they are also the reference signals for the control signals (LCD_TCON6 to LCD_TCON0 pins) passed to the panel.

The output Hsync signal always operates at a free-running frequency, and the horizontal period is set with SC_SCL0_FRC4.SC_RES_FH[10:0]. On the other hand, the output Vsync signal is selected from the external input and free-running Vsync signals with SC_SCL0_FRC3.SC_RES_VS_SEL of the scaler.

34.5.1 External Input Vsync

(1) Operation Outline

In this mode, the output Vsync signal is generated according to an external input Vsync signal. When displaying video image input from a digital pin on the panel and the pointer buffer is not in use, always use this mode. However, the output Hsync signal is free running even in this mode. Figure 34.4 shows the timing of external input Vsync signal.

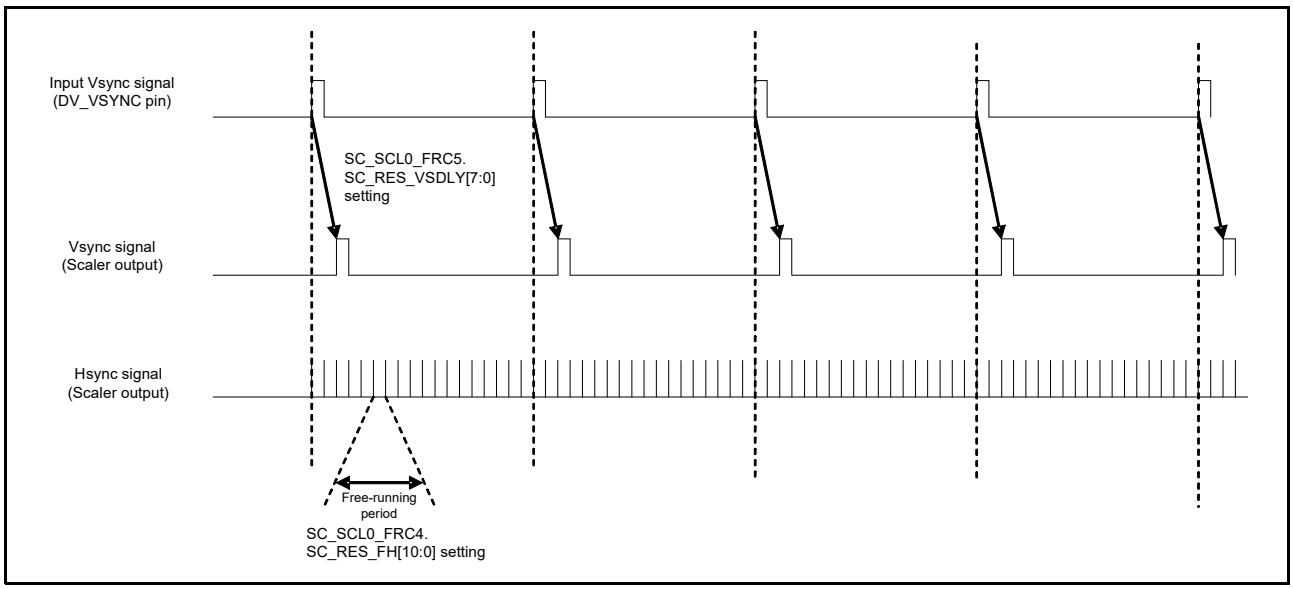


Figure 34.4 External Input Vsync Timing

(2) Notes

When Vsync is externally input, generation of the output Vsync signal is based on the external Vsync signal. That is, the output Vsync signal follows the input Vsync signal, so if an unstable Vsync signal is input, the output Vsync signal will also be unstable.

Since the output Hsync signal is generated according to the frequency generated by a free-running clock and the Vsync signal is generated from the video input as a base, the signals will not be in synchronization. This module adjusts the timing between these signals by adjusting the output Vsync signal so that it stays in time with the output Hsync signal. Therefore, even if the input Vsync signal is stable, the timing of the output Vsync signal may be increased or decreased by up to one line to stay in synchronization with the timing of the output Hsync signal.

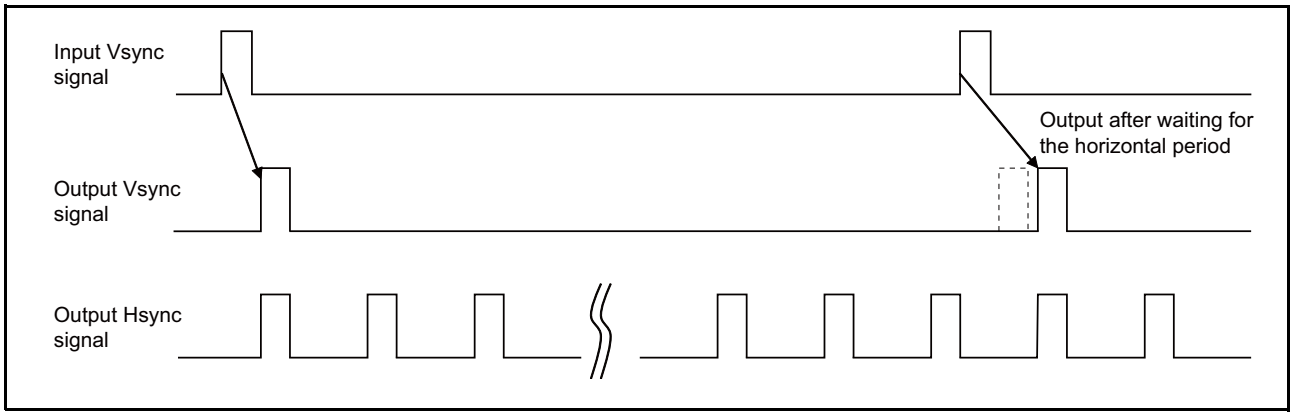


Figure 34.5 Detailed Timing Chart for Generation of the Output Vsync Signal

34.5.2 Free-Running Vsync

(1) Operation Outline

In this mode, the Vsync signal is generated according to the pixel clock (free running). The vertical period is selected with SC_SCL0_FRC4.SC_RES_FV[10:0]. The output Hsync signal is also free running. Figure 34.6 shows the timing.

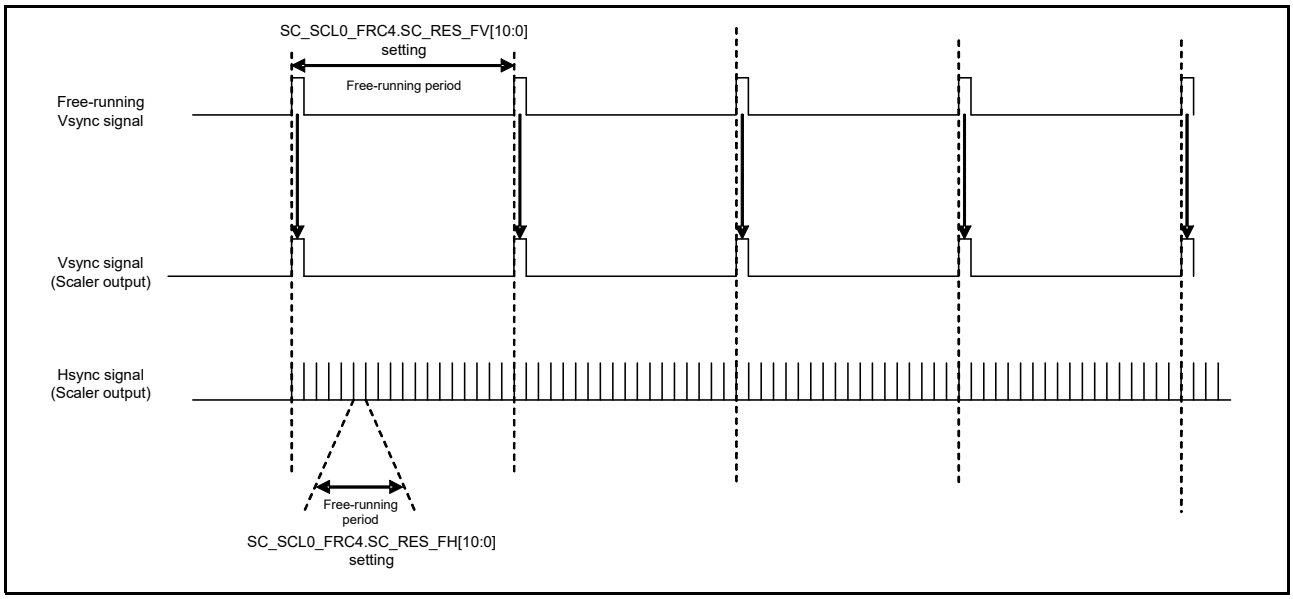


Figure 34.6 Free Running Vsync Timing

(2) Pointer Buffers

In free-running vertical synchronization mode, output of the input video image to a panel may lead to flicker in the output video image. This occurs when the input and output vertical sync signals are not in synchronization. To prevent this, use the pointer buffers to adjust the timing between the input and output video images in frame units. If the input Vsync signal is faster than the output Vsync signal, frames from the input video image are skipped in the output image for display. On the other hand, when the input Vsync signal is slower than the output Vsync signal, frames from the input video image are repeated. However, when the difference in timing between the input and output Vsync signals is too large, the pointer buffers cannot deal with the difference, so flicker may occur. As more buffers are used, the pointer buffers can deal with larger differences in frequency.

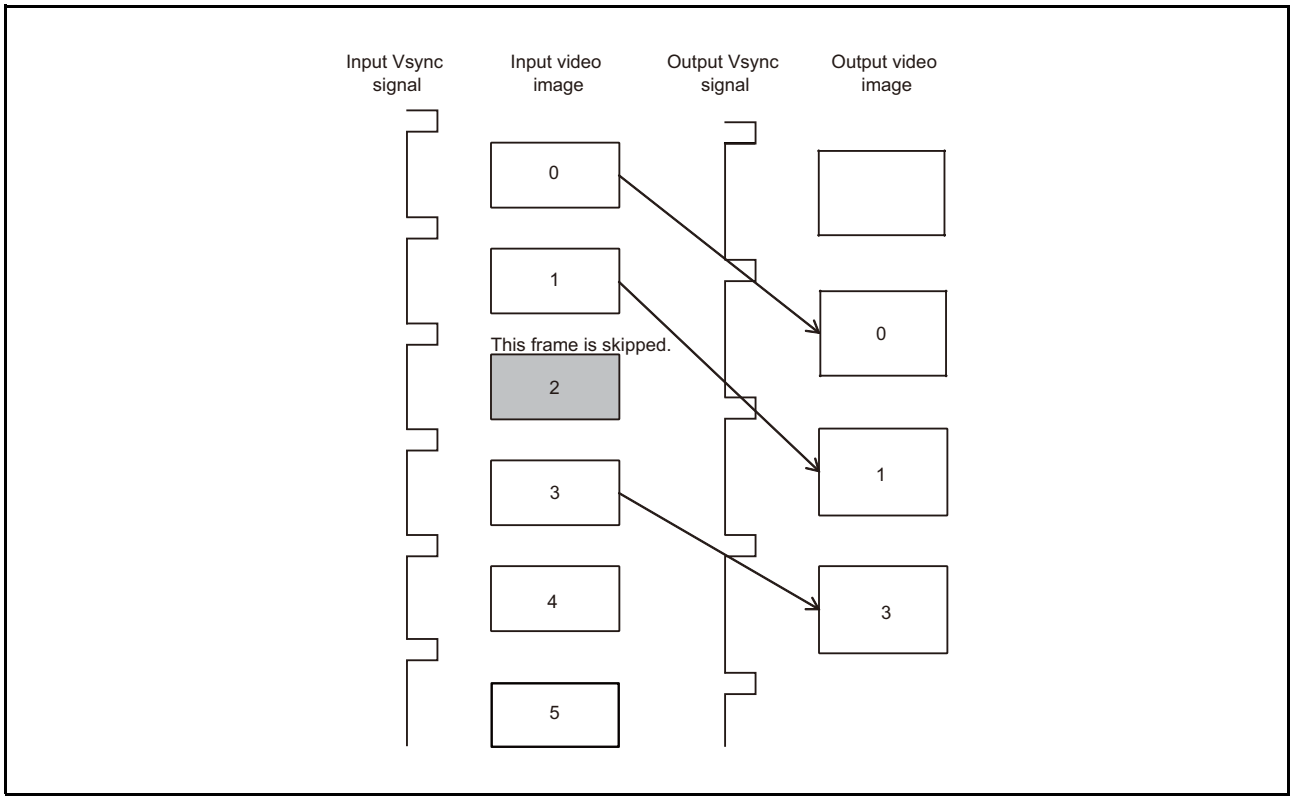


Figure 34.7 Timing when the Input Vsync Signal is Faster than the Output Vsync Signal

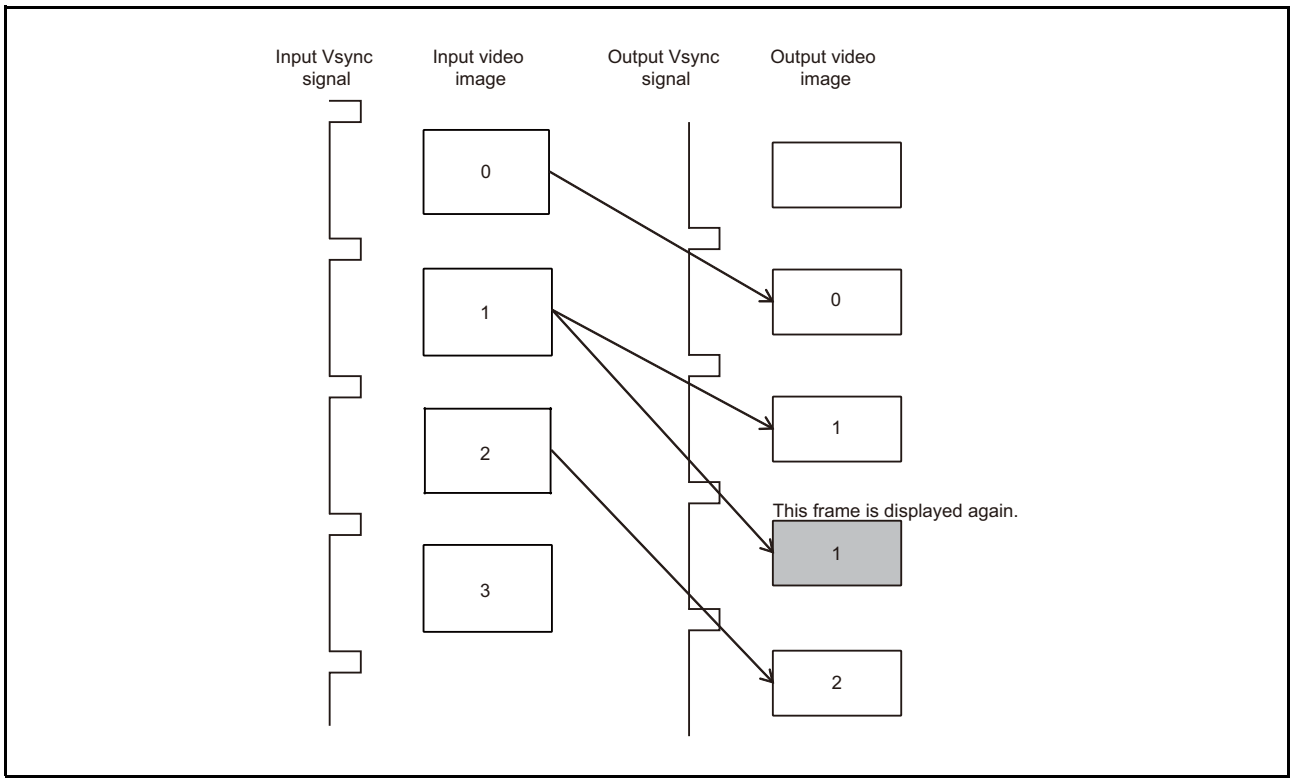


Figure 34.8 Timing when the Input Vsync Signal is Slower than the Output Vsync Signal

34.5.3 Usage Note on Changing Vsync Signal Selections

When the Vsync signal selection is changed, the output Vsync signal is discontinuous, resulting in disordered panel display. In this case, perform the mute processing according to the panel specification as necessary and change the Vsync signal selection.

35. Video Display Controller 6 (2): Input Controller

35.1 Input Controller Functions

35.1.1 Overview of Functions

The input controller performs the on/off control of input supplied via the external input pins, and subjects the signals to synchronization adjustment, horizontal noise reduction, and brightness adjustment, gain adjustment, and YCbCr ↔ GBR conversion using a color matrix.

The functional block diagram of the input controller is shown below.

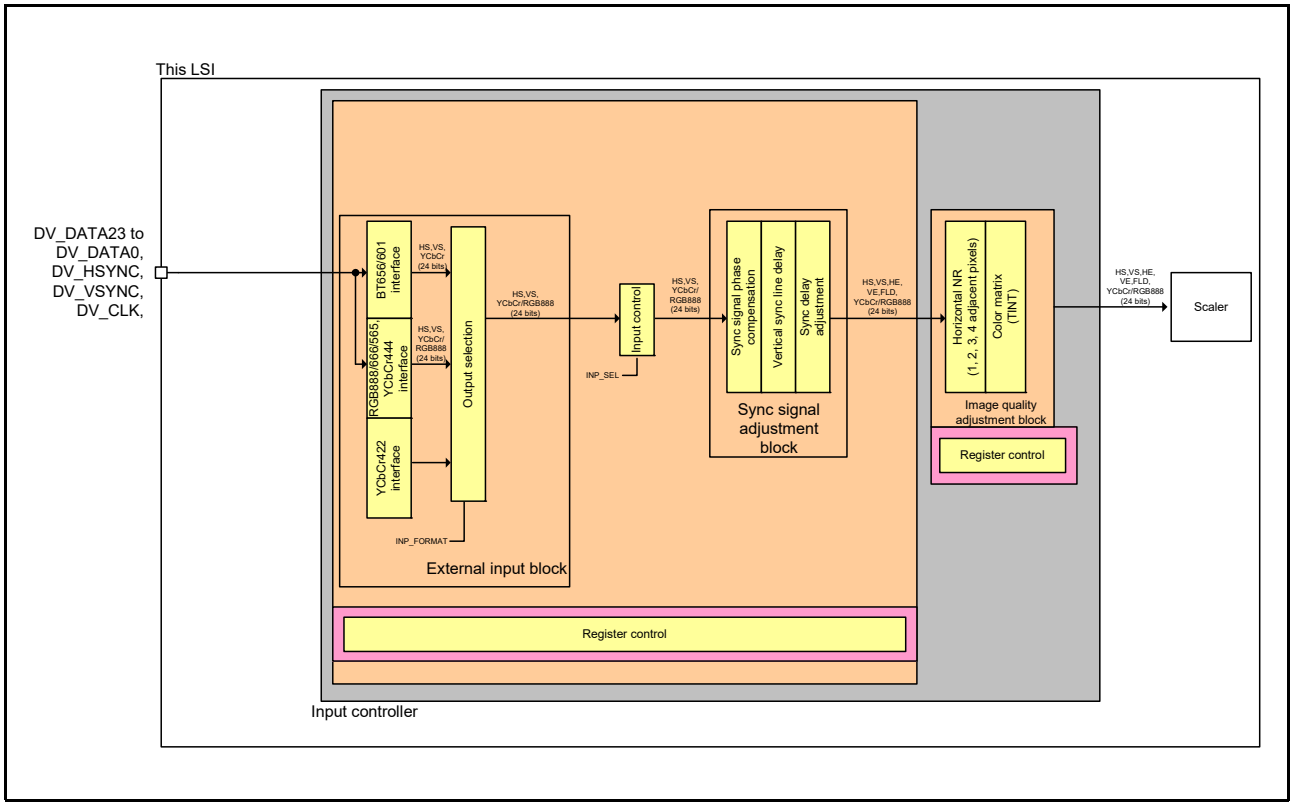


Figure 35.1 Functional Block Diagram of Input Controller

35.1.2 Updating Registers of External Signal Input Block and Sync Signal Adjustment Block

The control registers of the external input block and sync signal adjustment block are updated by setting the relevant update control bit to 1.

For the control registers other than the IMGCNT_DRC_REG register of the image quality adjustment block, the update timing is controlled using the Vsync signal.

After 1 is set to the bits in the update control register, the contents of the relevant registers are actually modified at the rising edge of the Vsync signal, when the update control register is automatically cleared to 0.

Table 35.1 Register Update Control

Register Name	Bit Name	Initial Value	Description
INP_UPDATE	INP_EXT_UPDATE	0	External Input Block Register Update 0: Registers are not updated. 1: Registers are updated.
INP_UPDATE	INP_IMG_UPDATE	0	Sync Signal Adjustment Block Register Update 0: Registers are not updated. 1: Registers are updated.
IMGCNT_UPDATE	IMGCNT_VEN	0	Image Quality Adjustment Block Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync signal.

35.1.3 Controlling Input

The input controller performs the on/off control of input supplied via the external input pins.

Table 35.2 Input Control

Register Name	Bit Name	Initial Value	Description
INP_SEL_CNT	INP_SEL	0	On/Off Control of Input Supplied via External Input Pin 0: Input supplied via the external input pins is off. 1: Input supplied via the external input pins is on.

35.1.4 Controlling Externally Input Video Signals

The externally input video image signals in the YCbCr444, RGB888, RGB666, RGB565, BT656 (extended), BT601 (extended), YCbCr422 (the 16-bit data format of the BT601 standard) formats can be handled.

The BT656 signals can be used for the 525-line and 59.94-Hz (27.0-MHz) or the 625-line and 50.00-Hz (27.0-MHz) interlace signals and for the 525-line and 59.94-Hz (54.0-MHz) or the 625-line and 50.00-Hz (54.0-MHz) BT656-extended progressive signals.

The BT601 signals can be used for the 8-bit data line 525-line and 59.94 Hz (27.0-MHz) or the 625-line and 50.00-Hz (27.0-MHz) interlace signals and for the 525-line and 59.94 Hz (54.0-MHz) or the 625-line and 50.00-Hz (54.0-MHz) extended progressive signals.

The YCbCr422 signals can be used for the 16-bit data line 525-line and 59.94-Hz (13.5-MHz) or the 625-line and 50.00-Hz (13.5-MHz) BT601 interlace signals.

The above signals can be selected by the INP_FORMAT[2:0] bits. Bit endian change and B/R signal swap are controlled by setting the INP_ENDIAN_ON and INP_SWAP_ON bits.

Table 35.3 Externally Input Video Signal Control

Register Name	Bit Name	Initial Value	Description
INP_SEL_CNT	INP_FORMAT[2:0]	000	External Input Format Select 0: YCbCr444, RGB888 1: RGB666 2: RGB565 3: BT656 4: BT601 5: YCbCr422 6, 7: Setting prohibited
INP_EXT_ SYNC_CNT	INP_ENDIAN_ON	0	External Input Bit Endian Change On/Off Control 0: Off 1: On
INP_EXT_ SYNC_CNT	INP_SWAP_ON	0	External Input B/R Signal Swap On/Off Control 0: Off 1: On

35.1.5 Selecting Clock Edge for Externally Input Signals

The clock edge for receiving the video image signals, Vsync signals, and Hsync signals is individually selected with the INP_PXD_EDGE, INP_VS_EDGE, INP_HS_EDGE bits.

Table 35.4 Externally Input Clock Edge Selection

Register Name	Bit Name	Initial Value	Description
INP_SEL_CNT	INP_PXD_EDGE	0	Clock Edge Select for Capturing External Input Video Image Signals DV_DATA23 to DV_DATA0 0: Rising edge 1: Falling edge
INP_SEL_CNT	INP_VS_EDGE	0	Clock Edge Select for Capturing External Input Vsync Signal DV_VSYNC 0: Rising edge 1: Falling edge
INP_SEL_CNT	INP_HS_EDGE	0	Clock Edge Select for Capturing External Input Hsync Signal DV_HSYNC 0: Rising edge 1: Falling edge

Figure 35.2 shows the typical input timing of externally input signals.

The input signals can be received at the rising edge of the clock signal DV_CLK when the INP_PXD_EDGE, INP_VS_EDGE, and INP_ES_EDGE bits are 0.

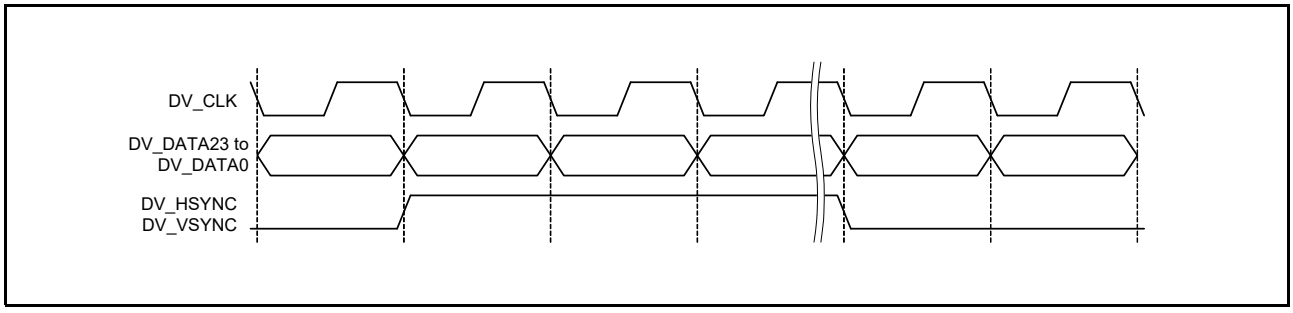


Figure 35.2 Typical Input Timing of Externally Input Signals (Clock Phase)

35.1.6 Externally Input Sync Signal Inversion Control

Inversion of polarity of the Vsync and Hsync signals can be controlled by the INP_VS_INV and INP_HS_INV bits.

Table 35.5 Sync Signal Inversion Control

Register Name	Bit Name	Initial Value	Description
INP_EXT_ SYNC_CNT	INP_VS_INV	0	External Input Vsync Signal DV_VSYNC Inversion Control 0: Not inverted (positive polarity) 1: Inverted (negative polarity)
INP_EXT_ SYNC_CNT	INP_HS_INV	0	External Input Hsync Signal DV_HSYNC Inversion Control 0: Not inverted (positive polarity) 1: Inverted (negative polarity)

35.1.7 Bit Allocation of Externally Input Video Image Signals

Allocation of the externally input video image signal pins DV_DATA to the signals in each format is described below.

(1) YCbCr444/RGB888 Input

When the external input is of YCbCr444/RGB888 format, the video image signal pins DV_DATA are allocated to the internal signals Y/GOUT, Cb/BOUT, Cr/ROUT, as shown in Table 35.6.

Table 35.6 Bit Allocation of DV_DATA Pin Inputs when the External Input is of YCbCr444/RGB888

INP_FORMAT[2:0]	0	0	0	0
INP_ENDIAN_ON	0	0	1	1
INP_SWAP_ON	0	1	0	1
DV_DATA23	Cr/ROUT[7]	Cb/BOUT[7]	Cr/ROUT[0]	Cb/BOUT[0]
DV_DATA22	Cr/ROUT[6]	Cb/BOUT[6]	Cr/ROUT[1]	Cb/BOUT[1]
DV_DATA21	Cr/ROUT[5]	Cb/BOUT[5]	Cr/ROUT[2]	Cb/BOUT[2]
DV_DATA20	Cr/ROUT[4]	Cb/BOUT[4]	Cr/ROUT[3]	Cb/BOUT[3]
DV_DATA19	Cr/ROUT[3]	Cb/BOUT[3]	Cr/ROUT[4]	Cb/BOUT[4]
DV_DATA18	Cr/ROUT[2]	Cb/BOUT[2]	Cr/ROUT[5]	Cb/BOUT[5]
DV_DATA17	Cr/ROUT[1]	Cb/BOUT[1]	Cr/ROUT[6]	Cb/BOUT[6]
DV_DATA16	Cr/ROUT[0]	Cb/BOUT[0]	Cr/ROUT[7]	Cb/BOUT[7]
DV_DATA15	Y/GOUT[7]	Y/GOUT[7]	Y/GOUT[0]	Y/GOUT[0]
DV_DATA14	Y/GOUT[6]	Y/GOUT[6]	Y/GOUT[1]	Y/GOUT[1]
DV_DATA13	Y/GOUT[5]	Y/GOUT[5]	Y/GOUT[2]	Y/GOUT[2]
DV_DATA12	Y/GOUT[4]	Y/GOUT[4]	Y/GOUT[3]	Y/GOUT[3]
DV_DATA11	Y/GOUT[3]	Y/GOUT[3]	Y/GOUT[4]	Y/GOUT[4]
DV_DATA10	Y/GOUT[2]	Y/GOUT[2]	Y/GOUT[5]	Y/GOUT[5]
DV_DATA9	Y/GOUT[1]	Y/GOUT[1]	Y/GOUT[6]	Y/GOUT[6]
DV_DATA8	Y/GOUT[0]	Y/GOUT[0]	Y/GOUT[7]	Y/GOUT[7]
DV_DATA7	Cb/BOUT[7]	Cr/ROUT[7]	Cb/BOUT[0]	Cr/ROUT[0]
DV_DATA6	Cb/BOUT[6]	Cr/ROUT[6]	Cb/BOUT[1]	Cr/ROUT[1]
DV_DATA5	Cb/BOUT[5]	Cr/ROUT[5]	Cb/BOUT[2]	Cr/ROUT[2]
DV_DATA4	Cb/BOUT[4]	Cr/ROUT[4]	Cb/BOUT[3]	Cr/ROUT[3]
DV_DATA3	Cb/BOUT[3]	Cr/ROUT[3]	Cb/BOUT[4]	Cr/ROUT[4]
DV_DATA2	Cb/BOUT[2]	Cr/ROUT[2]	Cb/BOUT[5]	Cr/ROUT[5]
DV_DATA1	Cb/BOUT[1]	Cr/ROUT[1]	Cb/BOUT[6]	Cr/ROUT[6]
DV_DATA0	Cb/BOUT[0]	Cr/ROUT[0]	Cb/BOUT[7]	Cr/ROUT[7]

(2) RGB666 Input

When the external input is of RGB666 format, the video image signal pins DV_DATA are allocated to the internal signals GOUT, BOUT, ROUT as shown in Table 35.7.

The internal signals GOUT, BOUT, ROUT to which the video image signal pins DV_DATA are allocated are output as a 24-bit video image from the RGB666 interface with the following formulae.

G[7:0] = GOUT[7:2] × 255 ÷ 63

B[7:0] = BOUT[7:2] × 255 ÷ 63

R[7:0] = ROUT[7:2] × 255 ÷ 63

Table 35.7 Bit Allocation of DV_DATA Pin Inputs When the External Input is of RGB666

INP_FORMAT[2:0]	1	1	1	1
INP_ENDIAN_ON	0	0	1	1
INP_SWAP_ON	0	1	0	1
DV_DATA17	ROUT[7]	BOUT[7]	ROUT[2]	BOUT[2]
DV_DATA16	ROUT[6]	BOUT[6]	ROUT[3]	BOUT[3]
DV_DATA15	ROUT[5]	BOUT[5]	ROUT[4]	BOUT[4]
DV_DATA14	ROUT[4]	BOUT[4]	ROUT[5]	BOUT[5]
DV_DATA13	ROUT[3]	BOUT[3]	ROUT[6]	BOUT[6]
DV_DATA12	ROUT[2]	BOUT[2]	ROUT[7]	BOUT[7]
DV_DATA11	GOUT[7]	GOUT[7]	GOUT[2]	GOUT[2]
DV_DATA10	GOUT[6]	GOUT[6]	GOUT[3]	GOUT[3]
DV_DATA9	GOUT[5]	GOUT[5]	GOUT[4]	GOUT[4]
DV_DATA8	GOUT[4]	GOUT[4]	GOUT[5]	GOUT[5]
DV_DATA7	GOUT[3]	GOUT[3]	GOUT[6]	GOUT[6]
DV_DATA6	GOUT[2]	GOUT[2]	GOUT[7]	GOUT[7]
DV_DATA5	BOUT[7]	ROUT[7]	BOUT[2]	ROUT[2]
DV_DATA4	BOUT[6]	ROUT[6]	BOUT[3]	ROUT[3]
DV_DATA3	BOUT[5]	ROUT[5]	BOUT[4]	ROUT[4]
DV_DATA2	BOUT[4]	ROUT[4]	BOUT[5]	ROUT[5]
DV_DATA1	BOUT[3]	ROUT[3]	BOUT[6]	ROUT[6]
DV_DATA0	BOUT[2]	ROUT[2]	BOUT[7]	ROUT[7]

(3) RGB565 Input

When the external input is of RGB565 format, the video image signal pins DV_DATA are allocated to the internal signals GOUT, BOUT, ROUT as shown in Table 35.8.

The internal signals GOUT, BOUT, ROUT to which the video image signal pins DV_DATA are allocated are output as a 24-bit video image from the RGB565 interface with the following formulae.

G[7:0] = GOUT[7:2] × 255 ÷ 63

B[7:0] = BOUT[7:3] × 255 ÷ 31

R[7:0] = ROUT[7:3] × 255 ÷ 31

Table 35.8 Bit Allocation of DV_DATA Pin Inputs When the External Input is of RGB565

INP_FORMAT[2:0]	2	2	2	2
INP_ENDIAN_ON	0	0	1	1
INP_SWAP_ON	0	1	0	1
DV_DATA15	ROUT[7]	BOUT[7]	ROUT[3]	BOUT[3]
DV_DATA14	ROUT[6]	BOUT[6]	ROUT[4]	BOUT[4]
DV_DATA13	ROUT[5]	BOUT[5]	ROUT[5]	BOUT[5]
DV_DATA12	ROUT[4]	BOUT[4]	ROUT[6]	BOUT[6]
DV_DATA11	ROUT[3]	BOUT[3]	ROUT[7]	BOUT[7]
DV_DATA10	GOUT[7]	GOUT[7]	GOUT[2]	GOUT[2]
DV_DATA9	GOUT[6]	GOUT[6]	GOUT[3]	GOUT[3]
DV_DATA8	GOUT[5]	GOUT[5]	GOUT[4]	GOUT[4]
DV_DATA7	GOUT[4]	GOUT[4]	GOUT[5]	GOUT[5]
DV_DATA6	GOUT[3]	GOUT[3]	GOUT[6]	GOUT[6]
DV_DATA5	GOUT[2]	GOUT[2]	GOUT[7]	GOUT[7]
DV_DATA4	BOUT[7]	ROUT[7]	BOUT[3]	ROUT[3]
DV_DATA3	BOUT[6]	ROUT[6]	BOUT[4]	ROUT[4]
DV_DATA2	BOUT[5]	ROUT[5]	BOUT[5]	ROUT[5]
DV_DATA1	BOUT[4]	ROUT[4]	BOUT[6]	ROUT[6]
DV_DATA0	BOUT[3]	ROUT[3]	BOUT[7]	ROUT[7]

(4) BT656/BT601 Input

When the external input is of BT656 or BT601 format, the video image signal pins DV_DATA are allocated to the internal signal BTOUT, as shown in Table 35.9.

The internal signal BTOUT to which the video image signal pins DV_DATA are allocated is expanded to the YCbCr signal.

For expansion to the YCbCr signal, see section 35.1.12, BT656/BT601/YCbCr422 Format Setting.

Table 35.9 Bit Allocation of DV_DATA Pin Inputs When the External Input is of BT656 or BT601

INP_FORMAT[2:0]	3, 4	3, 4
INP_ENDIAN_ON	0	1
INP_SWAP_ON	0	0
DV_DATA7	BTOUT[7]	BTOUT[0]
DV_DATA6	BTOUT[6]	BTOUT[1]
DV_DATA5	BTOUT[5]	BTOUT[2]
DV_DATA4	BTOUT[4]	BTOUT[3]
DV_DATA3	BTOUT[3]	BTOUT[4]
DV_DATA2	BTOUT[2]	BTOUT[5]
DV_DATA1	BTOUT[1]	BTOUT[6]
DV_DATA0	BTOUT[0]	BTOUT[7]

(5) YCbCr422 Input

When the external input is of YCbCr422 format, the video image signal pins DV_DATA are allocated to the internal signals Y and Cb/Cr, as shown in Table 35.10.

Table 35.10 Bit Allocation of DV_DATA Pin Inputs When the External Input is of YCbCr422

INP_FORMAT[2:0]	5	5	5	5
INP_ENDIAN_ON	0	0	1	1
INP_SWAP_ON	0	1	0	1
DV_DATA15	Y[7]	Cb/Cr[7]	Y[0]	Cb/Cr[0]
DV_DATA14	Y[6]	Cb/Cr[6]	Y[1]	Cb/Cr[1]
DV_DATA13	Y[5]	Cb/Cr[5]	Y[2]	Cb/Cr[2]
DV_DATA12	Y[4]	Cb/Cr[4]	Y[3]	Cb/Cr[3]
DV_DATA11	Y[3]	Cb/Cr[3]	Y[4]	Cb/Cr[4]
DV_DATA10	Y[2]	Cb/Cr[2]	Y[5]	Cb/Cr[5]
DV_DATA9	Y[1]	Cb/Cr[1]	Y[6]	Cb/Cr[6]
DV_DATA8	Y[0]	Cb/Cr[0]	Y[7]	Cb/Cr[7]
DV_DATA7	Cb/Cr[7]	Y[7]	Cb/Cr[0]	Y[0]
DV_DATA6	Cb/Cr[6]	Y[6]	Cb/Cr[1]	Y[1]
DV_DATA5	Cb/Cr[5]	Y[5]	Cb/Cr[2]	Y[2]
DV_DATA4	Cb/Cr[4]	Y[4]	Cb/Cr[3]	Y[3]
DV_DATA3	Cb/Cr[3]	Y[3]	Cb/Cr[4]	Y[4]
DV_DATA2	Cb/Cr[2]	Y[2]	Cb/Cr[5]	Y[5]
DV_DATA1	Cb/Cr[1]	Y[1]	Cb/Cr[6]	Y[6]
DV_DATA0	Cb/Cr[0]	Y[0]	Cb/Cr[7]	Y[7]

35.1.8 Typical Signal Timing of BT601 Format

Figure 35.3 and Figure 35.4 show the horizontal timings and Figure 35.5 and Figure 35.6 show the vertical timings of the BT601 format.

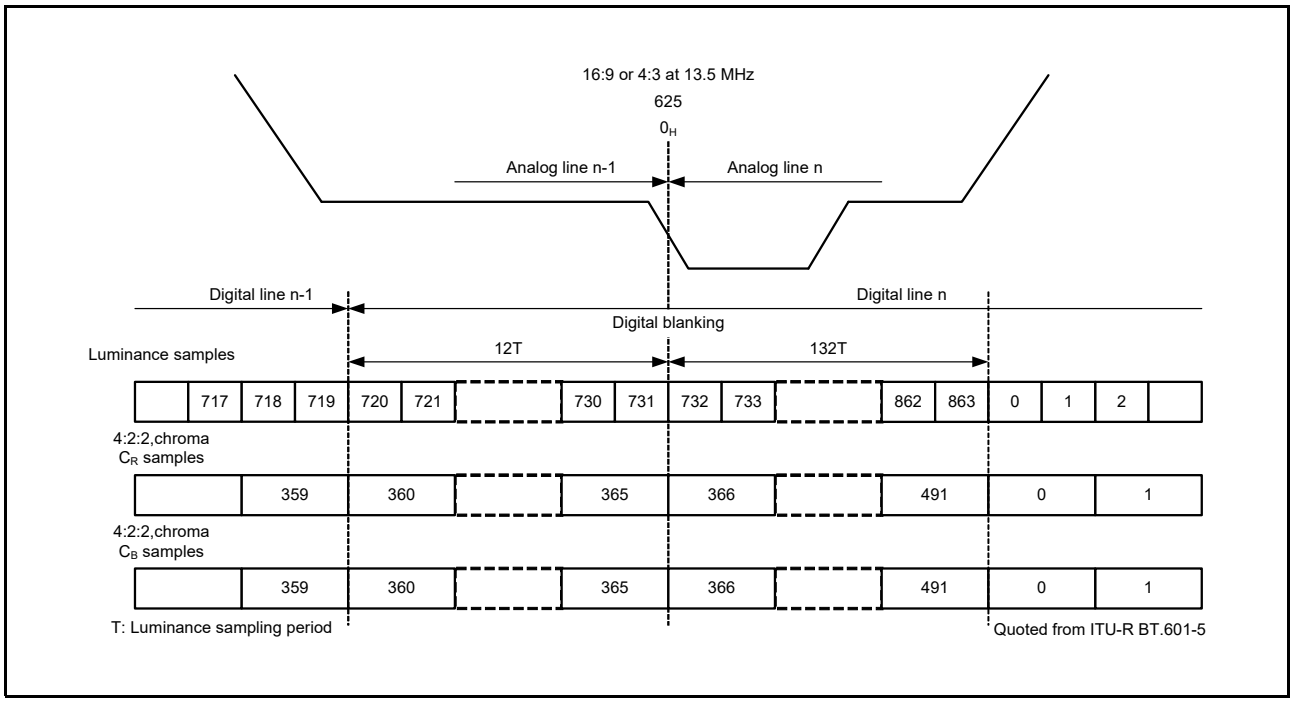


Figure 35.3 BT601 Horizontal Timing (625 Lines/50.00 Hz)

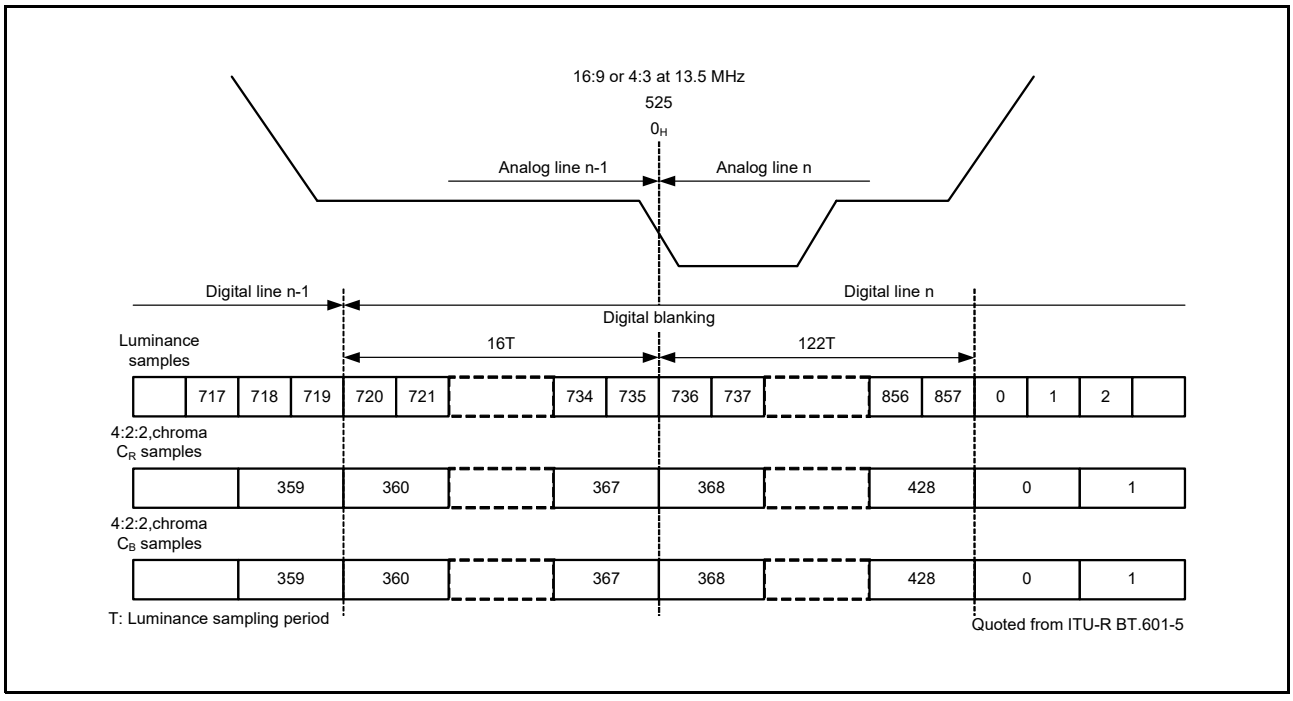


Figure 35.4 BT601 Horizontal Timing (525 Lines/59.94 Hz)

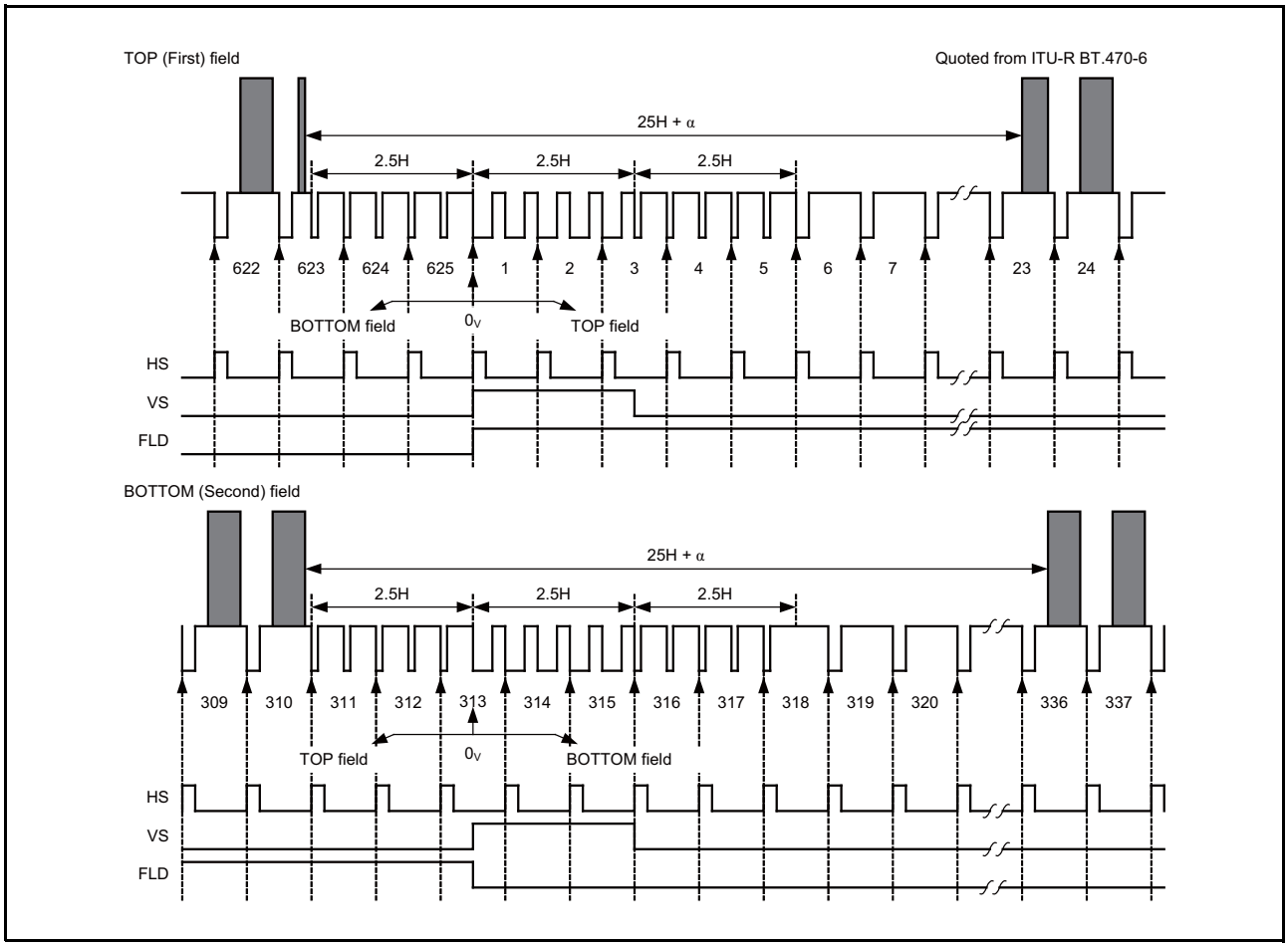


Figure 35.5 BT601 Vertical Timing (625 Lines/50.00 Hz)

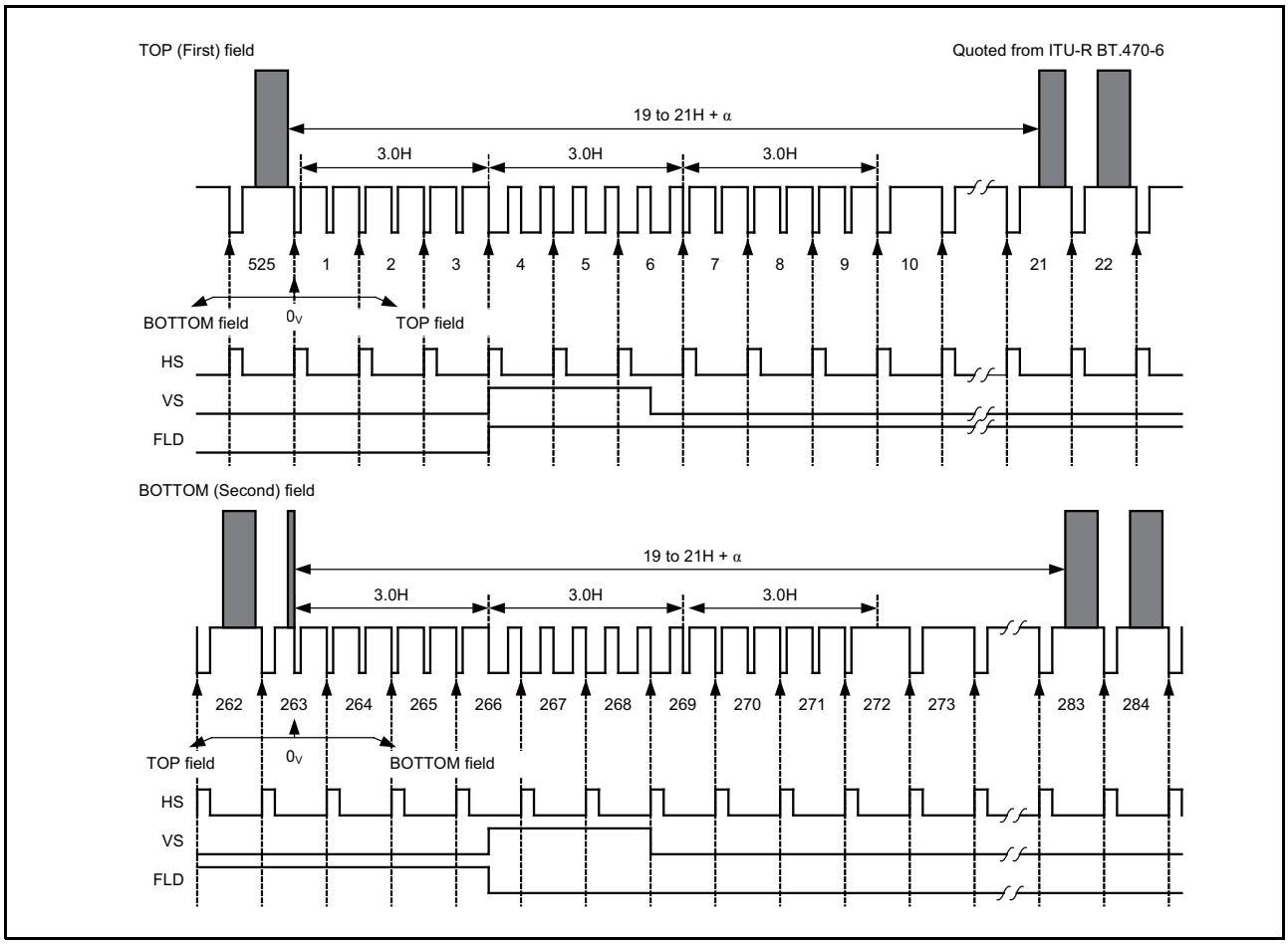


Figure 35.6 BT601 Vertical Timing (525 Lines/59.94 Hz)

35.1.9 Typical Signal Timing of BT656 Format

Figure 35.7 and Figure 35.8 show the horizontal timings of the BT656 format.

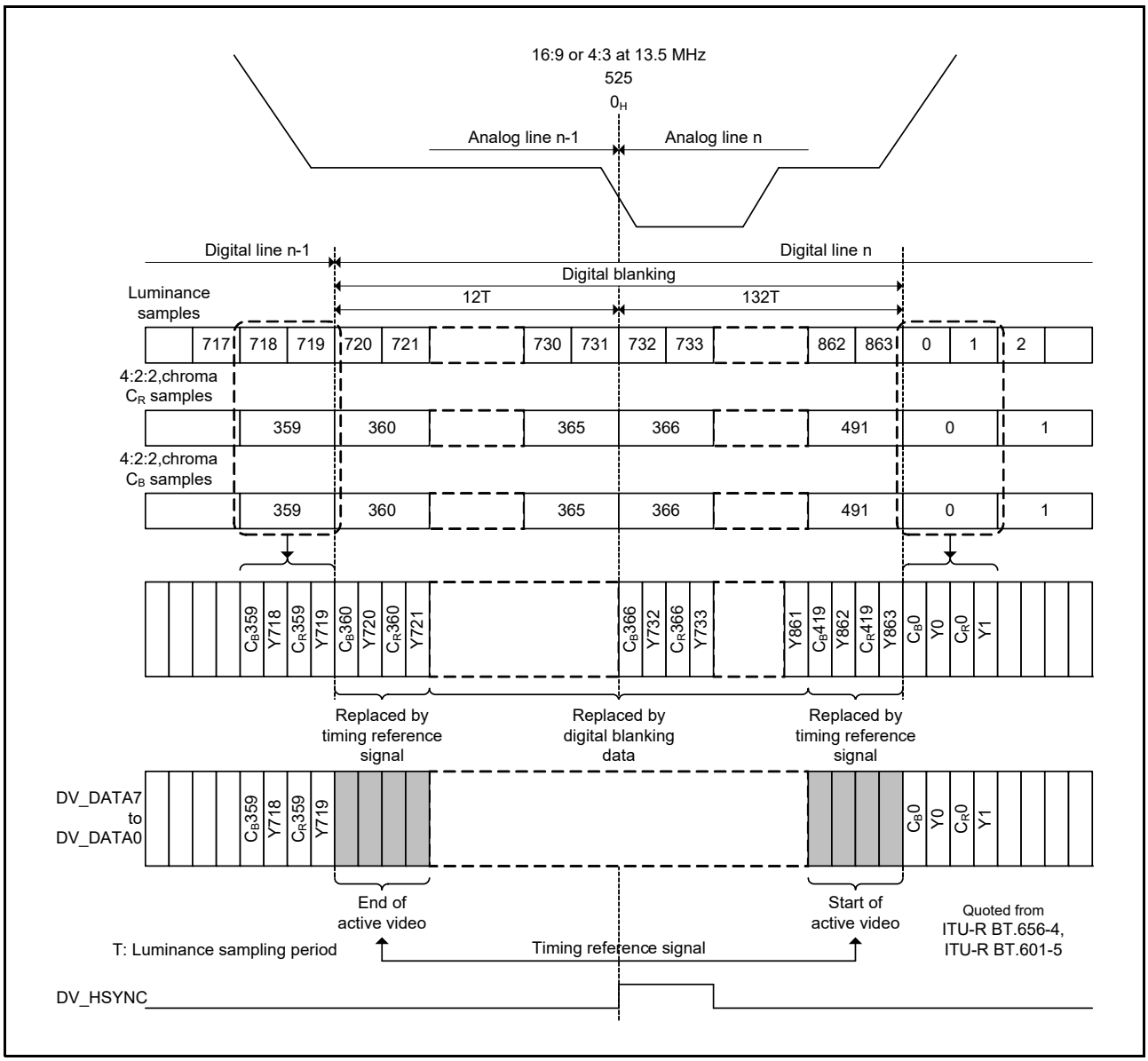


Figure 35.7 BT656 Horizontal Timing (625 Lines/50.00 Hz)

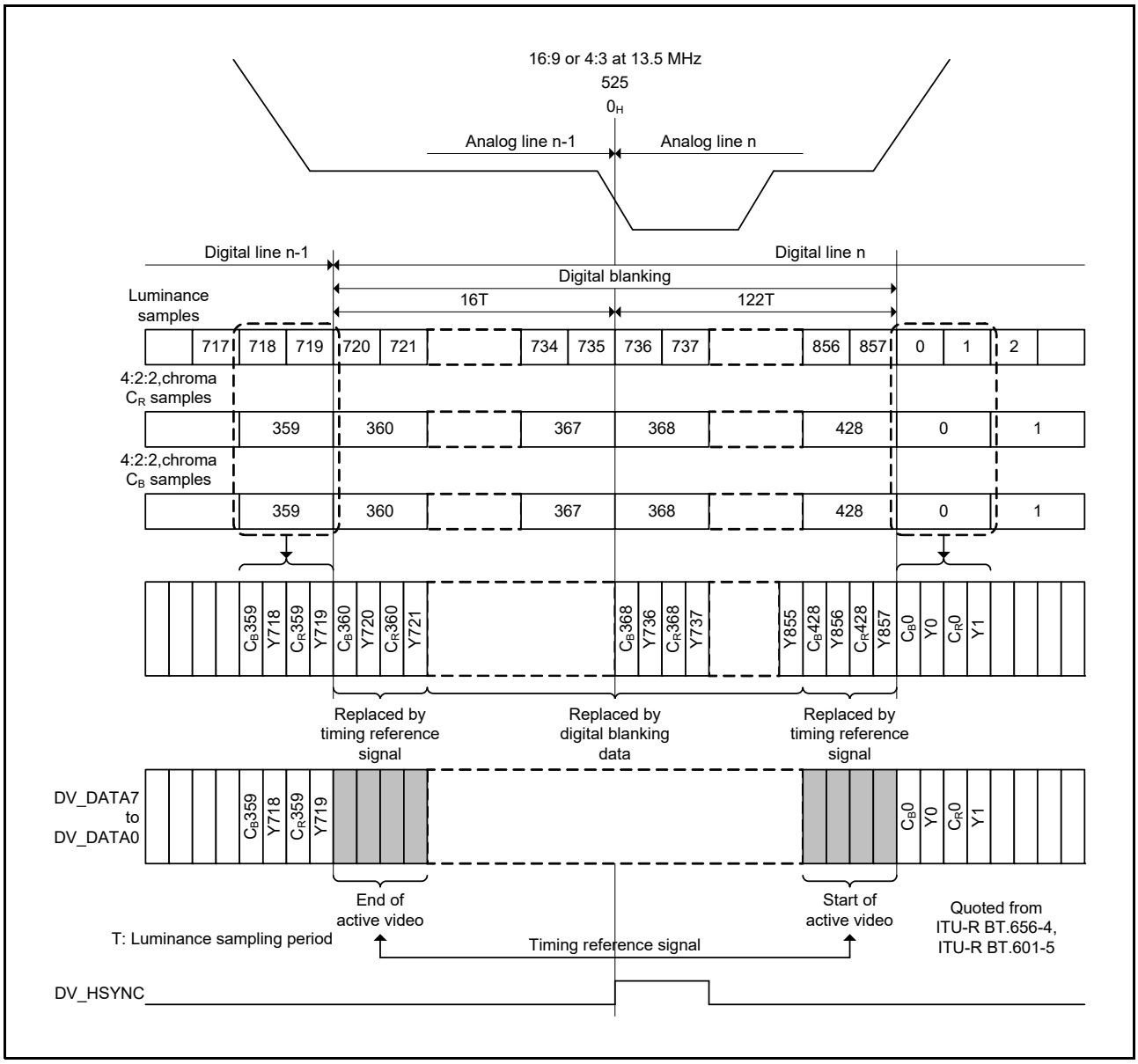


Figure 35.8 BT656 Horizontal Timing (525 Lines/59.94 Hz)

35.1.10 SAV/EAV Code in BT656 Format

Table 35.11 shows the timing of inserting the SAV/EAV code in the BT656 format. Bit information is shown in Table 35.12 and Table 35.13. This module does not refer to the parity bits P3 to P0 shown in Table 35.13.

Table 35.11 SAV/EAV Code Insertion Timing (Line)

		625	525
V-digital field blanking			
Field 1	Start (V = 1)	Line 624	Line 1
	Finish (V = 0)	Line 23	Line 20
Field 2	Start (V = 1)	Line 311	Line 264
	Finish (V = 0)	Line 336	Line 283
V-digital field blanking			
Field 1	F = 0	Line 1	Line 4
Field 2	F = 1	Line 313	Line 266

Table 35.12 SAV/EAV Code Bit Information (1)

Data Bit Number	1st Word (FF)	2nd Word (00)	3rd Word (00)	4th Word (XY)
7 (MSB)	1	0	0	1
6	1	0	0	F
5	1	0	0	V
4	1	0	0	H
3	1	0	0	P3
2	1	0	0	P2
1	1	0	0	P1
0	1	0	0	P0

[Legend]
F = 0 during field 1
F = 1 during field 2
V = 0 elsewhere
V = 1 during field blanking
H = 0 is SAV
H = 1 is EAV

Table 35.13 SAV/EAV Code Bit Information (2)

F	V	H	P3	P2	P1	P0
0	0	0	0	0	0	0
0	0	1	1	1	0	1
0	1	0	1	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	1	1
1	0	1	1	0	1	0
1	1	0	1	1	0	0
1	1	1	0	0	0	1

Figure 35.9 and Figure 35.10 show the SAV/EAV code tables.

		One Horizontal Period																
		EAV				H blank	SAV				Valid area							
		1	2	3	4		285	286	287	288	289	290	291	292	...	1725	1726	1727	1728
Field1 (top)	1	FF	00	00	B6		FF	00	00	AB	Digital Blanking Data								
	:	FF	00	00	B6		FF	00	00	AB									
	22	FF	00	00	B6		FF	00	00	AB									
	23	FF	00	00	9D		FF	00	00	80	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719
	:	FF	00	00	9D		FF	00	00	80	:	Valid pixel data area							:
	:	FF	00	00	9D		FF	00	00	80	:								:
	:	FF	00	00	9D		FF	00	00	80	:								:
	:	FF	00	00	9D		FF	00	00	80	:								:
	:	FF	00	00	9D		FF	00	00	80	:								:
	310	FF	00	00	9D		FF	00	00	80	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719
	311	FF	00	00	B6		FF	00	00	AB	Digital Blanking Data								
	312	FF	00	00	B6		FF	00	00	AB									
Field2 (bottom)	313	FF	00	00	F1		FF	00	00	EC	Digital Blanking Data								
	:	FF	00	00	F1		FF	00	00	EC									
	335	FF	00	00	F1		FF	00	00	EC									
	336	FF	00	00	DA		FF	00	00	C7	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719
	:	FF	00	00	DA		FF	00	00	C7	:	Valid pixel data area							:
	:	FF	00	00	DA		FF	00	00	C7	:								:
	:	FF	00	00	DA		FF	00	00	C7	:								:
	:	FF	00	00	DA		FF	00	00	C7	:								:
	:	FF	00	00	DA		FF	00	00	C7	:								:
	623	FF	00	00	DA		FF	00	00	C7	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719
	624	FF	00	00	F1		FF	00	00	EC	Digital Blanking Data								
	625	FF	00	00	F1		FF	00	00	EC									

Figure 35.9 SAV/EAV Code in BT656 Format (625 Lines/50.00 Hz)

		One Horizontal Period																	
		EAV				H blank	SAV				Valid area								
		1	2	3	4		273	274	275	276	277	278	279	280	...	1713	1714	1715	1716
Field2	1	FF	00	00	F1		FF	00	00	EC	Digital Blanking Data								
	2	FF	00	00	F1		FF	00	00	EC									
	3	FF	00	00	F1		FF	00	00	EC									
Field1 (top)	4	FF	00	00	B6		FF	00	00	AB	Digital Blanking Data								
	:	FF	00	00	B6		FF	00	00	AB									
	19	FF	00	00	B6		FF	00	00	AB									
	20	FF	00	00	9D		FF	00	00	80	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719
	:	FF	00	00	9D		FF	00	00	80	:								:
	:	FF	00	00	9D		FF	00	00	80	:								:
	:	FF	00	00	9D		FF	00	00	80	:	Valid pixel data area							:
	:	FF	00	00	9D		FF	00	00	80	:								:
	:	FF	00	00	9D		FF	00	00	80	:								:
	263	FF	00	00	9D		FF	00	00	80	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719
	264	FF	00	00	B6		FF	00	00	AB	Digital Blanking Data								
	265	FF	00	00	B6		FF	00	00	AB									
Field2 (bottom)	266	FF	00	00	F1		FF	00	00	EC	Digital Blanking Data								
	:	FF	00	00	F1		FF	00	00	EC									
	282	FF	00	00	F1		FF	00	00	EC									
	283	FF	00	00	DA		FF	00	00	C7	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719
	:	FF	00	00	DA		FF	00	00	C7	:								:
	:	FF	00	00	DA		FF	00	00	C7	:								:
	:	FF	00	00	DA		FF	00	00	C7	:	Valid pixel data area							:
	:	FF	00	00	DA		FF	00	00	C7	:								:
	:	FF	00	00	DA		FF	00	00	C7	:								:
	525	FF	00	00	DA		FF	00	00	C7	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719

Figure 35.10 SAV/EAV Code in BT656 Format (525 Lines/59.94 Hz)

35.1.11 BT656 Progressive Format

This product can be connected with devices which output data in the BT656 progressive format. Because the standard for the BT656 format does not include description of output in the progressive format, there is no guarantee that this product is connected with devices which output data in the progressive format. The following description shows how to generate a vertical/horizontal synchronization signal by decoding the SAV/EAV code input via the BT656 interface of this module. Confirm the connection with devises which output data in the BT656 progressive format in accordance with this section.

(1) SAV/EAV Code

The SAV/EAV code consists of four words. When the first word is set to FF and the second and third words are set to 00, timing signals are generated by decoding the value of the fourth word (XY).

For bit information, see Table 35.12 in section 35.1.10, SAV/EAV Code in BT656 Format. This product does not refer to the parity bits (P3 to P0).

(2) Vertical/Horizontal Synchronization Signal

Based on the SAV/EAV code, the vertical/horizontal synchronization signal is generated.

(a) Vertical Synchronization Signal

The vertical synchronization signal is output when the value of the V bit is changed from 0 to 1 in the BT656 format. The timing of the output varies with the setting of INP_EXT_SYNC_CNT.INP_F525_625 setting and the value of the F bit in the BT656 format.

Table 35.14 lists the timing.

Table 35.14 Timing of Delay for Output of Vertical Synchronization Signal

INP_EXT_SYNC_CNT. INP_F525_625	F Bit in BT656 Format	Output Timing	Remark
0: 525 lines	0 (Field 1)	2.5 lines after setting of V bit to 1 is detected	525 lines, vertical synchronization signal for field 2
	1 (Field 2)	3 lines after setting of V bit to 1 is detected	525 lines, vertical synchronization signal for field 1
1: 625 lines	0 (Field 1)	2.5 lines after setting of V bit to 1 is detected	625 lines, vertical synchronization signal for field 2
	1 (Field 2)	2 lines after setting of V bit to 1 is detected	625 lines, vertical synchronization signal for field 1

(b) Horizontal Synchronization Signal

Based on the setting of the INP_EXT_SYNC_CNT.INP_H_EDGE_SEL bit, the horizontal synchronization signal is output.

(c) Timing Example of 525-Line Interface Input in BT656 Format

Figure 35.11 and Figure 35.12 show examples of the timing of vertical/horizontal synchronization signal extracted from 525-line interlaced input in the BT656 format.

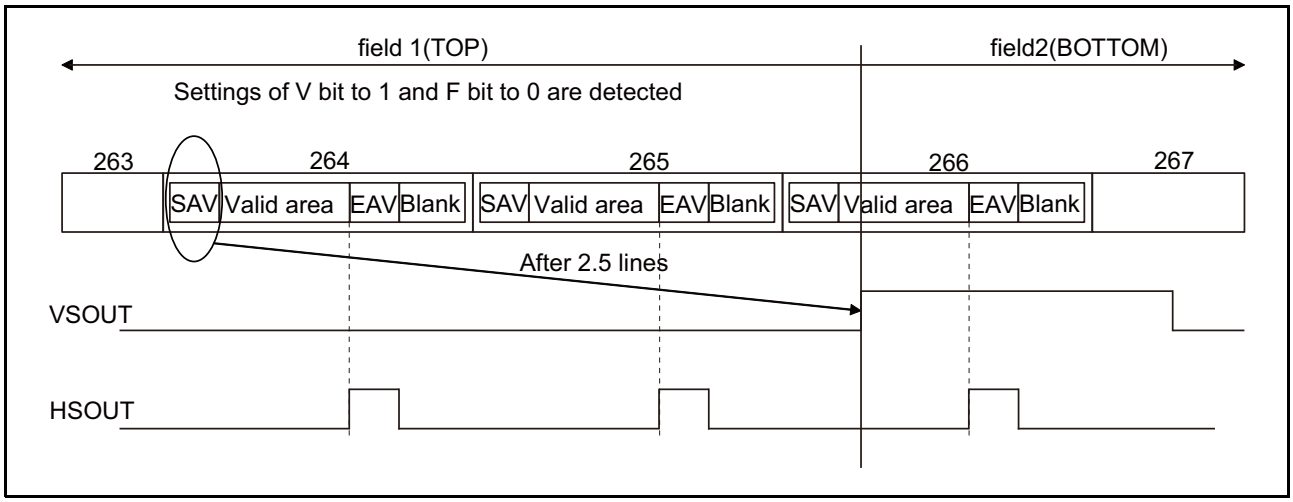


Figure 35.11 Timing of Vertical/Horizontal Synchronization Signal from 525-Line Interlaced Input in BT656 Format (Top to Bottom)

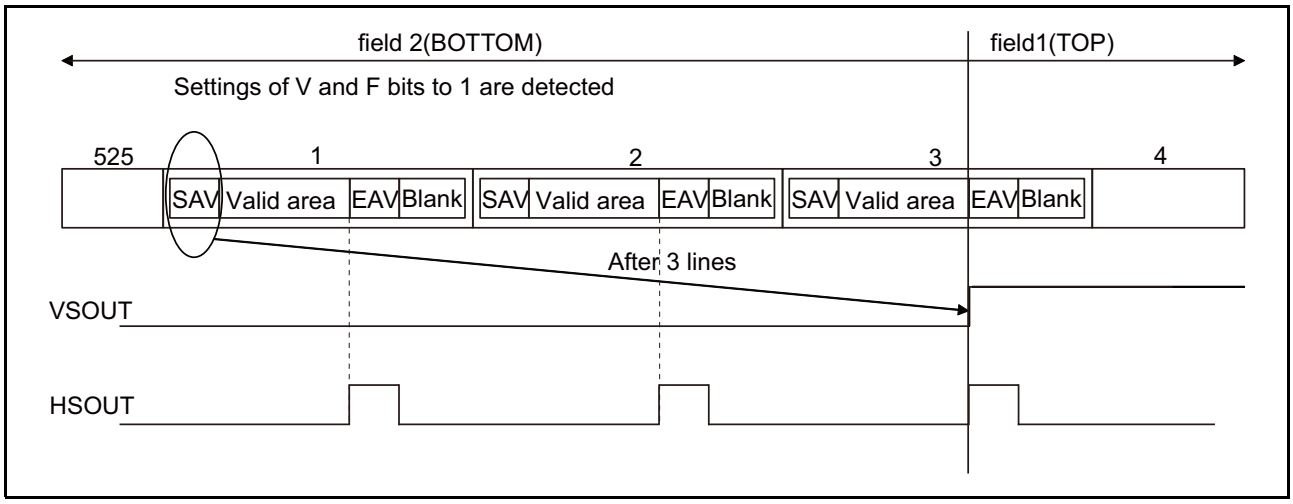


Figure 35.12 Timing of Vertical/Horizontal Synchronization Signal from 525-Line Interlaced Input in BT656 Format (Bottom to Top)

(3) Example of Timing for Progressive Input in BT656 Format

Figure 35.13 shows an example of the SAV/EAV code in 525-line progressive input in the BT656 format. Figure 35.14 shows the vertical/horizontal synchronization signal extracted from 525-line progressive input in the BT656 format. The field is detected as field 1 in this example, because the value of the F bit is set to 0 when that of the V bit is changed from 0 to 1. The filed is regarded as the bottom field. The vertical synchronization signal is output 2.5 lines after the detection of the SAV code.

		One Horizontal Period																
		EAV				H blank	SAV				Valid area							
		1	2	3	4		273	274	275	276	277	278	279	280	...	1713	1714	1715	1716
Field1 (top)	1	FF	00	00	BX		FF	00	00	AX	Digital Blanking Data								
	:	FF	00	00	BX		FF	00	00	AX									
	19	FF	00	00	BX		FF	00	00	AX									
	20	FF	00	00	9X		FF	00	00	8X	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719
	:	FF	00	00	9X		FF	00	00	8X	:	Valid pixel data area							:
	:	FF	00	00	9X		FF	00	00	8X	:								:
	:	FF	00	00	9X		FF	00	00	8X	:								:
	:	FF	00	00	9X		FF	00	00	8X	:								:
	:	FF	00	00	9X		FF	00	00	8X	:								:
	504	FF	00	00	9X		FF	00	00	8X	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719
	505	FF	00	00	BX		FF	00	00	AX	Digital Blanking Data								
	:	FF	00	00	BX		FF	00	00	AX									
	525	FF	00	00	BX		FF	00	00	AX									

Figure 35.13 SAV/EAV Code in BT656 Progressive Format (525 Lines, 59.94 Hz)

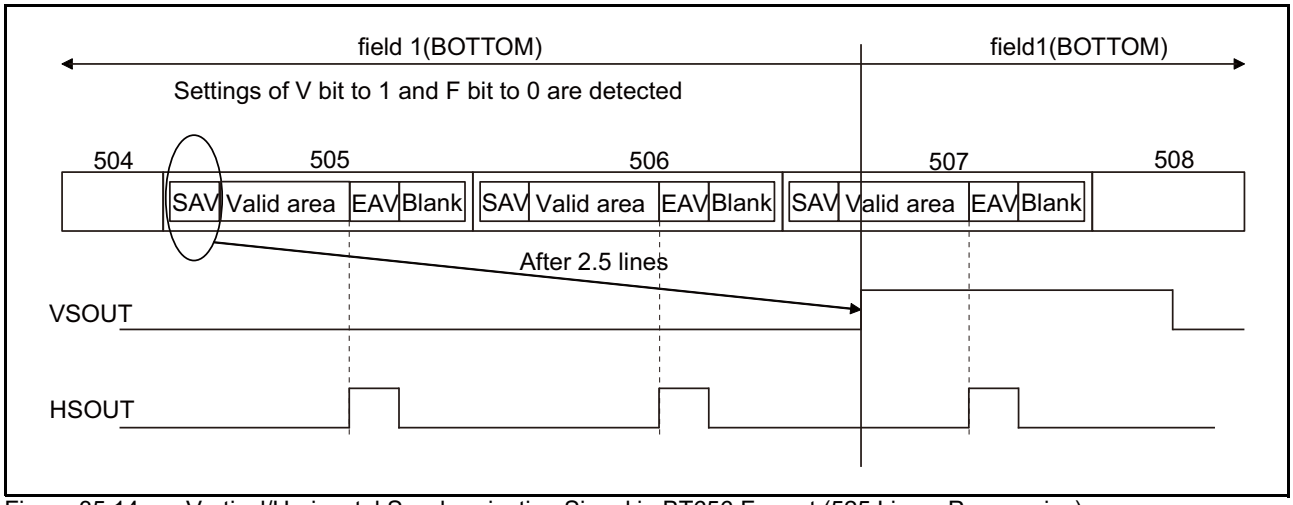


Figure 35.14 Vertical/Horizontal Synchronization Signal in BT656 Format (525 Lines, Progressive)

35.1.12 BT656/BT601/YCbCr422 Format Setting

The BT656 format can be used for the 525-line and 59.94-Hz or the 625-line and 50.00-Hz interlace signal format and progressive signal format (extended).

The BT601 format can be used for the 525-line and 59.94-Hz or the 625-line and 50.00-Hz interlace signal format and progressive signal format (extended).

The YCbCr422 format can be used for the 525-line and 59.94-Hz or the 625-line and 50.00-Hz interlace signal format in the 16-bit data-bus format of the BT601 standard.

The Vsync signal timing for the 525-line BT656 format and 625-line BT656 format are different.

The operating mode is set by the INP_F525_625 bit.

Table 35.15 Operating Mode Setting for BT656 Format

Register Name	Bit Name	Initial Value	Description
INP_EXT_SYNC_CNT	INP_F525_625	0	Number of Lines for BT656 Input of External Input System 0: 525 lines 1: 625 lines

When the interlace signals are to be input in BT656/BT601/YCbCr422 format, half of 2fH phase timings of the Vsync signal and the Hsync signal are set with the INP_FH50[11:0] bits.

The INP_FH50[11:0] bits are also used for the vertical synchronous phase adjustment block. Therefore, for bit description, see Table 35.20.

When the external input is of BT656 format, the reference point of the Hsync signal is set with the INP_H_EDGE_SEL bit.

Table 35.16 Hsync Signal Reference Selection for BT656 Format

Register Name	Bit Name	Initial Value	Description
INP_EXT_SYNC_CNT	INP_H_EDGE_SEL	0	Hsync Signal Reference Select for BT656 Format of External Input System 0: EAV 1: SAV

When the external input is of BT656/BT601 format, the internal signal BTOUT[7:0], which is input from the DV_DATA pins and allocated, is expanded to the 24-bit YCbCr signal.

Expansion timing with respect to the Hsync signal reference is set with the INP_H_POS[1:0] bits.

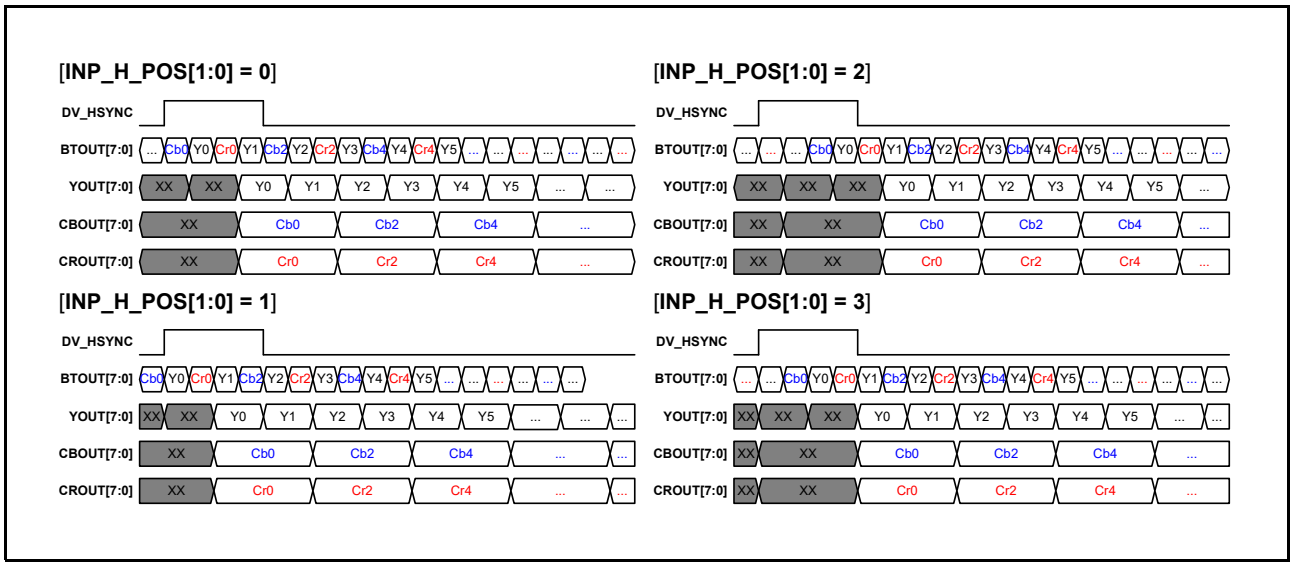


Figure 35.15 YCbCr Data Expansion for BT656/BT601 Input

Table 35.17 Data String Start Timing Selection for BT656/BT601 Input

Register Name	Bit Name	Initial Value	Description
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	0	Y/Cb/Y/Cr Data String Start Timing with respect to Hsync Reference 0: Cb/Y/Cr/Y 1: Y/Cr/Y/Cb 2: Cr/Y/Cb/Y 3: Y/Cb/Y/Cr

When the external input is in YCbCr422 format, the input from the DV_DATA pins is allocated to the internal Y[7:0] and CbCr[7:0] signals, and the CbCr[7:0] are expanded to a 16-bit signal.

Expansion timing with respect to the Hsync signal reference is set with the INP_H_POS[1:0] bits.

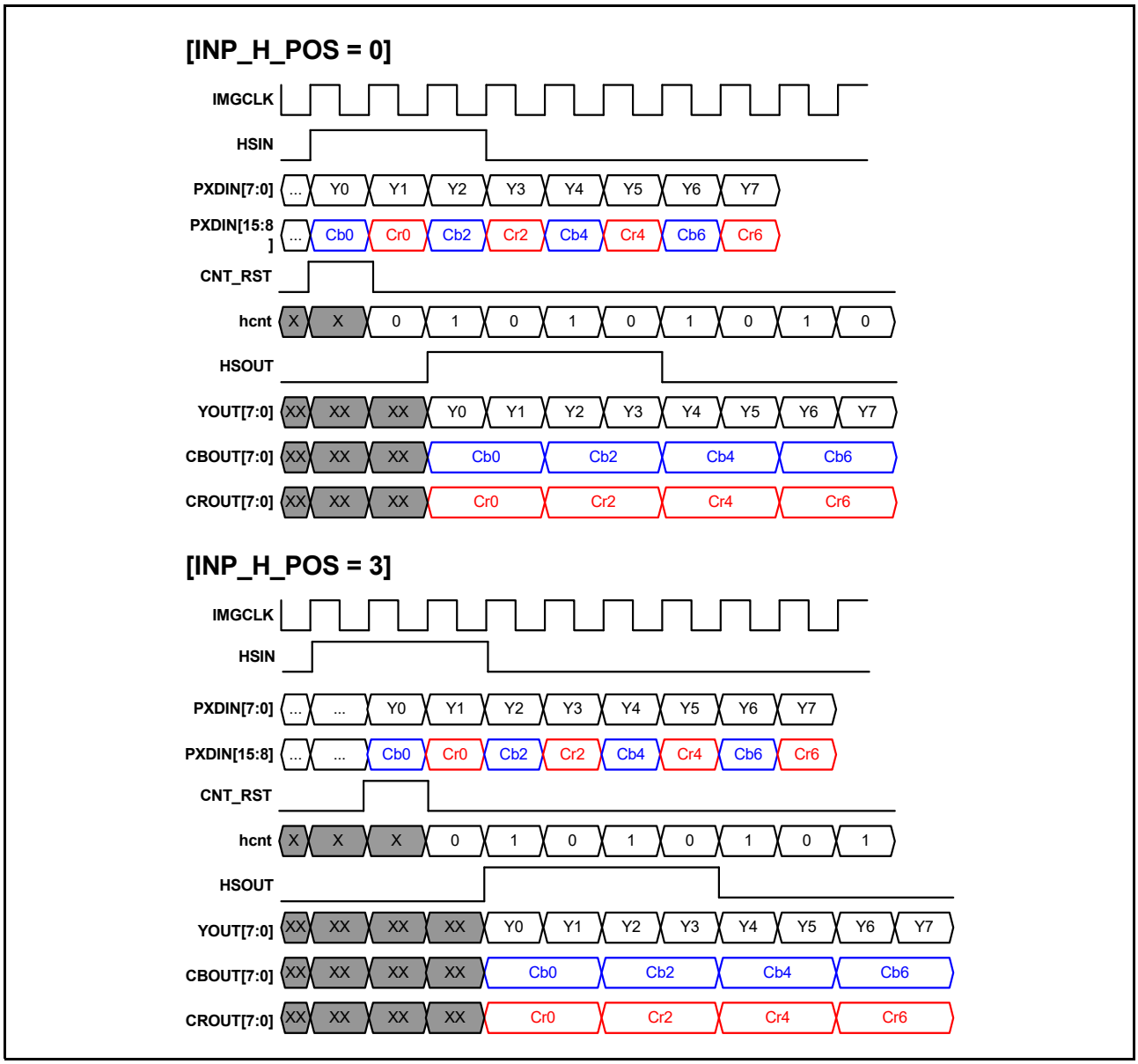


Figure 35.16 YCbCr Data Expansion for YCbCr422 Input

Table 35.18 Data String Start Timing Selection for YCbCr422 Input

Register Name	Bit Name	Initial Value	Description
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	0	Cb/Cr Data String Start Timing with respect to Hsync Reference 0: Cb/Cr 3: Cr/Cb 1, 2: Setting prohibited

35.1.13 YCbCr444/RBG888/666/565 Input Timing

The YCbCr444/RGB888/666/565 format can be used for the progressive YCbCr/RGB signal.

The sync signal width (H_SYNC, V_SYNC), sync signal polarity (H_POL, V_POL), valid period start position (H_BP, V_BP), valid period end position (H_FP, V_FP), and valid period video width (H_ACTIVE, V_ACTIVE) are shown in Table 35.19.

Table 35.19 YCbCr/RGB Signal Reception Timing

Item	Description
External input clock	Maximum external input clock frequency: 87.00 MHz
Vsync signal width (V_SYNC)	Minimum Vsync signal width: 1 CLK
Vsync signal polarity (V_POL)	Positive or negative polarity is selected by the relevant registers.
Vertical valid period start position (V_BP)	From Vsync reference to the head of the video image: 5 lines or more
Vertical valid period video width (V_ACTIVE)	Maximum vertical valid period: 1080lines
Vertical valid period end position (V_FP)	From the end of the video image to the Vsync reference: 4 lines or more *1
Hsync signal width (H_SYNC)	Minimum Hsync signal width: 1 CLK
Hsync signal polarity (H_POL)	Positive or negative polarity is selected by the relevant registers.
Horizontal valid period start position (H_BP)	From Hsync reference to the head of the video image: 16 CLK or more
Horizontal valid period video width (H_ACTIVE)	Maximum horizontal valid period: 1920 pixels
Horizontal valid period end position (H_FP)	From the end of the video image to the Hsync reference: 16 CLK or more *2
Number of vertical lines (V_BP+V_ACTIVE+V_FP)	Between vertical synchronization signals: 2047 lines or less
Number of horizontal pixels (H_BP+H_ACTIVE+H_FP)	Between horizontal synchronization signals: 4095 CLK or less

Note 1. When V_FP is below four lines, the setting of INP_DLY_ADJ.INP_VS_DLY_L[2:0] should be adjusted so that V_FP is at least four lines.

Note 2. When H_FP is below 16 cycles, the settings of INP_DLY_ADJ.INP_VS_DLY[7:0], INP_HS_DLY[7:0], and INP_FLD_DLY[7:0] should be adjusted so that H_FP is at least 16 cycles of the pixel clock.

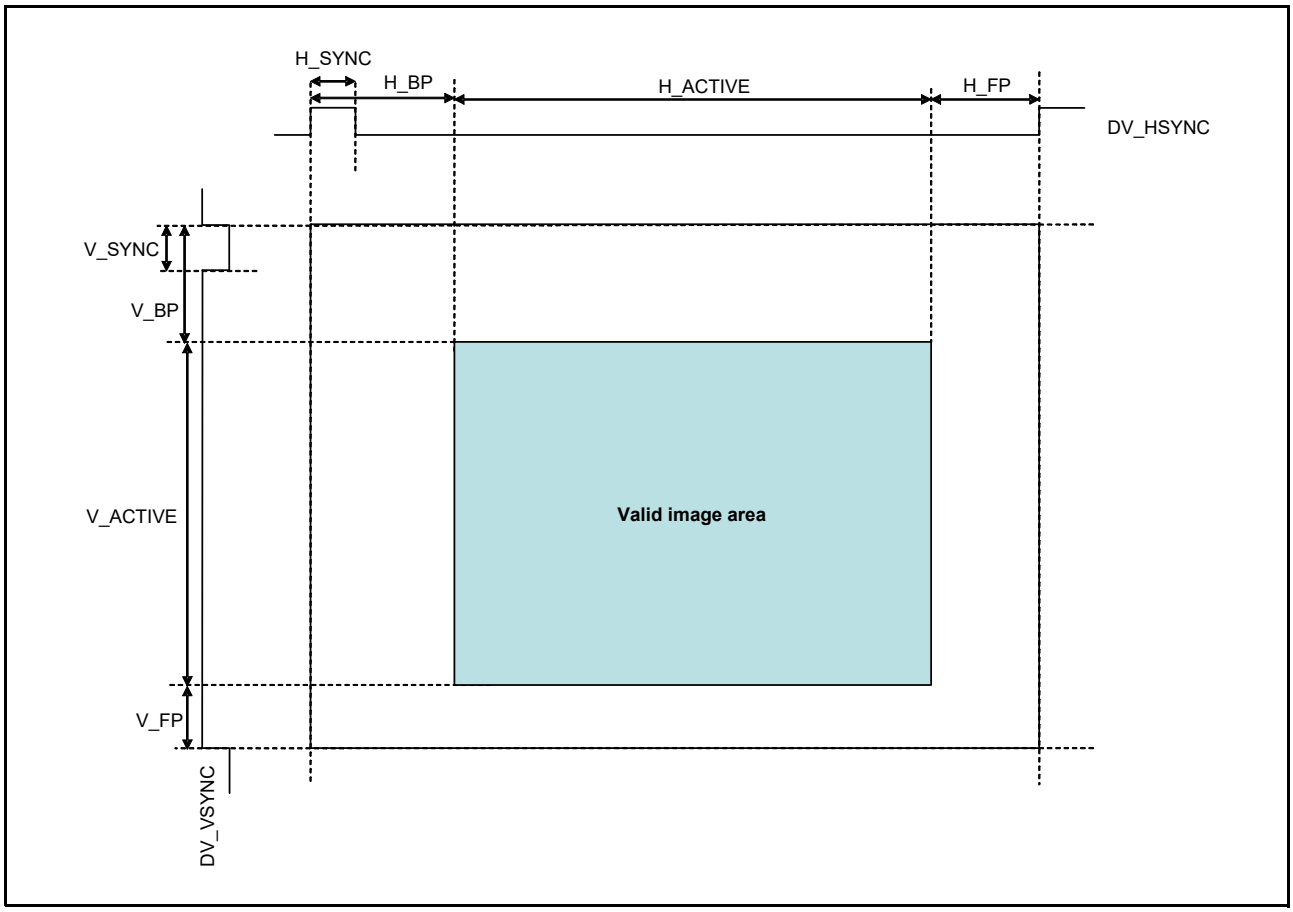


Figure 35.17 YCbCr/RGB Signal Reception Timing

35.1.14 Field Differentiation and Vsync Signal Phase Adjustment

The phase of the input Vsync signal and Hsync signal is detected and the field of the interlace signal is determined. When the reference point of the Vsync signal is detected within ±0.5 horizontal period with respect to the Hsync signal, it is determined as the interlace top field. When the reference point of the Vsync signal is detected outside ±0.5 horizontal period with respect to the Hsync signal, it is determined as the interlace bottom field.

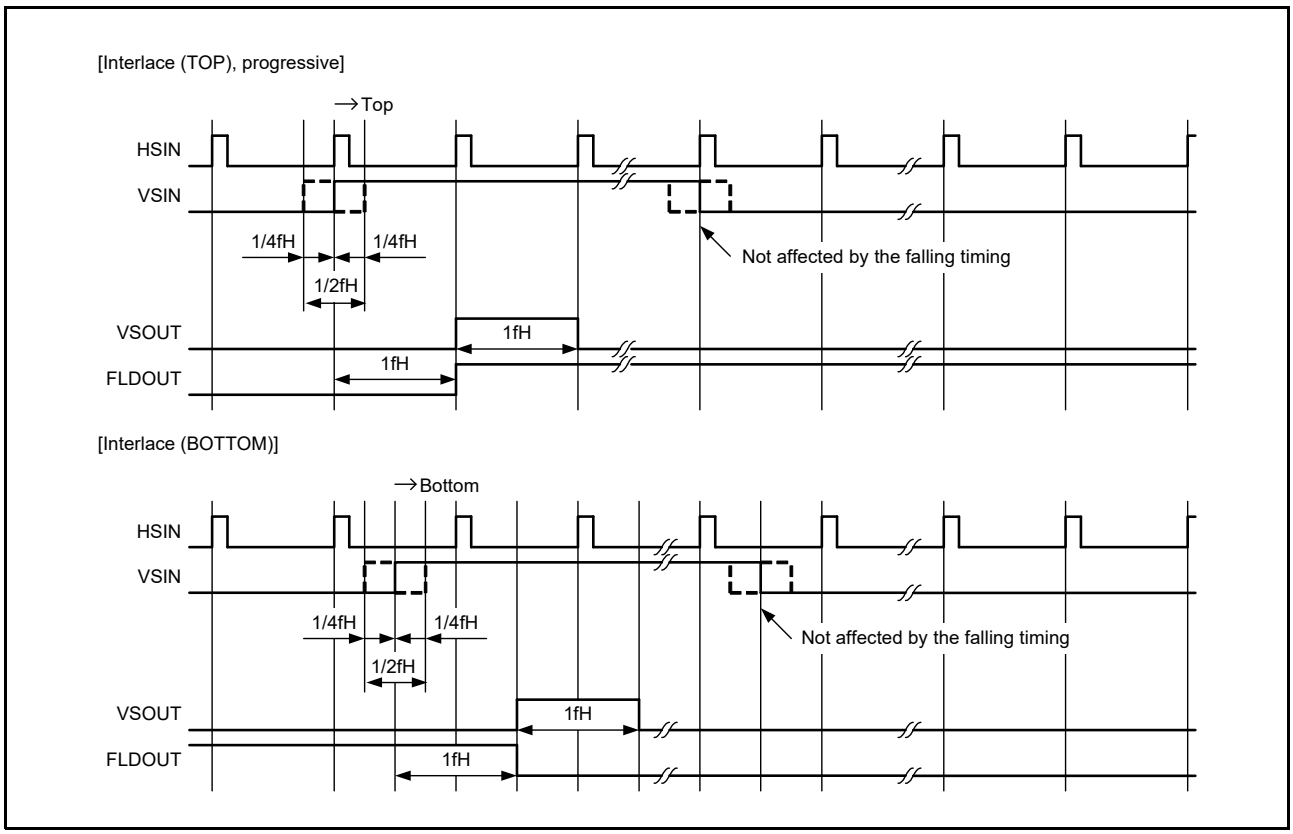


Figure 35.18 Vsync Signal Phase Adjustment

The timings of 1/2fH Vsync signal phase and 1/4fH Vsync signal phase are set with INP_FH50[11:0] and INP_FH25[11:0], respectively.

Table 35.20 Vsync Signal Phase Timing Setting

Register Name	Bit Name	Initial Value	Description
INP_VSYNC_PH_ADJ	INP_FH50[11:0]	858	Vsync Signal 1/2fH Phase Timing Should be 1/2 the horizontal cycle.
INP_VSYNC_PH_ADJ	INP_FH25[11:0]	429	Vsync Signal 1/4fH Phase Timing Should be 1/4 the horizontal cycle.

35.1.15 Vsync Signal Delay Adjustment in Line Units

The Vsync signal line delay adjust block can delay the Vsync signal and the field differentiation signal in line units.

When a video signal with a short vertical front porch is input, the vertical front porch is adjusted.

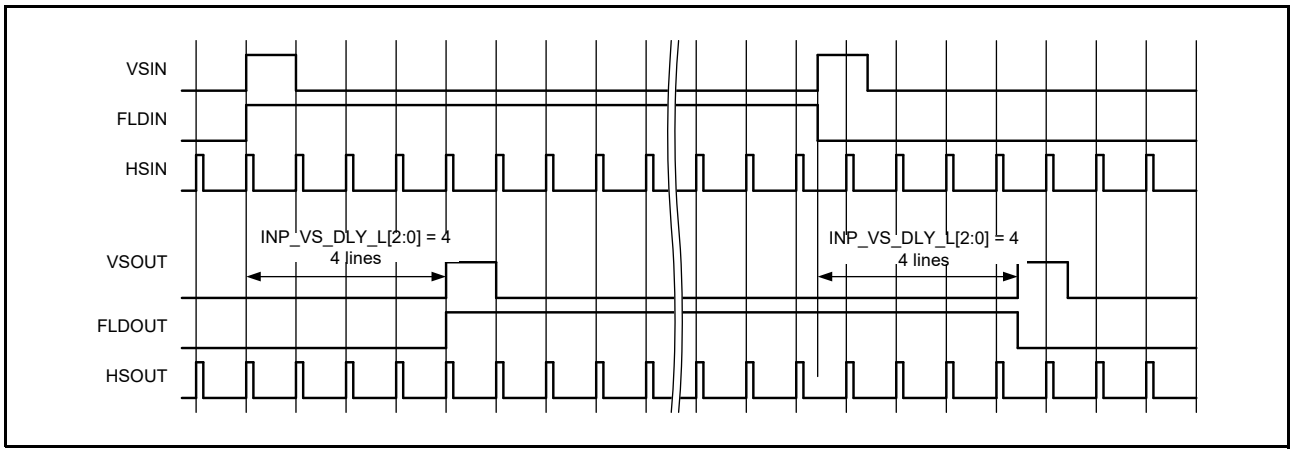


Figure 35.19 Timing of Vsync Signal Delay in Line Units

Table 35.21 Adjustment of Vsync Signal Delay in Line Units

Register Name	Bit Name	Initial Value	Description
INP_DLY_ADJ	INP_VS_DLY_L[2:0]	0	Number of Lines for Delaying Vsync Signal and Field Differentiation Signal Delay amount: 0 to 7 (lines)

35.1.16 Sync Signal Delay Adjustment

Delay can be adjusted independently for the Vsync signal, Hsync signal, and field differentiation signal in the units of clock.

Lacking margin of the horizontal front porch is adjusted according to the input synchronization disturbance.

Table 35.22 Sync Signal Delay Adjustment

Register Name	Bit Name	Initial Value	Description
INP_DLY_ADJ	INP_VS_DLY[7:0]	0	Vsync Signal Delay Amount Delay amount: 0 to 254 (clock cycles)
INP_DLY_ADJ	INP_HS_DLY[7:0]	0	Hsync Signal Delay Amount Delay amount: 0 to 254 (clock cycles)
INP_DLY_ADJ	INP_FLD_DLY[7:0]	0	Field Differentiation Signal Delay Amount Delay amount: 0 to 254 (clock cycles)

35.1.17 Horizontal Noise Reduction

Noise can be reduced according to horizontal pixel reference.

Noise reduction is controlled through noise component frequency band (TAP), noise level (threshold), and noise reduction intensity (gain).

(1) Frequency Band (TAP) Setting for Noise Component

The noise frequency band can be selected independently from the following four types by using the NR1D_Y_TAP[1:0], NR1D_CB_TAP[1:0], and NR1D_CR_TAP[1:0] bits.

When the number of adjacent pixels is one (noise reduction NR1D_Y/CB/CR_TAP is 0):

$$BPF_{(1)} = \frac{1}{4}(-1 \times Z_{(-1)}, 2 \times Z_{(0)}, -1 \times Z_{(+1)})$$

When the number of adjacent pixels is two (noise reduction NR1D_Y/CB/CR_TAP is 1):

$$BPF_{(2)} = \frac{1}{4}(-1 \times Z_{(-2)}, 2 \times Z_{(0)}, -1 \times Z_{(+2)})$$

When the number of adjacent pixels is three (noise reduction NR1D_Y/CB/CR_TAP is 2):

$$BPF_{(3)} = \frac{1}{4}(-1 \times Z_{(-3)}, 2 \times Z_{(0)}, -1 \times Z_{(+3)})$$

When the number of adjacent pixels is four (noise reduction NR1D_Y/CB/CR_TAP is 3):

$$BPF_{(4)} = \frac{1}{4}(-1 \times Z_{(-4)}, 2 \times Z_{(0)}, -1 \times Z_{(+4)})$$

Note: $Z_{(0)}$ indicates the target pixel for noise reduction and $Z_{(N)}$ indicates the pixel that is n pixels off from $Z_{(0)}$ In the horizontal direction.

(2) Setting Noise Level (Threshold)

The absolute value of the detected noise amount (BPF output value) is compared with the values of the NR1D_Y_TH[6:0], NR1D_CB_TH[6:0], and NR1D_CR_TH[6:0] bits. When the detected noise amount is greater than NR1D_Y/CB/CR_TH, the absolute value of the detected noise amount is considered as NR1D_Y/CB/CR_TH (fixed value).

ABS(BPF(n)) ≤ absolute value of detected noise amount when ABS(BPF(n)) ≤ NR1D_Y/CB/CR_TH:
NOISE_ABS = ABS(BPF(n))
ABS(BPF(n)) > absolute value of detected noise amount when ABS(BPF(n)) > NR1D_Y/CB/CR_TH:
NOISE_ABS = NR1D_Y/CB/CR_TH

(3) Setting Noise Reduction Intensity (Gain)

The absolute value of the detected noise amount is multiplied by the value of gain specified by the NR1D_Y_GAIN[1:0], NR1D_CB_GAIN[1:0], and NR1D_CR_GAIN[1:0] bits, and the feedback is calculated for the original signal.

Computation when the amount of detected noise (BPF(n)) is negative (−):
$DOUT = DIN + NOISE\_ABS \div 2^{(NR1D\_Y/CB/CR\_GAIN+1)}$
Computation when the amount of detected noise (BPF(n)) is positive (+):
$DOUT = DIN - NOISE\_ABS \div 2^{(NR1D\_Y/CB/CR\_GAIN+1)}$

Table 35.23 Horizontal Noise Reduction

Register Name	Bit Name	Initial Value	Description
IMGCNT_NR_CNT0	NR1D_MD	1	Horizontal Noise Reduction Operating Mode 0: R/G/B mode 1: Y/Cb/Cr mode
IMGCNT_NR_CNT0	NR1D_ON	0	Noise Reduction On/Off Control 0: Noise Reduction Off 1: Noise Reduction On
IMGCNT_NR_CNT0	NR1D_Y_TAP[1:0]	0	Y/G Signal TAP Select 0: Adjacent pixel 1: 2 adjacent pixels 2: 3 adjacent pixels 3: 4 adjacent pixels
IMGCNT_NR_CNT0	NR1D_Y_TH[6:0]	8	Maximum Value (Absolute Value) of Y/G Signal Coring Coring is implemented when detected noise amount value ≤ NR1D_Y_TH. Unsigned: 0 to 127 [LSB]
IMGCNT_NR_CNT0	NR1D_Y_GAIN[1:0]	3	Noise Reduction Gain Adjustment of Y/G Signal 0: 1/2 1: 1/4 2: 1/8 3: 1/16
IMGCNT_NR_CNT1	NR1D_CB_TAP[1:0]	0	Cb/B Signal TAP Select 0: Adjacent pixel 1: 2 adjacent pixels 2: 3 adjacent pixels 3: 4 adjacent pixels
IMGCNT_NR_CNT1	NR1D_CB_TH[6:0]	8	Maximum Value (Absolute Value) of Cb/B Signal Coring Coring is implemented when detected noise amount value ≤ NR1D_C_TH. Unsigned: 0 to 127 [LSB]
IMGCNT_NR_CNT1	NR1D_CB_GAIN[1:0]	3	Noise Reduction Gain Adjustment of Cb/B Signal 0: 1/2 1: 1/4 2: 1/8 3: 1/16
IMGCNT_NR_CNT1	NR1D_CR_TAP[1:0]	0	Cr/R Signal TAP Select 0: Adjacent pixel 1: 2 adjacent pixels 2: 3 adjacent pixels 3: 4 adjacent pixels
IMGCNT_NR_CNT1	NR1D_CR_TH[6:0]	8	Maximum Value (Absolute Value) of Cr/R Signal Coring Coring is implemented when detected noise amount value ≤ NR1D_C_TH. Unsigned: 0 to 127 [LSB]
IMGCNT_NR_CNT1	NR1D_CR_GAIN[1:0]	3	Noise Reduction Gain Adjustment of Cr/R Signal 0: 1/2 1: 1/4 2: 1/8 3: 1/16

35.1.18 Color Matrix

By using a color matrix, input signal offsets and nine-axis gain can be adjusted. This enables brightness adjustment, gain adjustment, and YCbCr and GBR mutual conversion.

(1) GBR to GBR Conversion

YGIN_A = YGIN + IMGCNT_MTX_YG – 128
CBBIN_A = CBBIN + IMGCNT_MTX_B – 128
CRRIN_A = CRRIN + IMGCNT_MTX_R – 128
YGOUT = (IMGCNT_MTX_GG×YGIN_A + IMGCNT_MTX_GB×CBBIN_A + IMGCNT_MTX_GR×CRRIN_A) ÷ 256
CBBOUT = (IMGCNT_MTX_BG×YGIN_A + IMGCNT_MTX_BB×CBBIN_A + IMGCNT_MTX_BR×CRRIN_A) ÷ 256
CRROUT = (IMGCNT_MTX_RG×YGIN_A + IMGCNT_MTX_RB×CBBIN_A + IMGCNT_MTX_RR×CRRIN_A) ÷ 256

(2) GBR to YCbCr Conversion

YGIN_A = YGIN + IMGCNT_MTX_YG – 128
CBBIN_A = CBBIN + IMGCNT_MTX_B – 128
CRRIN_A = CRRIN + IMGCNT_MTX_R – 128
YGOUT = (IMGCNT_MTX_GG×YGIN_A + IMGCNT_MTX_GB×CBBIN_A +IMGCNT_MTX_GR×CRRIN_A) ÷ 256
CBBOUT = (IMGCNT_MTX_BG×YGIN_A + IMGCNT_MTX_BB×CBBIN_A + IMGCNT_MTX_BR×CRRIN_A) ÷ 256 + 128
CRROUT = (IMGCNT_MTX_RG×YGIN_A + IMGCNT_MTX_RB×CBBIN_A + IMGCNT_MTX_RR×CRRIN_A) ÷ 256 + 128

Table 35.24 Matrix Coefficient (Typical Value) for SMPTE 293M

	YGIN		CBBIN		CRRIN	
	Coefficient	Set Value	Coefficient	Set Value	Coefficient	Set Value
YGOUT	0.587	IMGCNT_ MTX_GG = 150	0.114	IMGCNT_MTX_GB = 29	0.299	IMGCNT_MTX_GR = 77
CBBOUT	–0.331	IMGCNT_ MTX_BG = 1963	0.500	IMGCNT_MTX_BB = 128	–0.169	IMGCNT_MTX_BR = 2005
CRROUT	–0.419	IMGCNT_ MTX_RG = 1941	–0.081	IMGCNT_MTX_RB = 2027	0.500	IMGCNT_MTX_RR = 128

(3) YCbCr to GBR Conversion

YGIN_A = YGIN + IMGCNT_MTX_YG – 128
CBBIN_A = CBBIN – 128
CRRIN_A = CRRIN – 128
YGOUT = (IMGCNT_MTX_GG×YGIN_A + IMGCNT_MTX_GB×CBBIN_A +IMGCNT_MTX_GR×CRRIN_A) ÷ 256
CBBOUT = (IMGCNT_MTX_BG×YGIN_A + IMGCNT_MTX_BB×CBBIN_A + IMGCNT_MTX_BR×CRRIN_A) ÷ 256
CRROUT = (IMGCNT_MTX_RG×YGIN_A + IMGCNT_MTX_RB×CBBIN_A + IMGCNT_MTX_RR×CRRIN_A) ÷ 256

Table 35.25 Matrix Coefficient (Typical Value) for SMPTE 293M

	YGIN		CBBIN		CRRIN	
	Coefficient	Set Value	Coefficient	Set Value	Coefficient	Set Value
YGOUT	1.000	IMGCNT_ MTX_GG = 256	–0.344	IMGCNT_MTX_GB = 1960	–0.714	IMGCNT_MTX_GR = 1865
CBBOUT	1.000	IMGCNT_ MTX_BG = 256	1.772	IMGCNT_MTX_BB = 454	0.000	IMGCNT_MTX_BR = 0
CRROUT	1.000	IMGCNT_ MTX_RG = 256	0.000	IMGCNT_MTX_RB = 0	1.402	IMGCNT_MTX_RR = 359

(4) YCbCr to YCbCr Conversion

YGIN_A = YGIN + IMGCNT_MTX_YG – 128
CBBIN_A = CBBIN – 128
CRRIN_A = CRRIN – 128
YGOUT = (IMGCNT_MTX_GG×YGIN_A + IMGCNT_MTX_GB×CBBIN_A + IMGCNT_MTX_GR×CRRIN_A) ÷ 256
CBBOUT = (IMGCNT_MTX_BG×YGIN_A + IMGCNT_MTX_BB×CBBIN_A + IMGCNT_MTX_BR×CRRIN_A) ÷ 256 + 128
CRROUT = (IMGCNT_MTX_RG×YGIN_A + IMGCNT_MTX_RB×CBBIN_A + IMGCNT_MTX_RR×CRRIN_A) ÷ 256 + 128

Table 35.26 YCbCr to GBR Conversion

Register Name	Bit Name	Initial Value	Description
IMGCNT_MTX_MODE	IMGCNT_MTX_MD [1:0]	3	Operating Mode 0: GBR → GBR 1: GBR → YCbCr 2: YCbCr → GBR 3: YCbCr → YCbCr
IMGCNT_MTX_YG_ ADJ0	IMGCNT_MTX_YG [7:0]	128	Offset (DC) Adjustment of Y/G Signal Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB], 512 [LSB])
IMGCNT_MTX_CBB_ ADJ0	IMGCNT_MTX_B [7:0]	128	Offset (DC) Adjustment of B Signal Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
IMGCNT_MTX_CRR_ ADJ0	IMGCNT_MTX_R [7:0]	128	Offset (DC) Adjustment of R Signal Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
IMGCNT_MTX_YG_ ADJ0	IMGCNT_MTX_GG [10:0]	256	Y/G Signal Gain Adjustment for Y/G Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_YG_ ADJ1	IMGCNT_MTX_GB [10:0]	0	Cb/B Signal Gain Adjustment for Y/G Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_YG_ ADJ1	IMGCNT_MTX_GR [10:0]	0	Cr/R Signal Gain Adjustment for Y/G Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_CBB_ ADJ0	IMGCNT_MTX_BG [10:0]	0	Y/G Signal Gain Adjustment for Cb/B Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_CBB_ ADJ1	IMGCNT_MTX_BB [10:0]	256	Cb/B Signal Gain Adjustment for Cb/B Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_CBB_ ADJ1	IMGCNT_MTX_BR [10:0]	0	Cr/R Signal Gain Adjustment for Cb/B Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_CRR_ ADJ0	IMGCNT_MTX_RG [10:0]	0	Y/G Signal Gain Adjustment for Cr/R Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_CRR_ ADJ1	IMGCNT_MTX_RB [10:0]	0	Cb/B Signal Gain Adjustment for Cr/R Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_CRR_ ADJ1	IMGCNT_MTX_RR [10:0]	256	Cr/R Signal Gain Adjustment for Cr/R Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

35.2 Register Descriptions

Table 35.27 and Table 35.28 show register Configuration.

- Symbols used in Register Description:

Initial value: Register value after a power-on reset
—: Undefined value
R/W: Readable/writable. The written value can be read.
R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.
R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.
R: Read-only. The write value should always be 0.
—/W: Write-only. The read value is undefined.

Table 35.27 Register Configuration of Input Controller

Name	Abbreviation	R/W	Initial Value	Address	Access Size
External input block register update control register	INP_UPDATE	R/WC1	H'0000 0000	H'FCFF 7400	32
Input select control register	INP_SEL_CNT	R/W	H'0000 0000	H'FCFF 7404	32
External input sync signal control register	INP_EXT_SYNC_CNT	R/W	H'0000 0000	H'FCFF 7408	32
Vsync signal phase adjustment register	INP_VSYNC_PH_ADJ	R/W	H'035A 01AD	H'FCFF 740C	32
Sync signal delay adjustment register	INP_DLY_ADJ	R/W	H'0000 0000	H'FCFF 7410	32

Table 35.28 Register Configuration of Image Quality Adjustment Block

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Image quality adjustment block register update control register	IMGCNT_UPDATE	R/WC1	H'0000 0000	H'FCFF 7480	32
NR control register 0	IMGCNT_NR_CNT0	R/W	H'0010 0803	H'FCFF 7484	32
NR control register 1	IMGCNT_NR_CNT1	R/W	H'0803 0803	H'FCFF 7488	32
Image quality adjustment block matrix mode register	IMGCNT_MTX_MODE	R/W	H'0000 0003	H'FCFF 74A0	32
Image quality adjustment block matrix YG adjustment register 0	IMGCNT_MTX_YG_ ADJ0	R/W	H'0080 0100	H'FCFF 74A4	32
Image quality adjustment block matrix YG adjustment register 1	IMGCNT_MTX_YG_ ADJ1	R/W	H'0000 0000	H'FCFF 74A8	32
Image quality adjustment block matrix CBB adjustment register 0	IMGCNT_MTX_CBB_ ADJ0	R/W	H'0080 0000	H'FCFF 74AC	32
Image quality adjustment block matrix CBB adjustment register 1	IMGCNT_MTX_CBB_ ADJ1	R/W	H'0100 0000	H'FCFF 74B0	32
Image quality adjustment block matrix CRR adjustment register 0	IMGCNT_MTX_CRR_ ADJ0	R/W	H'0080 0000	H'FCFF 74B4	32
Image quality adjustment block matrix CRR adjustment register 1	IMGCNT_MTX_CRR_ ADJ1	R/W	H'0000 0100	H'FCFF 74B8	32

35.2.1 External Input Block Register Update Control Register (INP_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	INP_EXT_UPDATE	—	—	—	INP_IMG_UPDATE
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INP_EXT_UPDATE	0	R/WC1	External Input Block Register Update 0: Registers are not updated. 1: Registers are updated.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INP_IMG_UPDATE	0	R/WC1	Sync Signal Adjustment Block Register Update 0: Registers are not updated. 1: Registers are updated.

35.2.2 Input Select Control Register (INP_SEL_CNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	INP_SEL	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	INP_FORMAT[2:0]			—	—	—	INP_PXD_EDGE	—	—	—	INP_VS_EDGE	—	—	—	INP_HS_EDGE
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	INP_SEL	0	R/W	Input Select 0: Input supplied via the external input pins is off. 1: Input supplied via the external input pins is on.
19 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14 to 12	INP_FORMAT [2:0]	0	R/W	External Input Format Select 0: YcbCr444, RGB888 1: RGB666 2: RGB565 3: BT656 4: BT601 5: YCbCr422 6, 7: Setting prohibited
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INP_PXD_EDGE	0	R/W	Clock Edge Select for Capturing External Input Video Image Signals DV_DATA23 to DV_DATA0 0: Rising edge 1: Falling edge
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INP_VS_EDGE	0	R/W	Clock Edge Select for Capturing External Input Vsync Signals DV_VSYNC 0: Rising edge 1: Falling edge
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INP_HS_EDGE	0	R/W	Clock Edge Select for Capturing External Input Hsync Signals DV_HSYNC 0: Rising edge 1: Falling edge

Note: INP_FORMAT, INP_PXD_EDGE, INP_VS_EDGE, and INP_HS_EDGE are updated when the INP_EXT_UPDATE bit in INP_UPDATE is 1. INP_SEL is updated when set.

35.2.3 External Input Sync Signal Control Register (INP_EXT_SYNC_CNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	INP_ENDIAN_ON	—	—	—	INP_SWAP_ON	—	—	—	INP_VS_INV	—	—	—	INP_HS_INV
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	INP_H_EDGE_SEL	—	—	—	INP_F525_625	—	—	INP_H_POS[1:0]	
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	INP_ENDIAN_ON	0	R/W	External Input Bit Endian Change On/Off Control 0: Off 1: On
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	INP_SWAP_ON	0	R/W	External Input B/R Signal Swap On/Off Control 0: Off 1: On
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	INP_VS_INV	0	R/W	External Input Vsync Signal DV_VSYNC Inversion Control 0: Not inverted (positive polarity) 1: Inverted (negative polarity)
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	INP_HS_INV	0	R/W	External Input Hsync Signal DV_HSYNC Inversion Control 0: Not inverted (positive polarity) 1: Inverted (negative polarity)
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INP_H_EDGE_SEL	0	R/W	Reference Select for External Input BT656 Hsync Signal 0: EAV 1: SAV
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INP_F525_625	0	R/W	Number of Lines for BT656 External Input 0: 525 lines 1: 625 lines
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	INP_H_POS[1:0]	0	R/W	Y/Cb/Y/Cr Data String Start Timing to Hsync Reference for BT656/601 or YCbCr422 External Input 0: Cb/Y/Cr/Y(BT656/601), Cb/Cr (YCbCr422) 1: Y/Cr/Y/Cb(BT656/601), setting prohibited (YCbCr422) 2: Cr/Y/Cb/Y(BT656/601), setting prohibited (YCbCr422) 3: Y/Cb/Y/Cr(BT656/601), Cr/Cb (YCbCr422)

Note: This register is updated when the INP_EXT_UPDATE bit in INP_UPDATE is 1.

35.2.4 Vsync Signal Phase Adjustment Register (INP_VSYNC_PH_ADJ)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	INP_FH50[11:0]											
Initial Value:	0	0	0	0	0	0	1	1	0	1	0	1	1	0	1	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	INP_FH25[11:0]											
Initial Value:	0	0	0	0	0	0	0	1	1	0	1	0	1	1	0	1
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	INP_FH50 [11:0]	858	R/W	Vsync Signal 1/2fH Phase Timing 1/2 clock cycle of the horizontal cycle should be set.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	INP_FH25 [11:0]	429	R/W	Vsync Signal 1/4fH Phase Timing 1/4 clock cycle of the horizontal cycle should be set.

Note: The INP_FH50[11:0] bits are updated when the INP_EXT_UPDATE and INP_IMG_UPDATE bits in INP_UPDATE are 1. The INP_FH25[11:0] bits are updated when the INP_IMG_UPDATE bit is 1.

35.2.5 Sync Signal Delay Adjustment Register (INP_DLY_ADJ)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	INP_VS_DLY_L[2:0]			INP_FLD_DLY[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INP_VS_DLY[7:0]								INP_HS_DLY[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 24	INP_VS_ DLY_L[2:0]	0	R/W	Number of lines for Delaying Vsync signal and Field Differentiation Signal Delay amount: 0 to 7 (lines)
23 to 16	INP_FLD_ DLY[7:0]	0	R/W	Field Differentiation Signal Delay Amount Delay amount: 0 to 254 (clock cycles)
15 to 8	INP_VS_ DLY[7:0]	0	R/W	Vsync Signal Delay Amount Delay amount: 0 to 254 (clock cycles)
7 to 0	INP_HS_ DLY[7:0]	0	R/W	Hsync Signal Delay Amount Delay amount: 0 to 254 (clock cycles)

Note: This register is updated when the INP_IMG_UPDATE bit in INP_UPDATE is 1.

35.2.6 Image Quality Adjustment Block Register Update Control Register (IMGCNT_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IMGCNT_VEN
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	IMGCNT_VEN	0	R/WC1	Image Quality Adjustment Block Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync signal.

35.2.7 NR Control Register 0 (IMGCNT_NR_CNT0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	NR1D_MD	—	—	—	NR1D_ON
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	NR1D_Y_TH[6:0]							—	—	NR1D_Y_TAP[1:0]		—	—	NR1D_Y_GAIN[1:0]	
Initial Value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	NR1D_MD	1	R/W	Horizontal Noise Reduction Operating Mode 0: G/B/R mode 1: Y/Cb/Cr mode
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	NR1D_ON	0	R/W	Noise Reduction On/Off Control 0: Noise reduction Off 1: Noise reduction On
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 8	NR1D_Y_TH[6:0]	8	R/W	Maximum Value (Absolute Value) of Y/G Signal Coring Coring is implemented when detected noise amount value ≤ NR1D_Y_TH. Unsigned: 0 to 127 [LSB]
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	NR1D_Y_TAP[1:0]	0	R/W	Y/G Signal TAP Select 0: Adjacent pixel 1: 2 adjacent pixels 2: 3 adjacent pixels 3: 4 adjacent pixels
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	NR1D_Y_GAIN[1:0]	3	R/W	Noise Reduction Gain Adjustment of Y/G Signal 0: 1/2 1: 1/4 2: 1/8 3: 1/16

Note: This register is updated when the IMGCNT_VEN bit in IMGCNT_UPDATE is 1.

35.2.8 NR Control Register 1 (IMGCNT_NR_CNT1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	NR1D_CB_TH[6:0]							—	—	NR1D_CB_TAP[1:0]		—	—	NR1D_CB_GAIN[1:0]	
Initial Value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	NR1D_CR_TH[6:0]							—	—	NR1D_CR_TAP[1:0]		—	—	NR1D_CR_GAIN[1:0]	
Initial Value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 24	NR1D_CB_TH[6:0]	8	R/W	Maximum Value (Absolute Value) of Cb/B Signal Coring Coring is implemented when detected noise amount value ≤ NR1D_CB_TH. Unsigned: 0 to 127 [LSB]
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	NR1D_CB_TAP[1:0]	0	R/W	Cb/B Signal TAP Select 0: Adjacent pixel 1: 2 adjacent pixels 2: 3 adjacent pixels 3: 4 adjacent pixels
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	NR1D_CB_GAIN[1:0]	3	R/W	Noise Reduction Gain Adjustment of Cb/B Signal 0: 1/2 1: 1/4 2: 1/8 3: 1/16
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 8	NR1D_CR_TH[6:0]	8	R/W	Maximum Value (Absolute Value) of Cr/R Signal Coring Coring is implemented when detected noise amount value ≤ NR1D_CR_TH. Unsigned: 0 to 127 [LSB]
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	NR1D_CR_TAP[1:0]	0	R/W	Cr/R Signal TAP Select 0: Adjacent pixel 1: 2 adjacent pixels 2: 3 adjacent pixels 3: 4 adjacent pixels
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	NR1D_CR_GAIN[1:0]	3	R/W	Noise Reduction Gain Adjustment of Cr/R Signal 0: 1/2 1: 1/4 2: 1/8 3: 1/16

Note: This register is updated when the IMGCNT_VEN bit in IMGCNT_UPDATE is 1.

35.2.9 Image Quality Adjustment Block Matrix Mode Register (IMGCNT_MTX_MODE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IMGCNT_MTX_MD[1:0]	
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	IMGCNT_MTX_MD[1:0]	3	R/W	Operating Mode 0: GBR → GBR 1: GBR → YCbCr 2: YCbCr → GBR 3: YCbCr → YCbCr

Note: This register is updated when the IMGCNT_VEN bit in IMGCNT_UPDATE is 1.

35.2.10 Image Quality Adjustment Block Matrix YG Adjustment Register 0 (IMGCNT_MTX_YG_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	IMGCNT_MTX_YG[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_GG[10:0]										
Initial Value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	IMGCNT_MTX_YG[7:0]	128	R/W	Offset (DC) Adjustment of Y/G Signal Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	IMGCNT_MTX_GG[10:0]	256	R/W	Y/G Signal Gain Adjustment for Y/G Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when the IMGCNT_VEN bit in IMGCNT_UPDATE is 1.

35.2.11 Image Quality Adjustment Block Matrix YG Adjustment Register 1 (IMGCNT_MTX_YG_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	IMGCNT_MTX_GB[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_GR[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	IMGCNT_MTX_GB[10:0]	0	R/W	Cb/B Signal Gain Adjustment for Y/G Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	IMGCNT_MTX_GR[10:0]	0	R/W	Cr/R Signal Gain Adjustment for Y/G Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when the IMGCNT_VEN bit in IMGCNT_UPDATE is 1.

35.2.12 Image Quality Adjustment Block Matrix CBB Adjustment Register 0 (IMGCNT_MTX_CBB_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	IMGCNT_MTX_B[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_BG[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	IMGCNT_MTX_B[7:0]	128	R/W	Offset (DC) Adjustment of Cb/B Signal Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	IMGCNT_MTX_BG[10:0]	0	R/W	Y/G Signal Gain Adjustment for Cb/B Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when the IMGCNT_VEN bit in IMGCNT_UPDATE is 1.

35.2.13 Image Quality Adjustment Block Matrix CBB Adjustment Register 1 (IMGCNT_MTX_CBB_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	IMGCNT_MTX_BB[10:0]										
Initial Value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_BR[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	IMGCNT_MTX_BB[10:0]	256	R/W	Cb/B Signal Gain Adjustment for Cb/B Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	IMGCNT_MTX_BR[10:0]	0	R/W	Cr/R Signal Gain Adjustment for Cb/B Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when the IMGCNT_VEN bit in IMGCNT_UPDATE is 1.

35.2.14 Image Quality Adjustment Block Matrix CRR Adjustment Register 0 (IMGCNT_MTX_CRR_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	IMGCNT_MTX_R[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_RG[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	IMGCNT_MTX_R[7:0]	128	R/W	Offset (DC) Adjustment of Cr/R Signal Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	IMGCNT_MTX_RG[10:0]	0	R/W	Y/G Signal Gain Adjustment for Cr/R Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when the IMGCNT_VEN bit in IMGCNT_UPDATE is 1.

35.2.15 Image Quality Adjustment Block Matrix CRR Adjustment Register 1 (IMGCNT_MTX_CRR_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	IMGCNT_MTX_RB[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_RR[10:0]										
Initial Value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	IMGCNT_ MTX_RB[10:0]	0	R/W	Cb/B Signal Gain Adjustment for Cr/R Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	IMGCNT_ MTX_RR[10:0]	256	R/W	Cr/R Signal Gain Adjustment for Cr/R Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when the IMGCNT_VEN bit in IMGCNT_UPDATE is 1.

35.3 Usage Methods

35.3.1 Input Format Adjustment Method

Setting examples of each input format are shown below.

Table 35.29 External Input (BT656, 525i) Setting Example

Register Name	Bit Name	Description	Setting Value
INP_SEL_CNT	INP_SEL	Performs the on/off control of input supplied via the external input pins.	1
INP_SEL_CNT	INP_FORMAT[2:0]	Selects the externally input format.	3
INP_SEL_CNT	INP_PXD_EDGE	Selects the clock edge for capturing the externally input video signals.	0
INP_SEL_CNT	INP_VS_EDGE	Selects the clock edge for capturing the externally input Vsync signals.	0
INP_SEL_CNT	INP_HS_EDGE	Selects the clock edge for capturing the externally input Hsync signals.	0
INP_EXT_SYNC_CNT	INP_ENDIAN_ON	Changes the bit endian of the external input.	0
INP_EXT_SYNC_CNT	INP_SWAP_ON	Enables or disables the B/R signal swap of the external input.	0
INP_EXT_SYNC_CNT	INP_HS_INV	Enables or disables the Hsync signal inversion of the external input.	1
INP_EXT_SYNC_CNT	INP_VS_INV	Enables or disables the Hsync signal inversion of the external input.	1
INP_EXT_SYNC_CNT	INP_H_EDGE_SEL	Selects the Hsync reference for BT656 input.	0
INP_EXT_SYNC_CNT	INP_F525_625	Sets the number of lines for BT656 input.	0
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	Sets the data start timing with respect to the Hsync in the BT656/601 format.	0
INP_VSYNC_PH_ADJ	INP_FH50[11:0]	Sets the 1/2fH phase in clock cycle units.	858
INP_VSYNC_PH_ADJ	INP_FH25[11:0]	Sets the 1/4fH phase in clock cycle units.	429
INP_DLY_ADJ	INP_VS_DLY_L[2:0]	Sets the number of lines for delaying the Vsync signal and field differentiation signal.	0
INP_DLY_ADJ	INP_VS_DLY[7:0]	Sets the amount of delay of the Vsync signal in clock cycle units.	0
INP_DLY_ADJ	INP_HS_DLY[7:0]	Sets the amount of delay of the Hsync signal in clock cycle units.	0
INP_DLY_ADJ	INP_FLD_DLY[7:0]	Sets the amount of delay of the field differentiation signal in clock units.	0

Note: Some registers require, after they are set, that the INP_EXT_UPDATE and INP_IMG_UPDATE bits in INP_UPDATE should be set to 1.

Table 35.30 External Input (BT601, 525i) Setting Example

Register Name	Bit Name	Description	Setting Value
INP_SEL_CNT	INP_SEL	Performs the on/off control of input supplied via the external input pins.	1
INP_SEL_CNT	INP_FORMAT[2:0]	Selects the externally input format.	4
INP_SEL_CNT	INP_PXD_EDGE	Selects the clock edge for capturing the externally input video signals.	0
INP_SEL_CNT	INP_VS_EDGE	Selects the clock edge for capturing the externally input Vsync signals.	0
INP_SEL_CNT	INP_HS_EDGE	Selects the clock edge for capturing the externally input Hsync signals.	0
INP_EXT_SYNC_CNT	INP_ENDIAN_ON	Changes the bit endian of the external input.	0
INP_EXT_SYNC_CNT	INP_SWAP_ON	Enables or disables the B/R signal swap of the external input.	0
INP_EXT_SYNC_CNT	INP_HS_INV	Enables or disables the Hsync signal inversion of the external input.	1
INP_EXT_SYNC_CNT	INP_VS_INV	Enables or disables the Hsync signal inversion of the external input.	1
INP_EXT_SYNC_CNT	INP_H_EDGE_SEL	Selects the Hsync reference for BT656 input.	0
INP_EXT_SYNC_CNT	INP_F525_625	Sets the number of lines for BT656 input.	0
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	Sets the data start timing with respect to the Hsync in the BT656/601 format.	0
INP_VSYNC_PH_ADJ	INP_FH50[11:0]	Sets the 1/2fH phase in clock cycle units.	858
INP_VSYNC_PH_ADJ	INP_FH25[11:0]	Sets the 1/4fH phase in clock cycle units.	429
INP_DLY_ADJ	INP_VS_DLY_L[2:0]	Sets the number of lines for delaying the Vsync signal and field differentiation signal.	0
INP_DLY_ADJ	INP_VS_DLY[7:0]	Sets the amount of delay of the Vsync signal in clock cycle units.	0
INP_DLY_ADJ	INP_HS_DLY[7:0]	Sets the amount of delay of the Hsync signal in clock cycle units.	0
INP_DLY_ADJ	INP_FLD_DLY[7:0]	Sets the amount of delay of the field differentiation signal in clock units.	0

Note: Some registers require, after they are set, that the INP_EXT_UPDATE and INP_IMG_UPDATE bits in INP_UPDATE should be set to 1.

35.3.2 Usage Method of Conversion Color Matrix

Typical data conversion setting examples are shown below.

Table 35.31 Conversion Color Matrix

Register Name	Bit Name	GBR to GBR	GBR to YCbCr	YCbCr to GBR	YCbCr to YCbCr
IMGCNT_MTX_MODE	IMGCNT_MTX_MD[1:0]	0	1	2	3
IMGCNT_MTX_YG_ADJ0	IMGCNT_MTX_YG[7:0]	128	128	128	128
IMGCNT_MTX_YG_ADJ0	IMGCNT_MTX_GG[10:0]	256	150	256	256
IMGCNT_MTX_YG_ADJ1	IMGCNT_MTX_GB[10:0]	0	29	1960	0
IMGCNT_MTX_YG_ADJ1	IMGCNT_MTX_GR[10:0]	0	77	1865	0
IMGCNT_MTX_CBB_ADJ0	IMGCNT_MTX_B[7:0]	128	128	128	128
IMGCNT_MTX_CBB_ADJ0	IMGCNT_MTX_BG[10:0]	0	1963	256	0
IMGCNT_MTX_CBB_ADJ1	IMGCNT_MTX_BB[10:0]	256	128	454	256
IMGCNT_MTX_CBB_ADJ1	IMGCNT_MTX_BR[10:0]	0	2005	0	0
IMGCNT_MTX_CRR_ADJ0	IMGCNT_MTX_R[7:0]	128	128	128	128
IMGCNT_MTX_CRR_ADJ0	IMGCNT_MTX_RG[10:0]	0	1941	256	0
IMGCNT_MTX_CRR_ADJ1	IMGCNT_MTX_RB[10:0]	0	2027	0	0
IMGCNT_MTX_CRR_ADJ1	IMGCNT_MTX_RR[10:0]	256	128	359	256

Note: The registers require, after they are set, that the IMGCNT_VEN bit in IMGCNT_UPDATE should be set to 1.

36. Video Display Controller 6 (3): Scaler

36.1 Scaler

36.1.1 Overview of Functions

The scaler subjects the YCbCr and RGB signals output from the input controller, to sync signal generation; and reduction, enlargement, and rotation of the images.

The scaler also records video image in the frame buffer.

In scaler 0, either enlargement process or graphics 0 process can be used.

The functional block diagram of scaler 0 is shown below.

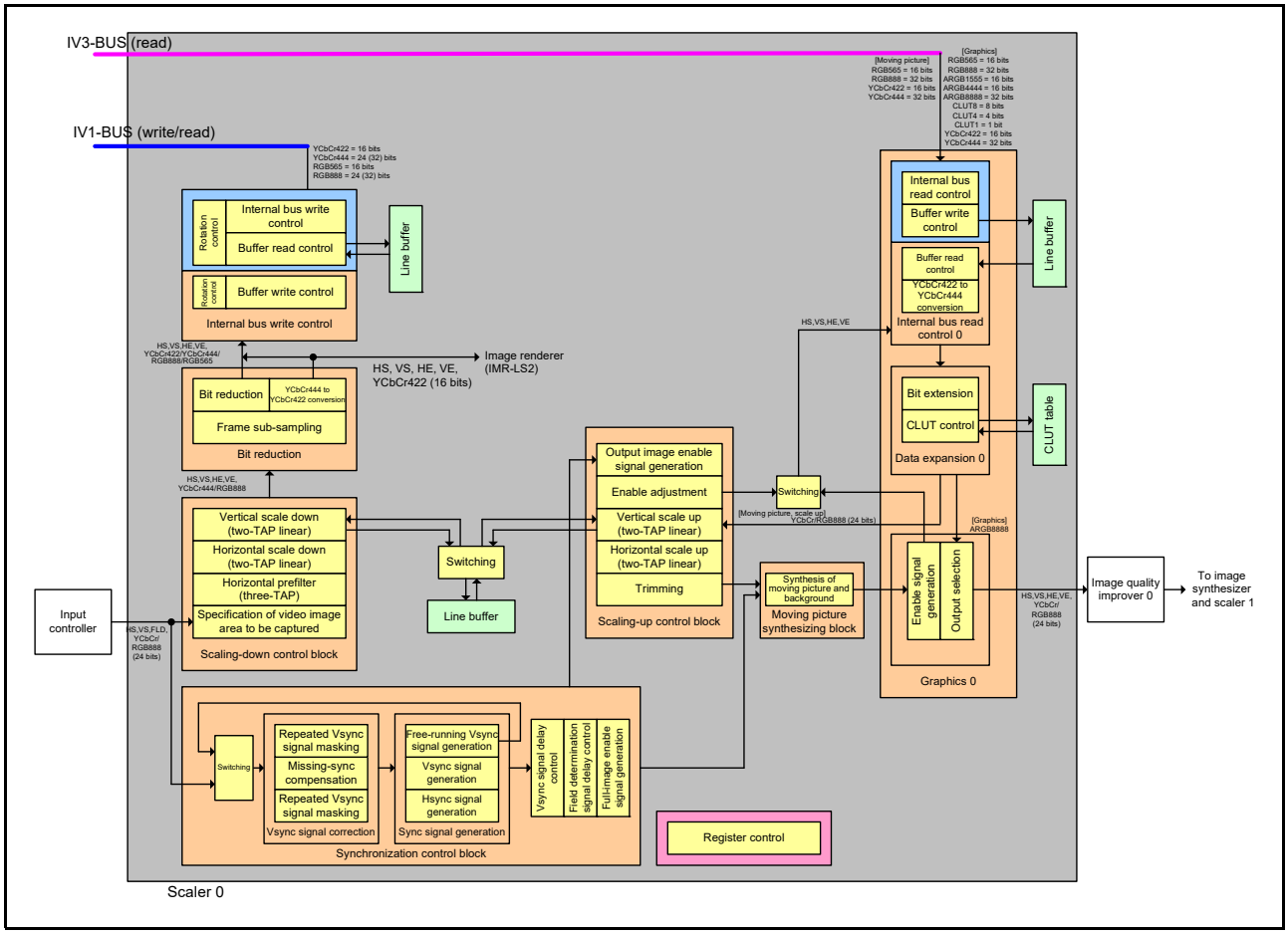


Figure 36.1 Functional Block Diagram of Scaler 0

The registers and bits in the scaler are named SC0_xxxx and those in the graphics blocks are named GR0_xxxx, but in this section, they are collectively called SC_xxxx or GR_xxxx.

36.1.2 Register Control

(1) Updating Registers

The Vsync signal is used to control the update timing of all the registers of the scaling and graphics blocks except some registers of the sync control block and some of the other blocks.

After 1 is set to the bits in the update control register, the contents of the relevant registers are modified at the rising edge of the Vsync signal. The update control register is automatically cleared to 0 after the modification.

Table 36.1 Register Update Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_UPDATE	SC_SCL0_UPDATE	0	SYNC Control Register Update 0: Registers are not updated. 1: Registers are updated.
SC_SCL0_UPDATE	SC_SCL0_VEN_D	0	Scaling-Up Control and Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
SC_SCL0_UPDATE	SC_SCL0_VEN_C	0	Scaling-Down Control and Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
SC_SCL0_UPDATE	SC_SCL0_VEN_B	0	Synchronization Control and Scaling-up Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
SC_SCL0_UPDATE	SC_SCL0_VEN_A	0	Scaling-Down Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
SC_SCL1_UPDATE	SC_SCL1_UPDATE_B	0	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated.
SC_SCL1_UPDATE	SC_SCL1_UPDATE_A	0	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated.
SC_SCL1_UPDATE	SC_SCL1_VEN_B	0	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
SC_SCL1_UPDATE	SC_SCL1_VEN_A	0	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GR_UPDATE	GR_UPDATE	0	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated.
GR_UPDATE	GR_P_VEN	0	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GR_UPDATE	GR_IBUS_VEN	0	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

The registers controlled by SC_SCL0_VEN_A, SC_SCL0_VEN_C, SC_SCL1_VEN_A, and SC_SCL1_VEN_B are modified at the rising edge of the input Vsync signal.

The registers controlled by SC_SCL0_VEN_B, SC_SCL0_VEN_D, GR_P_VEN, and GR_IBUS_VEN are modified at the rising edge of the output Vsync signal.

36.1.3 Synchronization Control

(1) Selecting Vsync Signal

The Vsync signal to be output from the scaler can be selected.

When an external input signal is to be displayed, an external input Vsync signal should be selected to be output.

When an external input signal is not provided, a free-running Vsync signal should be selected to be output.

Table 36.2 Vsync Signal Selection Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC3	SC_RES_VS_SEL	1	Vsync Signal Output Select 0: External input Vsync signal 1: Internally generated free-running Vsync signal

(2) Masking Repeated Vsync Signals

It is possible to prevent receiving the Vsync signal with a period shorter than the standard period. This is achieved by setting the start timing to receive the next Vsync signal after receiving an input Vsync signal.

The Vsync signal reception masking period is set with the SC_RES_VMASK[15:0] bits.

Masking period [usec] = SC_RES_VMASK × 128 ÷ pixel clock frequency [MHz]

This function is enabled or disabled by the SC_RES_VMASK_ON bit.

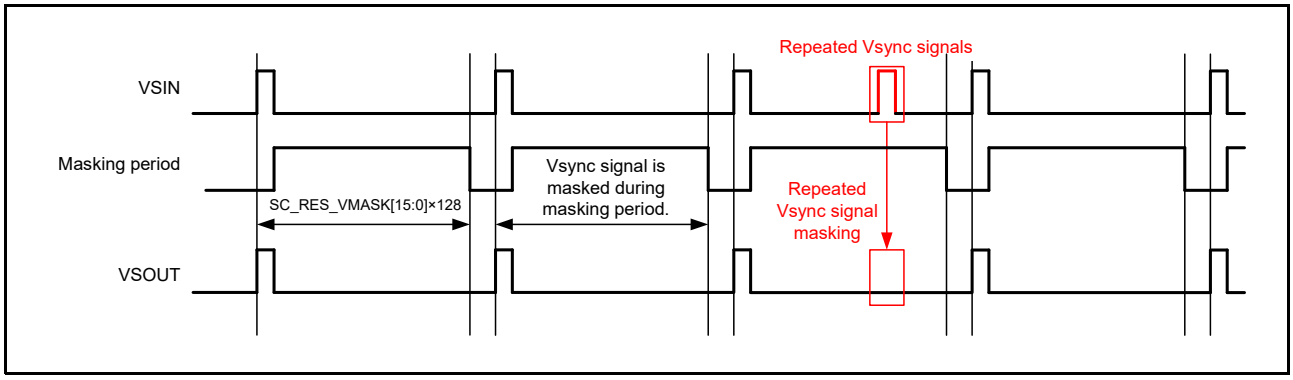


Figure 36.2 Timing for Masking Repeated Vsync Signals

Table 36.3 Repeated Vsync Signal Mask Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC1	SC_RES_VMASK_ON	1	Repeated Vsync Signal Masking Control 0: Repeated Vsync signal masking control is disabled. 1: Repeated Vsync signal masking control is enabled.
SC_SCL0_FRC1	SC_RES_VMASK[15:0]	2800	Repeated Vsync Signal Masking Period Sets the repeated Vsync signal masking period beginning at a Vsync signal in terms of 128 pixel-clock periods. Masking period [usec] = SC_RES_VMASK × 128 ÷ pixel clock frequency [MHz]

(3) Compensating for Missing Vsync Signals

It is possible to prevent output of the Vsync signal with a period longer than the standard period. This is achieved by setting the wait time after reception of an input Vsync signal until reception of the next Vsync signal.

If no Vsync signals are received during the wait time, an internally generated sync signal is inserted.

The wait time can be set using the SC_RES_VLACK[15:0] bits.

Wait time [usec] = SC_RES_VLACK × 128 ÷ pixel clock frequency [MHz]

This function is enabled or disabled by the SC_RES_VLACK_ON bit.

If no Vsync signals are input during the Vsync signal reception time, the SC_RES_QVLACK bit is set to the high level.

If Vsync signals are continuously detected four or more times during the Vsync signal reception time, the SC_RES_QVLOCK bit is set to the high level.

The SC_RES_QVLOCK bit is valid even when both the SC_RES_VMASK_ON and SC_RES_VLACK_ON bits are set to turn off the corresponding functions.

Note that, however, the SC_RES_VMASK and SC_RES_VLACK bits must be set correctly.

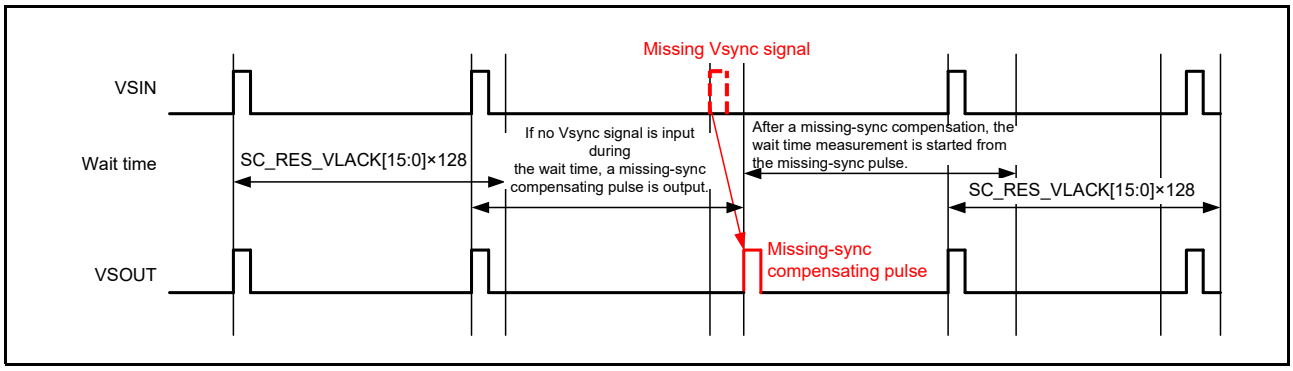


Figure 36.3 Compensation of Missing Vsync Signals

Table 36.4 Missing Vsync Compensation Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC2	SC_RES_VLACK_ON	1	Missing Vsync Signal Compensation 0: Compensation of missing Vsync signals is disabled. 1: Compensation of missing Vsync signals is enabled.
SC_SCL0_FRC2	SC_RES_VLACK[15:0]	3600	Missing-Sync Compensating Pulse Output Wait Time Sets the wait time before outputting a missing-sync compensating pulse after a Vsync signal. Wait time [usec] = SC_RES_VLACK × 128 ÷ pixel clock frequency [MHz]
SC_SCL0_FRC9	SC_RES_QVLACK	—	Missing Vsync Signal Detection Flag 1: Missing Vsync signal input has been detected. 0: No missing Vsync signal input has been detected.
SC_SCL0_FRC9	SC_RES_QVLOCK	—	Locked Vsync Signal Detection Flag 1: No repeated or missing Vsync signal input has been detected for four or more vertical periods. 0: Repeated or missing Vsync signal input has been detected.

For the Vsync signal, repeated-signal masking is first carried out and then missing-signal compensation is carried out, followed by another repeated-signal masking.

Repetition masking is inserted after missing-Vsync compensation to prevent output of the Vsync signal even in cases such as the input of a Vsync signal immediately after the input of a pulse to compensate for a missing Vsync signal.

On/off control of the missing-Vsync compensation also applies to the second repeated-Vsync masking; and masking period setting of the first repeated-Vsync masking also applies to the second repeated-Vsync masking.

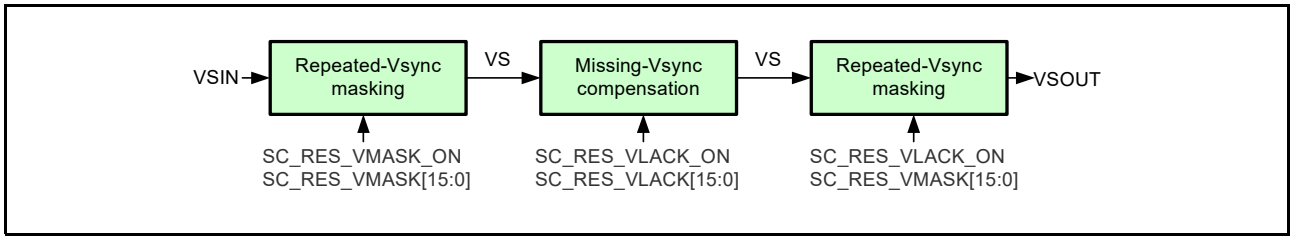


Figure 36.4 Repeated-Vsync Masking and Missing-Vsync Compensation

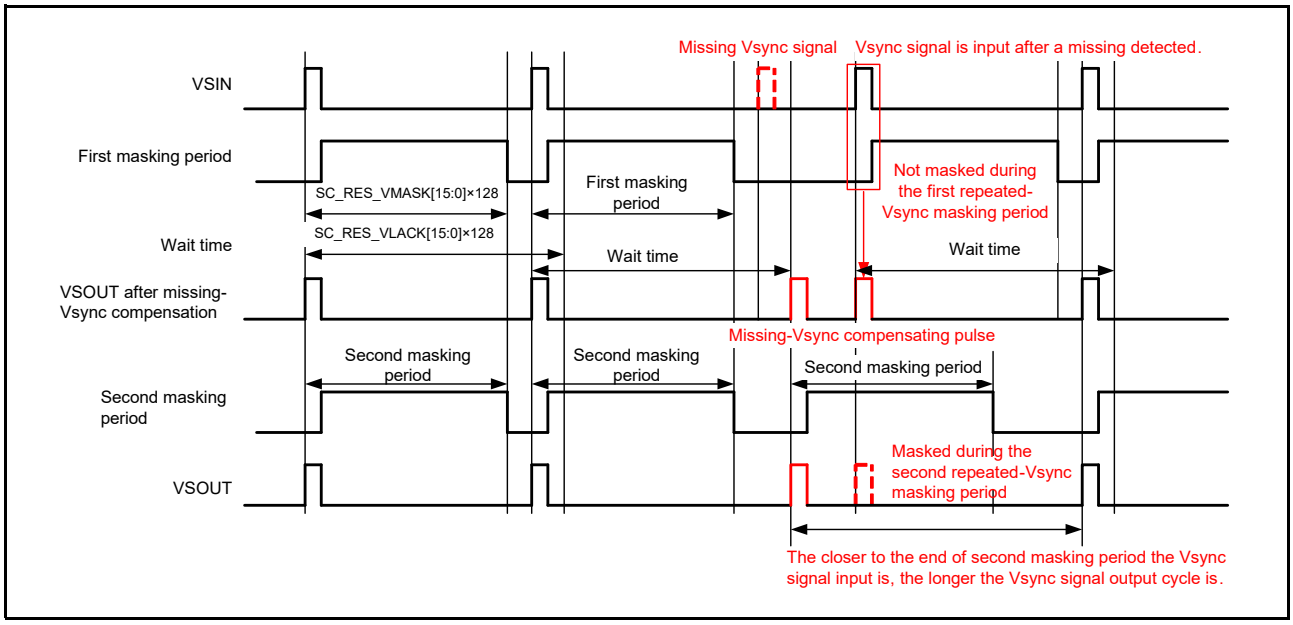


Figure 36.5 Timing for Masking Repeated Vsync Signals and Missing Vsync Signal Compensation

(4) Free-Running Period

Free-running Vsync and Hsync periods can be set.

Hsync period [usec] = (SC_RES_FH + 1) ÷ pixel clock frequency [MHz]
Vsync period [usec] = horizontal period [usec] × (SC_RES_FV + 1)

Table 36.5 Free-Running Period Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC4	SC_RES_FV[10:0]	524	Free-Running Vsync Period Setting Free-running Vsync period = (SC_RES_FV + 1) × horizontal period [usec]
SC_SCL0_FRC4	SC_RES_FH[10:0]	799	Hsync Period Setting Hsync period [usec] = (SC_RES_FH +1) ÷ pixel clock frequency [MHz]

When selecting an external input Vsync signal, set the SC_RES_VS_SEL bit to 0. At this time, the internally generated free-running Vsync signal is not output.

In the meantime, the Hsync signal is always generated according to the free-running signal setting and output from the scaler.

(5) Vsync Signal Delay Control

Delay of Vsync signal output from the scaler can be controlled.

The delay is used to adjust the frame buffer read timing.

Table 36.6 Vsync Output Delay Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC5	SC_RES_VSDLY[7:0]	1	Vsync Signal Delay Control Adjusts the Vsync signal delay in the output Hsync period units. Vsync signal delay [usec]: SC_RES_VSDLY × output Hsync period [usec]

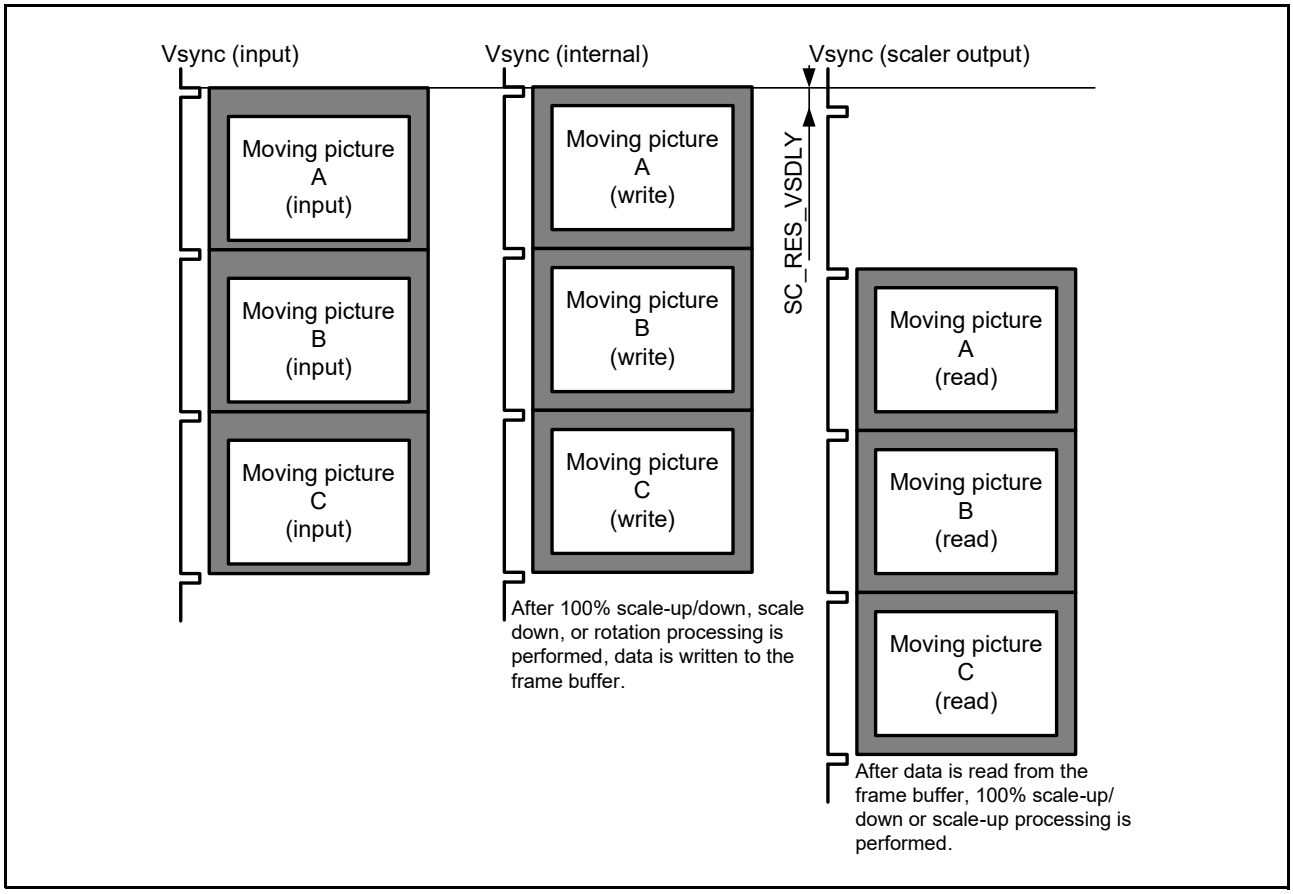


Figure 36.6 Vsync Signal Phases (Two Frame-Buffer Planes Used)

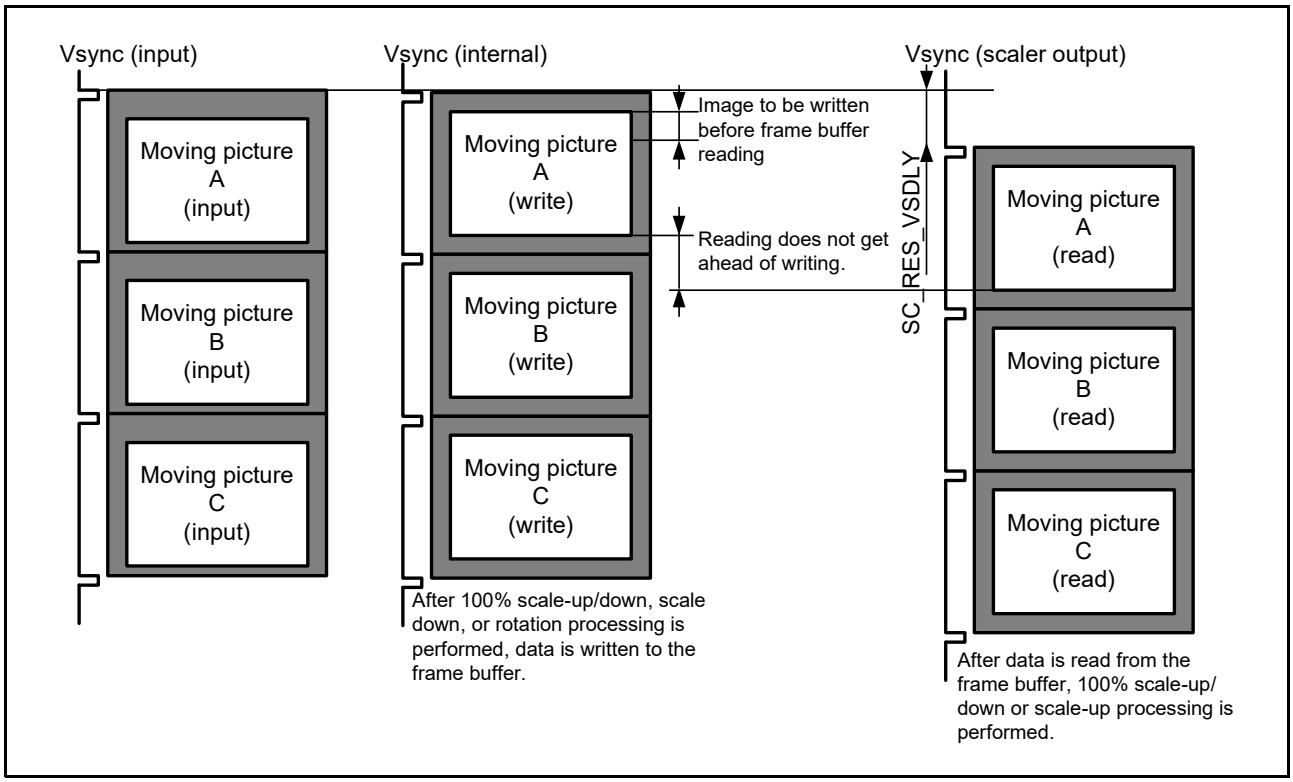


Figure 36.7 Vsync Signal Phases (One Frame-Buffer Plane Used)

36.1.4 Setting Angle of View

(1) Setting Image Area to be Captured

The image area to be captured can be set for reduction or enlargement.

The area is defined by specifying its start position and width based on the input Hsync and Vsync signals.

Table 36.7 Control of Image Area to be Captured

Register Name	Bit Name	Initial Value	Description
SC_SCL0_DS2	SC_RES_VS[10:0]	18	Vertical Position Setting for Video Signal Capturing (VSYNC + (V backporch - 1) lines) Note: The set value should be four or more (lines). SC_RES_VS + SC_RES_VW should be equal to or less than 2039 (lines).
SC_SCL0_DS2	SC_RES_VW[10:0]	240	Vertical Width of Video Signal to be Captured (lines) Note: SC_RES_VS + SC_RES_VW should be equal to or less than 2039 (lines).
SC_SCL0_DS3	SC_RES_HS[10:0]	244	Horizontal Position Setting for Video Signal Capturing (HSYNC + H backporch video-image clock cycles) Note: The set value should be 16 or more (clock cycles). SC_RES_HS + SC_RES_HW should be equal to or less than 4063 (clock cycles).
SC_SCL0_DS3	SC_RES_HW[11:0]	1440	Horizontal Width of Video Signal to be Captured (video-image clock cycles) Note: SC_RES_HS + SC_RES_HW should be equal to or less than 4063 (clock cycles).

(2) Generating a Full-Screen Enable Signal

The valid period of the full screen to be output from the scaler can be set.

The valid period is defined by specifying its start position and width based on the Hsync and Vsync signals output from the scaler.

The vertical front porch should be set to four or more lines, and the horizontal front porch should be 16 or more clock cycles.

Table 36.8 Full-Screen Enable Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC6	SC_RES_F_VS[10:0]	35	Vertical Enable Signal Start Position for Full Screen. (VSYNC + V backporch lines) Note: The set value should be four or more (lines). SC_RES_F_VS + SC_RES_F_VW should be equal to or less than 2039 (lines).
SC_SCL0_FRC6	SC_RES_F_VW[10:0]	480	Vertical Enable Signal Width for Full Screen (lines) Note: SC_RES_F_VS + SC_RES_F_VW should be equal to or less than 2039 (lines).
SC_SCL0_FRC7	SC_RES_F_HS[10:0]	144	Horizontal Enable Signal Start Position for Full Screen. (HSYNC + H backporch pixel-clock cycles) Note: The set value should be 16 or more (clock cycles). SC_RES_F_HS + SC_RES_F_HW should be equal to or less than 2015 (clock cycles).
SC_SCL0_FRC7	SC_RES_F_HW[10:0]	640	Horizontal Enable Signal Width for Full Screen (pixel-clock cycles) Note 1. SC_RES_F_HS + SC_RES_F_HW should be equal to or less than 2015 (clock cycles). Note 2. The set value should be equal to (horizontal signal width for full screen + 2) when serial RGB output is selected as an LCD output signal.

(3) Generating an Image Output Enable Signal

The valid period of the image to be output can be set.

The valid period is defined by specifying its start position and width based on the Hsync and Vsync signals output from the scaler.

Table 36.9 Image Output Enable Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_US2	SC_RES_P_VS[10:0]	35	Vertical Enable Signal Start Position for Output Image. (VSYNC + V backporch lines) Note: The set value should be four or more (lines). SC_RES_P_VS + SC_RES_P_VW should be equal to or less than 2039 (lines).
SC_SCL0_US2	SC_RES_P_VW[10:0]	480	Vertical Enable Signal Width for Output Image (lines) Note: SC_RES_P_VS + SC_RES_P_VW should be equal to or less than 2039 (lines).
SC_SCL0_US3	SC_RES_P_HS[10:0]	144	Horizontal Enable Signal Start Position for Output Image. (HSYNC + H backporch pixel-clock cycles) Note: The set value should be 16 or more (clock cycles). SC_RES_P_HS + SC_RES_P_HW should be equal to or less than 2015 (clock cycles).
SC_SCL0_US3	SC_RES_P_HW[10:0]	640	Horizontal Enable Signal Width for Output Image (pixel-clock cycles) Note: SC_RES_P_HS + SC_RES_P_HW should be equal to or less than 2015 (clock cycles).

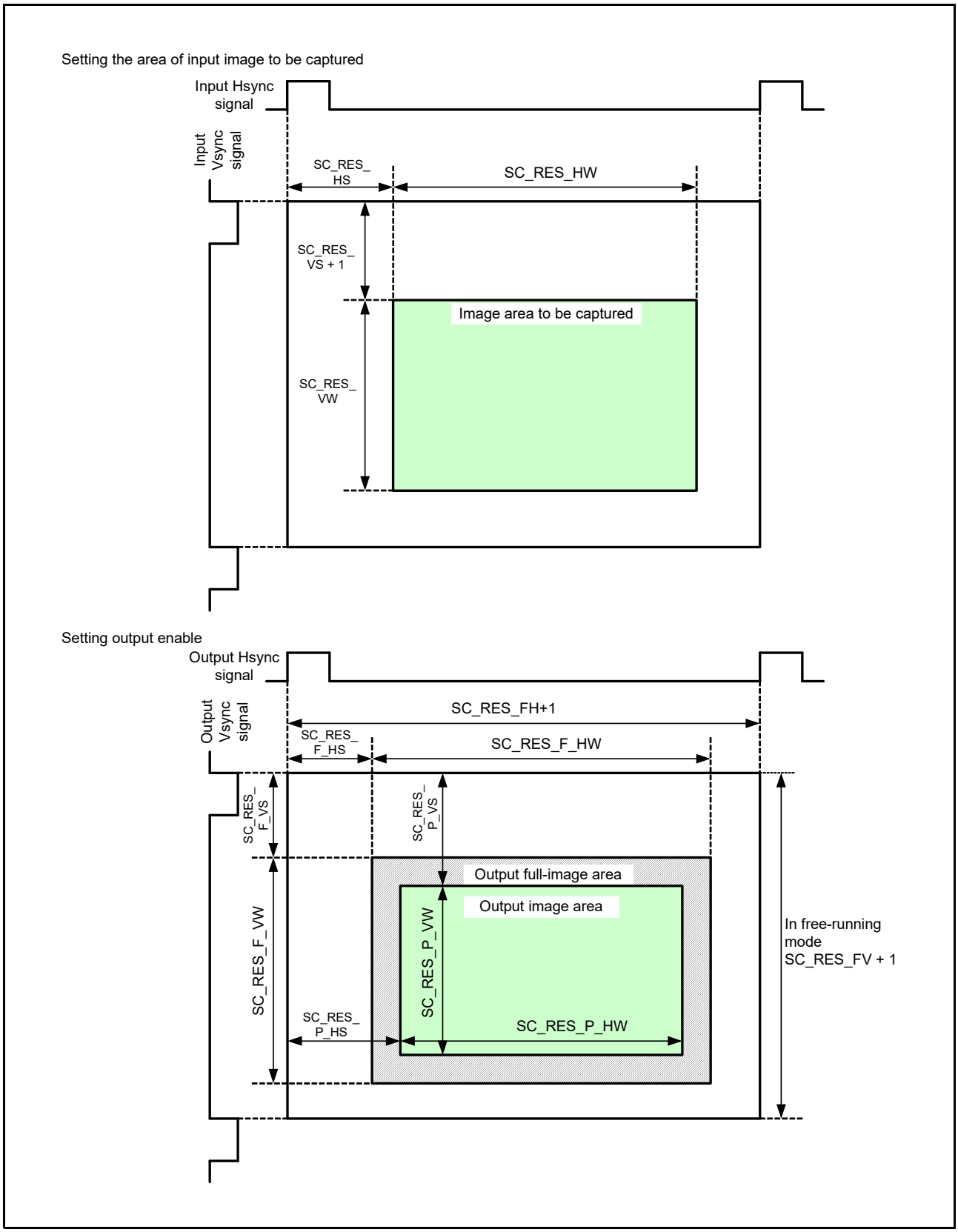


Figure 36.8 Enable Settings

36.1.5 Scaling Settings

(1) Scaling Processing Block

The scaling-down control block scales down the input image from the input controller.

When rotation is required, the scaling-down control block first scales down the image and then rotates it before writing it into the frame buffer.

The scaling-up control block reads the rotated image from the frame buffer and scales it up.
When the image input from the input controller has a vertical valid period greater than 1024 lines or a horizontal valid period greater than 1440 clock cycles, be sure to scale down the image so that the vertical valid period is no greater than 1024 lines and the horizontal valid period is no greater than 1440 clock cycles by using the scaling-down control block.

Table 36.10 Rotation and Scaling Process

Rotation	Horizontal Scaling	Vertical Scaling	Scaling-Down Control Block	Scaling-Up Control Block
Normal	Horizontal scale down	Vertical scale down	Horizontal scale down/ vertical scale down	Horizontal 100% scale up/ vertical 100% scale up
	Horizontal scale down	Vertical scale up	Horizontal scale down/ vertical 100% scale up	Horizontal 100% scale up/ vertical scale up
	Horizontal scale up	Vertical scale down	Horizontal 100% scale up/ vertical scale down	Horizontal scale up/ vertical 100% scale up
	Horizontal scale up	Vertical scale up	Horizontal 100% scale up/ vertical 100% scale up	Horizontal scale up/ vertical scale up
Horizontal mirroring	Horizontal scale down	Vertical scale down	Horizontal scale down/ vertical scale down	Horizontal 100% scale up/ vertical 100% scale up
	Horizontal scale down	Vertical scale up	Horizontal scale down/ vertical 100% scale up	Horizontal 100% scale up/ vertical scale-up
	Horizontal scale up	Vertical scale down	Horizontal 100% scale up/ vertical scale down	Horizontal scale up/ vertical 100% scale up
	Horizontal scale up	Vertical scale up	Horizontal 100% scale up/ vertical 100% scale up	Horizontal scale up/ vertical scale up
90° rotation	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale down	Horizontal scale down/ vertical scale down	Horizontal 100% scale up/ vertical 100% scale up
	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale up	Horizontal scale down/ vertical 100% scale up	Horizontal scale up/ vertical 100% scale up
	(Horizontal input → vertical output) scale up	(Vertical input → horizontal output) scale up	Horizontal 100% scale up/ vertical 100% scale up	Horizontal scale up/ vertical scale up
180° rotation	Horizontal scale down	Vertical scale down	Horizontal scale down/ vertical scale down	Horizontal 100% scale up/ vertical 100% scale up
	Horizontal scale down	Vertical scale up	Horizontal scale down/ vertical 100% scale up	Horizontal 100% scale up/ vertical scale up
	Horizontal scale up	Vertical scale down	Horizontal 100% scale up/ vertical scale down	Horizontal scale up/ vertical 100% scale up
	Horizontal scale up	Vertical scale up	Horizontal 100% scale up/ vertical 100% scale up	Horizontal scale up/ vertical scale up

Table 36.10 Rotation and Scaling Process

Rotation	Horizontal Scaling	Vertical Scaling	Scaling-Down Control Block	Scaling-Up Control Block
270° rotation	(Horizontal input→vertical output) scale down	(Vertical input→horizontal output) scale down	Horizontal scale down/ vertical scale down	Horizontal 100% scale up/ vertical 100% scale up
	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale up	Horizontal scale down/ vertical 100% scale up	Horizontal scale up/ vertical 100% scale up
	(Horizontal input → vertical output) scale up	(Vertical input → horizontal output) scale up	Horizontal 100% scale up/ vertical 100% scale up	Horizontal scale up/ vertical scale up

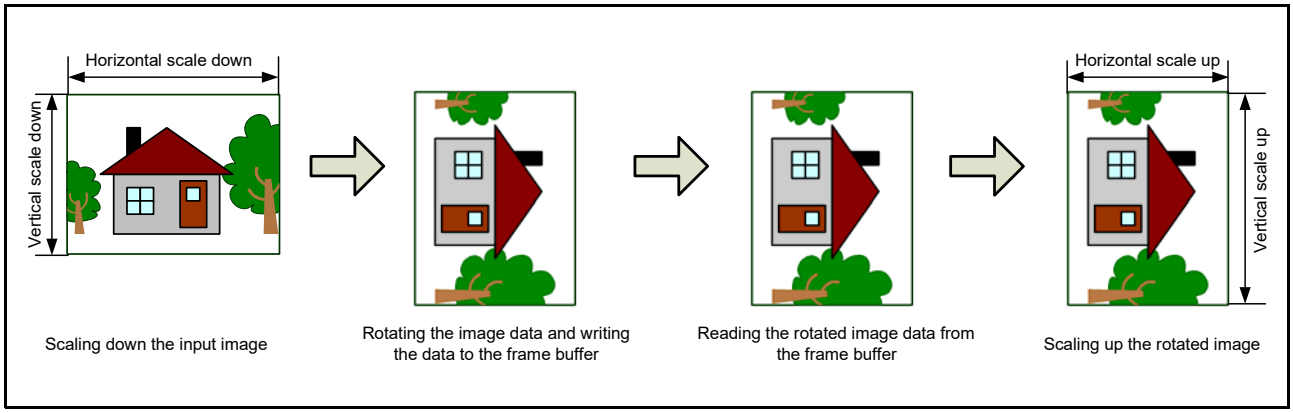


Figure 36.9 Rotation and Scaling Process

It is impossible to use vertical reduction by the scaling-down control block and vertical enlargement by the scaling-up control block simultaneously because they are mutually exclusive. Thus, the following scaling processes cannot be performed with 90° rotation or 270° rotation.

Table 36.11 Impossible Scaling Process

Rotation	Horizontal Scaling	Vertical Scaling	Scaling-Down Control Block	Scaling-Up Control Block
90° rotation 270° rotation	(Horizontal input → vertical output) scale up	(Vertical input → horizontal output) scale down	Horizontal 100% scale up/ vertical scale down	Horizontal 100% scale up/ vertical scale up

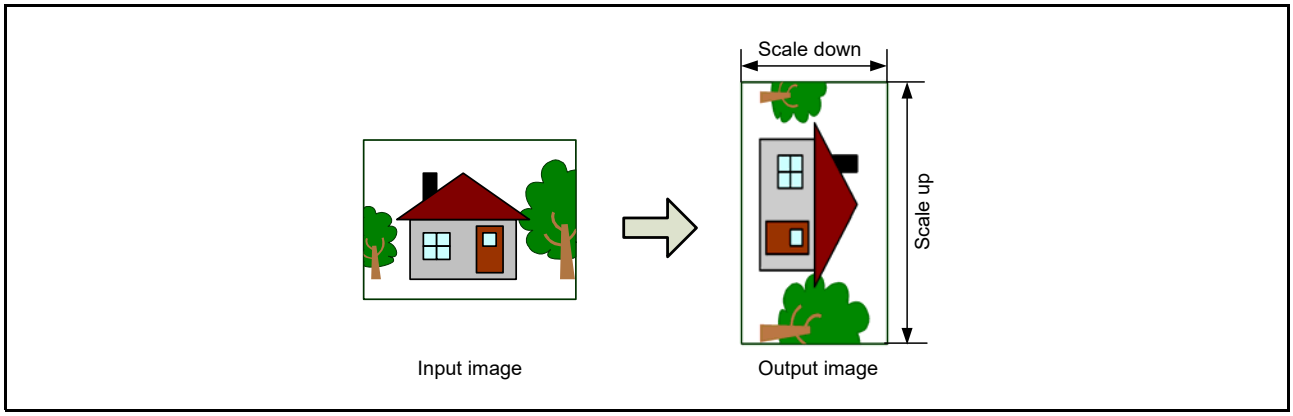


Figure 36.10 Impossible Scaling Process

36.1.6 Horizontal Prefilter

The horizontal prefilter can be turned on or off for brightness (Y) and RGB signals to suppress the frequency band of the signals during horizontal size reduction. The input format depends on the SC_RES_MD[1:0] bit setting in the writing mode register (SC_SCL1_WR1).

When the horizontal reduction ratio is high and there is too much folding frequency component to ignore, the horizontal prefilter should be turned on.

Table 36.12 Horizontal Prefilter Settings

Input Format	SC_RES_PFIL_SEL	Operation
YCbCr input	1	Turns on the horizontal prefilter for Y signal and turns off the horizontal prefilter for Cb/Cr signal.
	0	Turns off the horizontal prefilter.
RGB input	1	Turns on the horizontal prefilter for RGB signal.
	0	Turns off the horizontal prefilter.

Table 36.13 Horizontal Prefilter Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_DS4	SC_RES_PFIL_SEL	0	Prefilter Mode Select for Brightness Signals 0: The prefilter is turned off. 1: The prefilter is turned on. (1/4 + 1/2 + 1/4)

36.1.7 Horizontal Scale-Down

The number of horizontally arranged pixels can be decreased at a desired ratio in the range of 1/1 to 1/8 using pixel conversion.

For the scaling filter, either hold or linear interpolation mode can be selected.

(1) One-TAP Hold Interpolation

When the interpolation position is between input pixels X_n and X_{n+1}, the $X_{interpo}$ interpolation value is defined as follows.

$X_{interpo} = X_n$

(2) Two-TAP Linear Interpolation

When the interpolation position is between input pixels X_n and X_{n+1}, the $X_{interpo}$ interpolation value is defined as follows based on the interpolation position "phase".

$X_{interpo} = (X_n \times (4096 - phase) + X_{n+1} \times phase) / 4096$

(3) Calculation of Horizontal Scale Down Ratio

The value to be set to the horizontal scale-down ratio SC_RES_DS_H_RATIO can be obtained using the following equation based on the number of input pixels SC_RES_HW and number of output pixels SC_RES_OUT_HW, where the decimals are rounded off.

SC_RES_DS_H_RATIO = round (SC_RES_HW ÷ SC_RES_OUT_HW × 4096)

Note that, for 100% horizontal scale-up, the SC_RES_HW and SC_RES_OUT_HW values should be identical and the SC_RES_DS_H_RATIO bits should be set to 4096.

(4) Handling for Lack of Last-Input Pixel

Interpolation is carried out between the second-last-input and last-input pixels to produce the last-output pixel at the right end of a screen. The interpolation position of the last-output pixel may be close to the second-last-input pixel depending on the horizontal scale-down ratio; in this case, it may appear that the last-input pixel is lacking.

The undesirable influence by lack of last-input pixel can be decreased by appropriately adjusting the horizontal scale-down ratio using the following equations.

Pre-adjustment horizontal scale-down ratio RATIO_org should be calculated first to find adjustment value σ, and then scale-down ratio SC_RES_DS_H_RATIO should be determined.

RATIO_org = round (SC_RES_HW ÷ SC_RES_OUT_HW × 4096)
σ = (RATIO_org × (SC_RES_OUT_HW − 1) − (SC_RES_HW − 1) × 4096) ÷ (SC_RES_OUT_HW − 1)
SC_RES_DS_H_RATIO = roundup (RATIO_org − σ)

Table 36.14 Horizontal Scale Down Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_DS1	SC_RES_DS_H_ON	1	Horizontal Scale Down On/Off 0: Off 1: On
SC_SCL0_DS7	SC_RES_OUT_HW[10:0]	640	Number of Valid Horizontal Pixels Output by Scaling-down Control Block (Video-image clock cycles)
SC_SCL0_DS4	SC_RES_DS_H_INTERPOTYP	1	Horizontal Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
SC_SCL0_DS4	SC_RES_DS_H_RATIO[15:0]	9224	Horizontal Scale Down Ratio ([15:12]: Integer part, [11:0]: Decimal part) round(SC_RES_HW ÷ SC_RES_OUT_HW × 4096) SC_RES_DS_H_RATIO < 4096: Setting prohibited SC_RES_DS_H_RATIO = 4096: 100% scale up SC_RES_DS_H_RATIO > 4096: Scale down

Note: The SC_RES_OUT_HW value should be aligned in 4-pixel units and equal to or smaller than the SC_RES_HW value.

36.1.8 Vertical Scale-Down

The number of lines can be decreased in the vertical direction at a desired ratio in the range of 1/1 to 1/8 using pixel conversion.

For the scaling filter, either hold or linear interpolation mode can be selected.

(1) One-TAP Hold Interpolation

When the interpolation position is between input lines X_n and X_{n+1}, the $X_{interpo}$ interpolation value is defined as follows.

$X_{interpo} = X_n$

(2) Two-TAP Linear Interpolation

When the interpolation position is between input lines X_n and X_{n+1}, the $X_{interpo}$ interpolation value is defined as follows based on the interpolation position "phase".

$X_{interpo} = (X_n \times (4096 - phase) + X_{n+1} \times phase) / 4096$

(3) Calculation of Vertical Scale Down Ratio

The value to be set to the vertical scale-down ratio SC_RES_V_RATIO can be obtained using the following equation based on the number of input lines SC_RES_VW and number of output lines SC_RES_OUT_VW, where the decimals are rounded off.

SC_RES_V_RATIO = round (SC_RES_VW ÷ SC_RES_OUT_VW × 4096)

Note that the SC_RES_VW and SC_RES_OUT_VW values should be identical for vertical enlargement or 100% vertical enlargement.

For 100% vertical enlargement, reduction is carried out assuming SC_RES_V_RATIO as 4096.

(4) Handling for Lack of Last-Input Line

Interpolation is carried out between the second-last-input and last-input lines to produce the last-output line at the lower end of a screen. The interpolation position of the last-output line may be close to the second-last-input line depending on the vertical scale-down ratio; in this case, it may appear that the last-input line is lacking.

The undesirable influence by the lack of last-input line can be decreased by appropriately adjusting the vertical scale-down ratio using the following equations.

Pre-adjustment vertical scale-down ratio RATIO_org should be calculated first to find adjustment value σ, and then scale-down ratio SC_RES_V_RATIO should be determined.

RATIO_org = round (SC_RES_VW ÷ SC_RES_OUT_VW × 4096)
σ = (RATIO_org × (SC_RES_OUT_VW − 1) − (SC_RES_VW − 1) × 4096) ÷ (SC_RES_OUT_VW − 1)
SC_RES_V_RATIO = round (RATIO_org − σ)

Table 36.15 Vertical Scale Down Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_DS1	SC_RES_DS_V_ON	1	Vertical Scale Down On/Off 0: Off 1: On
SC_SCL0_DS7	SC_RES_OUT_VW [10:0]	240	Number of Valid Lines in Vertical Direction Output by Scaling-Down Control Block (lines) This bit setting is used for the number of lines to be written to the frame buffer. When SC_SCL1_WR1.SC_RES_LOOP is 0 (frame write mode), these bits specify the number of lines for one frame. When SC_SCL1_WR1.SC_RES_LOOP is 1 (line write mode), these bits specify the number of lines for writing in a ring configuration.
SC_SCL0_DS5	SC_RES_V_ INTERPOTYP	1	Vertical Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
SC_SCL0_DS6	SC_RES_V_RATIO [15:0]	2044	Vertical Scale UP/Down Ratio ([15:12]: Integer part, [11:0]: Decimal part) For scale down: round(SC_RES_VW ÷ SC_RES_OUT_VW × 4096) For scale up: round(SC_RES_IN_VW ÷ SC_RES_P_VW × 4096) SC_RES_V_RATIO < 4096: Scale up SC_RES_V_RATIO = 4096: 100% scale up SC_RES_V_RATIO > 4096: Scale down

Note: SC_RES_V_RATIO and SC_RES_V_INTERPOTYP are both shared by vertical reduction and vertical enlargement.
It is impossible to use vertical reduction and vertical enlargement simultaneously because they are mutually exclusive.
The SC_RES_OUT_VW value should be aligned in 4-line units and equal to or smaller than the SC_RES_VW value.

36.1.9 Horizontal Scale Up

The number of horizontally arranged pixels can be increased at a desired ratio in the range of 1/1 to 8/1 using pixel conversion.

For the scaling filter, either hold or linear interpolation mode can be selected.

(1) One-TAP Hold Interpolation

When the interpolation position is between input pixels X_n and X_{n+1}, the $X_{interpo}$ interpolation value is defined as follows.

$X_{interpo} = X_n$

(2) Two-TAP Linear Interpolation

When the interpolation position is between input pixels X_n and X_{n+1}, the $X_{interpo}$ interpolation value is defined as follows based on the interpolation position "phase".

$X_{interpo} = (X_n \times (4096 - phase) + X_{n+1} \times phase) / 4096$

(3) Calculation of Horizontal Scale Up Ratio

The value to be set to the horizontal scale-up ratio SC_RES_US_H_RATIO can be obtained using the following equation based on the number of input pixels SC_RES_IN_HW and number of output pixels SC_RES_P_HW, where the decimals are rounded off.

SC_RES_US_H_RATIO = round (SC_RES_IN_HW ÷ SC_RES_P_HW × 4096)

Note that, for 100% horizontal scale-up, the SC_RES_IN_HW and SC_RES_P_HW values should be identical and the SC_RES_US_H_RATIO bits should be set to 4096.

(4) Folding Handling

Since interpolation is carried out between the last-input pixel and second-last-input folding pixel to produce the last-output pixel at the right end of a screen, folding may undesirably stand out depending on the horizontal scale up ratio.

The undesirable influence by folding pixels can be decreased by appropriately adjusting the horizontal scale-up ratio using the following equations.

Pre-adjustment horizontal scale-up ratio RATIO_org should be calculated first to find adjustment value σ, and then scale-up ratio SC_RES_US_H_RATIO should be determined.

RATIO_org = round (SC_RES_IN_HW ÷ SC_RES_P_HW × 4096)
σ = (RATIO_org × (SC_RES_P_HW − 1) − (SC_RES_IN_HW − 1) × 4096 ÷ (SC_RES_P_HW − 1)
SC_RES_US_H_RATIO = round (RATIO_org − σ)

Table 36.16 Horizontal Scale Up Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_US1	SC_RES_US_H_ON	1	Horizontal Scale Up On/Off 0: Off 1: On
SC_SCL0_US4	SC_RES_IN_HW[10:0]	640	Number of Valid Horizontal Pixels Input to Scaling-up Control Block (Pixel-clock cycles)
SC_SCL0_US6	SC_RES_US_H_ INTERPOTYP	1	Horizontal Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
SC_SCL0_US5	SC_RES_US_H_RATIO [15:0]	9224	Horizontal Scale Up Ratio ([15:12]: Integer part, [11:0]: Decimal part) round(SC_RES_IN_HW ÷ SC_RES_P_HW × 4096) SC_RES_US_H_RATIO < 4096: Scale up SC_RES_US_H_RATIO = 4096: 100% scale-up SC_RES_US_H_RATIO > 4096: Setting prohibited

36.1.10 Vertical Scale-Up

The number of lines can be increased in the vertical direction at a desired ratio in the range of 1/1 to 8/1 using pixel conversion.

For the scaling filter, either hold or linear interpolation mode can be selected.

(1) One-TAP Hold Interpolation

When the interpolation position is between input lines X_n and X_{n+1}, the $X_{interpo}$ interpolation value is defined as follows.

$$X_{interpo} = X_n$$

(2) Two-TAP Linear Interpolation

When the interpolation position is between input lines X_n and X_{n+1}, the $X_{interpo}$ interpolation value is defined as follows based on the interpolation position "phase".

$$X_{interpo} = (X_n \times (4096 - phase) + X_{n+1} \times phase) / 4096$$

(3) Calculation of Vertical Scale Up Ratio

The value to be set to the vertical scale-up ratio SC_RES_V_RATIO can be obtained using the following equation based on the number of input lines SC_RES_IN_VW and number of output lines SC_RES_P_VW, where the decimals are rounded off.

SC_RES_V_RATIO = round (SC_RES_IN_VW ÷ SC_RES_P_VW × 4096)

Note that, for 100% vertical enlargement or vertical reduction, the SC_RES_IN_VW and SC_RES_P_VW values should be identical.

(4) Folding Handling

The last line to be output at the bottom of the screen is produced by interpolation between the last line and line for folding (second-last line to be input). According to the vertical scale-up rate, this may cause folding to stand out.

The undesirable influence by folding lines can be decreased by appropriately adjusting the vertical scale-up ratio using the following equations.

Pre-adjustment vertical scale-up ratio RATIO_org should be calculated first to find adjustment value σ, and then scale-up ratio SC_RES_V_RATIO should be determined.

RATIO_org = round (SC_RES_IN_VW ÷ SC_RES_P_VW × 4096)
σ = (RATIO_org × (SC_RES_P_VW − 1) − (SC_RES_IN_VW − 1) × 4096) ÷ (SC_RES_P_VW − 1)
SC_RES_V_RATIO = round (RATIO_org − σ)

Table 36.17 Vertical Scale Up Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_US1	SC_RES_US_V_ON	1	Vertical Scale Up On/Off 0: Off 1: On
SC_SCL0_US4	SC_RES_IN_VW[10:0]	240	Number of Valid Lines in Vertical Direction Input to Scaling-up Control Block (Lines)
SC_SCL0_DS5	SC_RES_V_INTERPOTYP	1	Vertical Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
SC_SCL0_DS6	SC_RES_V_RATIO[15:0]	2044	Vertical Scale Up Ratio ([15:12]: Integer part, [11:0]: Decimal part) For scale down: round(SC_RES_VW ÷ SC_RES_OUT_VW × 4096) For scale up: round(SC_RES_IN_VW ÷ SC_RES_P_VW × 4096) SC_RES_V_RATIO < 4096: Scale up SC_RES_V_RATIO = 4096: 100% scale up SC_RES_V_RATIO > 4096: Scale down

Note: SC_RES_V_RATIO and SC_RES_V_INTERPOTYP are both shared by vertical reduction and vertical enlargement.
It is impossible to use vertical reduction and vertical enlargement simultaneously because they are mutually exclusive.

36.1.11 IP Conversion

(1) Initial Phase Control

When interlaced signals are input, line flickering caused by the line offset between the top and bottom fields can be decreased before being displayed by independently adjusting the initial scaling phases of the fields.

For various operations, appropriate settings should be made referring to the relevant registers as listed in Table 36.18.

Table 36.18 Initial Scaling Phase Settings (Standard Values) for IP Conversion

Rotation	Horizontal Scaling	Vertical Scaling	Reference Bit (Setting)
Normal	Horizontal scale down	Vertical scale down	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale down	Vertical scale up	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale up	Vertical scale down	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale up	Vertical scale up	SC_RES_TOP_INIPHASE = 2048
Horizontal mirroring	Horizontal scale down	Vertical scale down	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale down	Vertical scale up	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale up	Vertical scale down	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale up	Vertical scale up	SC_RES_TOP_INIPHASE = 2048
90° rotation	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale down	SC_RES_TOP_INIPHASE = 2048
	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale up	SC_RES_TOP_INIPHASE = 2048
	(Horizontal input → vertical output) scale up	(Vertical input → horizontal output) scale up	SC_RES_US_HB_INIPHASE = 2048
180° rotation	Horizontal scale down	Vertical scale down	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale down	Vertical scale up	SC_RES_BTM_INIPHASE = 2048
	Horizontal scale up	Vertical scale down	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale up	Vertical scale up	SC_RES_BTM_INIPHASE = 2048
270° rotation	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale down	SC_RES_TOP_INIPHASE = 2048
	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale up	SC_RES_TOP_INIPHASE = 2048
	(Horizontal input → vertical output) scale up	(Vertical input → horizontal output) scale up	SC_RES_US_HT_INIPHASE = 2048

Note: Set 0 to the initial phase control registers where the specific value is not shown in the table.
Set 0 to all the initial phase control registers when progressive signals are input.

Table 36.19 Initial Scaling Phase Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_DS5	SC_RES_BTM_INIPHASE [11:0]	0	Vertical Interpolation Start Phase for Bottom Field 0 to 4095 (0 to approx. 1.0)
SC_SCL0_DS5	SC_RES_TOP_INIPHASE [11:0]	2048	Vertical Interpolation Start Phase for Top Field 0 to 4095 (0 to approx. 1.0)
SC_SCL0_US6	SC_RES_US_HB_INIPHASE [11:0]	0	Horizontal Interpolation Start Phase for Bottom Field 0 to 4095 (0 to approx. 1.0)
SC_SCL0_US6	SC_RES_US_HT_INIPHASE [11:0]	0	Horizontal Interpolation Start Phase for Top Field 0 to 4095 (0 to approx. 1.0)

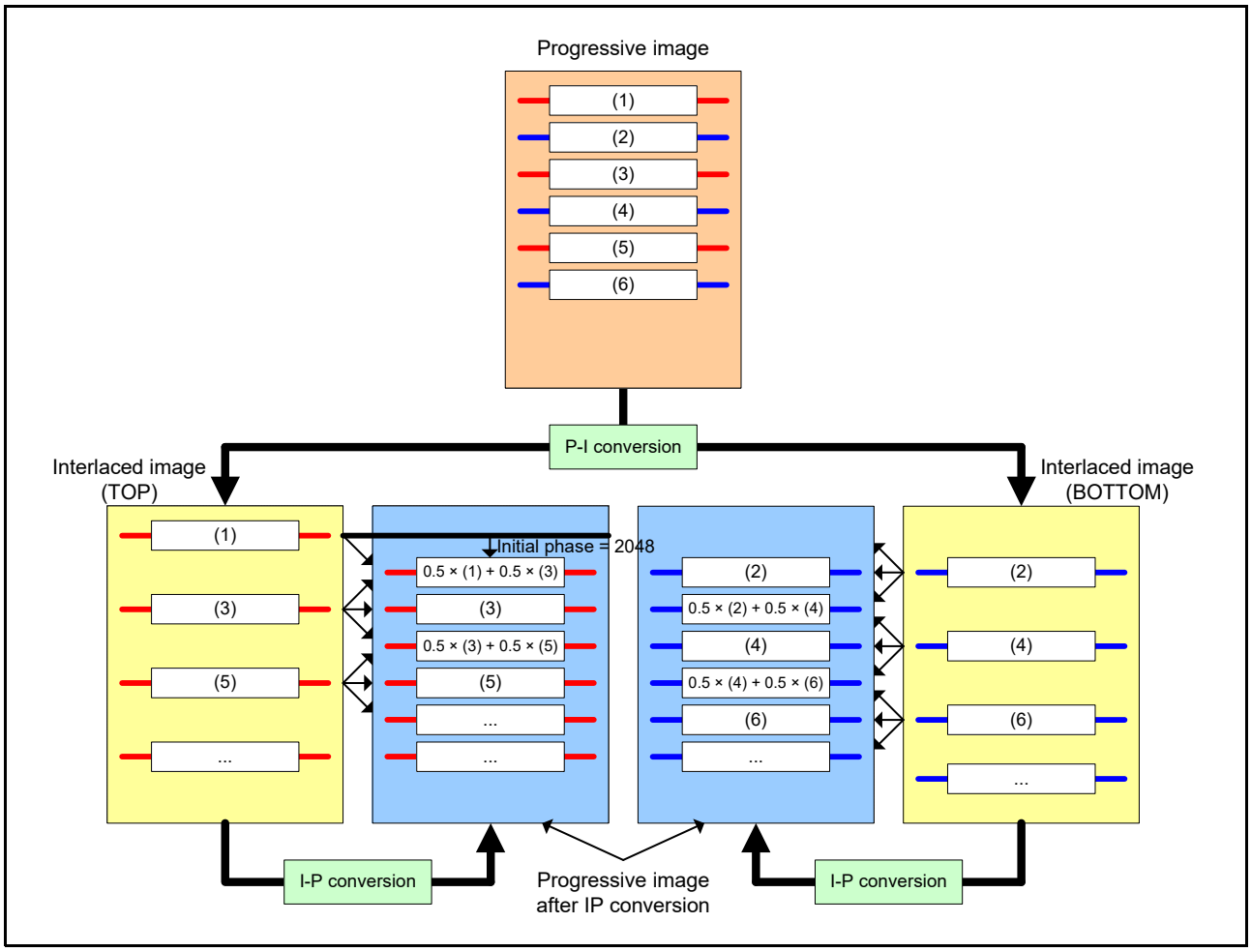


Figure 36.11 IP Conversion Processing Schematic Diagram

(2) Field Determination Signal Control

When interlaced signals are input, the field determination signal can be controlled, which is output to the scaling-up control block during vertical scaling.

When progressive signals are input or vertical scaling is carried out by the scaling-down control block, the field determination signal output to the scaling-up control block is fixed to the specific level, and thus either 0 or 1 can be set to the SC_RES_FLD_DLY_SEL bit.

Table 36.20 Settings for Field Determination Signal Control

Input Signal	Rotation	Vertical Processing	Frame Buffer	SC_RES_FLD_DLY_SEL
Progressive	—	—	—	—
Interlace	Normal Horizontal mirroring 180° rotation	Vertical scale down	—	—
		Vertical scale up	One plane or less	0
			Two planes or more	1
	90° rotation 270° rotation	(Horizontal input → vertical output) scale down	—	—
		(Horizontal input → vertical output) scale up	Two planes or more	1

Table 36.21 Field Determination Signal Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC5	SC_RES_FLD_DLY_SEL	1	Field Determination Signal Delay Control 0: No delay 1: Delay of one vertical cycle

36.1.12 Control of Interrupt on Specified Image Line before Scaling-down, and Reading of Current Image Line before Scaling-down

When the location of the image line input to the scaling-down control block matches the SC_SCL1_LINE setting, an interrupt processing is done. In addition, the current location of the line input to the scaling-down control block can be read from a register.

Table 36.22 Control of Interrupt on Specified Image Line before Scaling-down, and Reading of Current Image line before Scaling-down

Register Name	Bit Name	Initial Value	Description
SC_SCL0_INT	SC_RES_LINE[10:0]	All 0	Setting of Interrupt on Image Line Input to Scaling-down Control Block When the location of the image line input to the scaling-down control block matches the SC_SCL0_LINE setting, an interrupt signal is output. However, setting this is prohibited in this product.
SC_SCL0_MON0	SC_RES_LIN_STAT[10:0]	All 0	Current Location of Image Line Input to Scaling-down Control Block

36.1.13 Trimming

The upper, lower, right, and left parts of a post-scaling image can be trimmed off as specified by the SC_RES_VCUT and SC_RES_HCUT bits before being output.

The frame lines of the post-scaling image can also be displayed by setting the SC_RES_DISP_ON bit to 1.

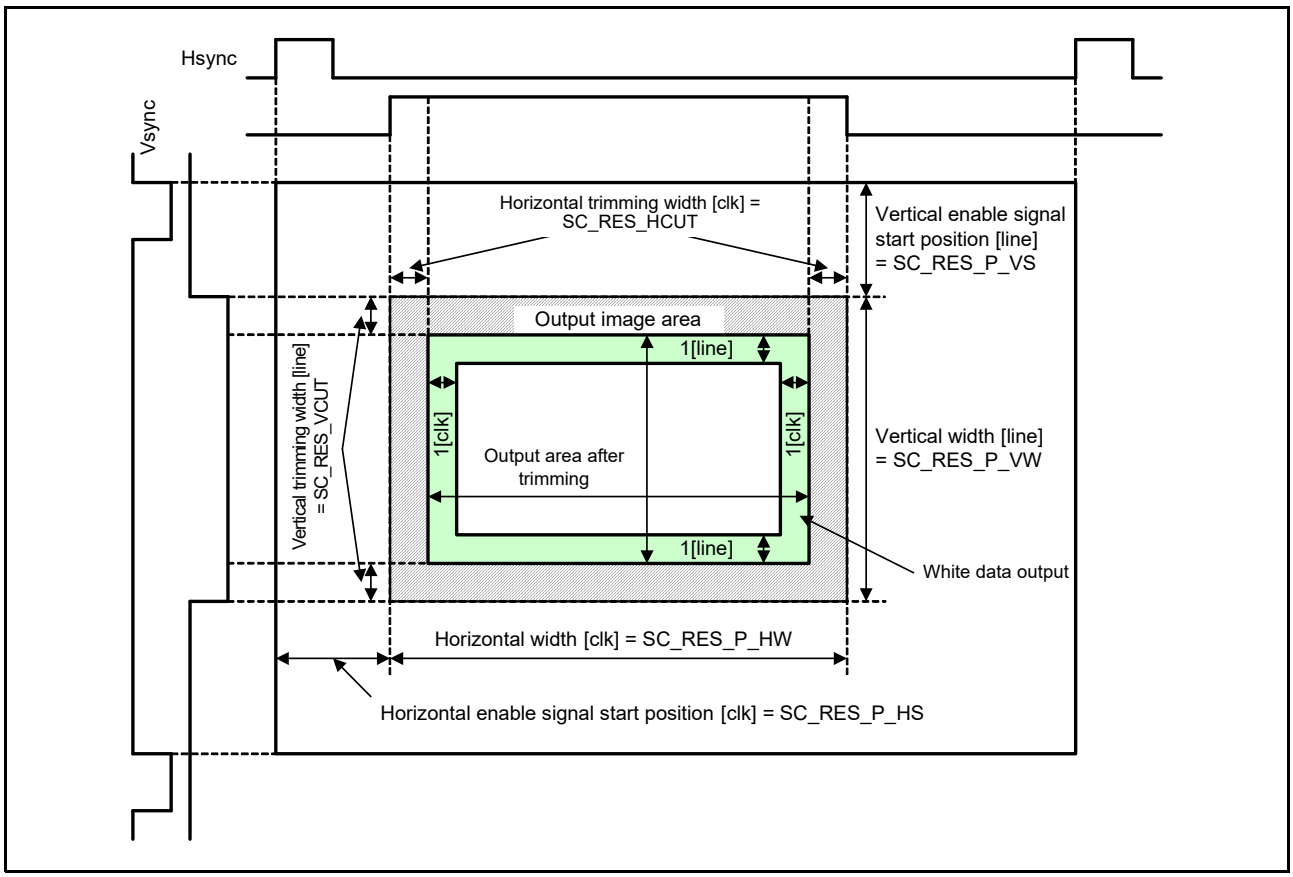


Figure 36.12 Area Relationship for Trimming (Frame Lines Displayed)

Table 36.23 Trimming Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_US7	SC_RES_HCUT[7:0]	0	Horizontal Amount of Cut-off Post-Scaling Image (Right and Left Parts) Sets the number of pixel-clock cycles.
SC_SCL0_US7	SC_RES_VCUT[7:0]	0	Vertical Amount of Cut-off Post-Scaling Image (Upper and Lower Parts) Sets the number of lines.
SC_SCL0_US8	SC_RES_DISP_ON	0	Post-Scaling Image Frame Display On/Off 0: Frame display on 1: Frame display off

36.1.14 Screen Synthesis

During the valid full-screen period, the image area can be overlaid before being output. If the image area to be output is smaller than a full-screen, the background color specified by the SC_RES_BK_COL_R, SC_RES_BK_COL_G, and SC_RES_BK_COL_B bits are displayed to fill the background.

Table 36.24 Screen Synthesis Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_OVR1	SC_RES_BK_COL_R [7:0]	128	Background Color Setting R/Cr Signal R: 8 bits; unsigned (0 to 255 [LSB]) Cr: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])
SC_SCL0_OVR1	SC_RES_BK_COL_B [7:0]	128	Background Color Setting B/Cb Signal B: 8 bits; unsigned (0 to 255 [LSB]) Cb: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])
SC_SCL0_OVR1	SC_RES_BK_COL_G [7:0]	0	Background Color Setting G/Y Signal G/Y: 8 bits; unsigned (0 to 255 [LSB])

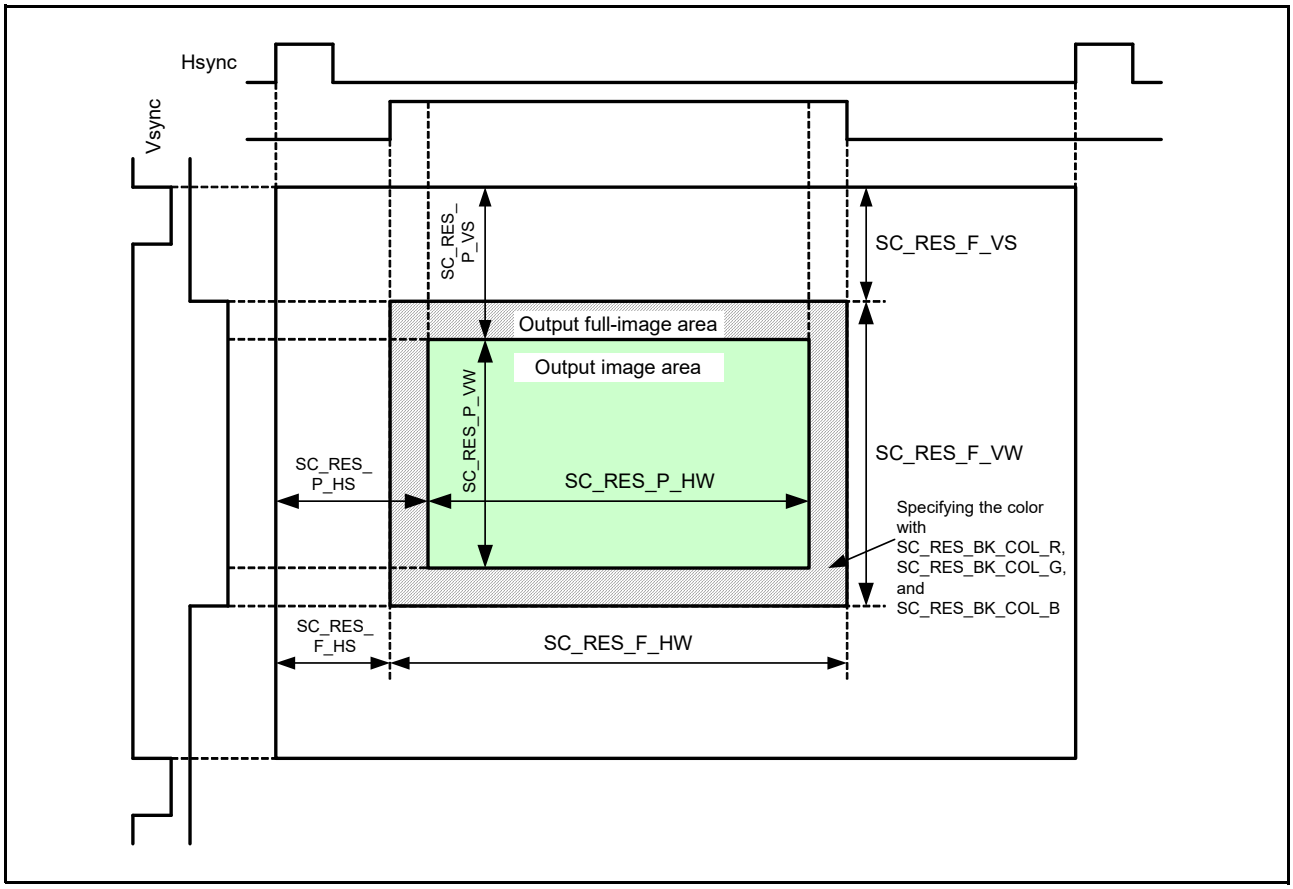


Figure 36.13 Area Relationship with Output Image Size Smaller than a Full Screen

36.1.15 Selecting Format for Writing Video Image Signals to Frame Buffer

A format can be selected for writing video image signals to the frame buffer.

Although 24-bit YCbCr signals or 24-bit RGB signals are input to the scaling control block, they are converted into 16-bit YCbCr422 signals, 16-bit RGB565 signals, 32-bit YCbCr444 signals, or 32-bit RGB888 signals before being written to the frame buffer.

As bit reduction processing of RGB565, rounding off or 2 × 2 pattern dither can be selected with the SC_RES_DTH_ON bit. For details on pattern dither, see section 39.1.7, Dither Process in section 39, Video Display Controller 6 (7): Output Controller.
Input YCbCr signals are converted into YCbCr422 signals and output to the image renderer. For distortion correction, refer to section 42, Image Renderer (IMR-LS2).

Table 36.25 Frame Buffer Writing Mode Setting

RES_BITDEC_ON	SC_RES_MD[1:0]	Writing Mode
0	3	YCbCr444 (normal, horizontal mirroring)
0	2	RGB888 (normal, horizontal mirroring)
1	1	RGB565 (normal, horizontal mirroring, rotation)
*	0	YCbCr422 (normal, horizontal mirroring, rotation), YCbCr422 (image renderer)

Table 36.26 Video Signal Writing Format Selection Control

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR1	SC_RES_MD[1:0]	0	Frame Buffer Video-Signal Writing Format 0: YCbCr422 (16 bits) 1: RGB565 (16 bits) 2: RGB888 (24 (32) bits) 3: YCbCr444 (24 (32) bits)
SC_SCL1_WR6	SC_RES_BITDEC_ON	0	Bit Reduction On/Off 0: Off 1: On
SC_SCL1_WR6	SC_RES_DTH_ON	0	Dither Correction On/Off 0: Off (rounded off) 1: On (2 × 2 pattern dither)

36.1.16 Horizontal Mirroring and Rotation

Horizontal mirroring and rotation can be carried out for scaled-down images before being written to the frame buffer.

Table 36.27 and Table 36.28 show the relationship between various writing modes for image processing and video signals.

Table 36.27 Relationship between Writing Modes and Video Signals

RES_DS_WR_MD[2:0]	Writing Modes	YCbCr444	YCbCr422	RGB565	RGB888
0	Normal writing	Enabled	Enabled	Enabled	Enabled
1	Horizontal mirroring	Enabled	Enabled	Enabled	Enabled
2	90° rotation	Disabled	Enabled	Enabled	Disabled
3	180° rotation	Disabled	Enabled	Enabled	Disabled
4	270° rotation	Disabled	Enabled	Enabled	Disabled
5 to 7	Setting prohibited	—	—	—	—

Table 36.28 Horizontal Mirroring and Rotation Control

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR1	SC_RES_DS_WR_MD [2:0]	0	Frame Buffer Writing Mode for Image Processing 0: Normal 1: Horizontal mirroring 2: 90° rotation 3: 180° rotation 4: 270° rotation 5 to 7: Setting prohibited

36.1.17 Writing to Frame Buffer

(1) Frame Buffer Transfer Mode

Either 32-byte or 128-byte transfer mode can be selected for accessing the frame buffer in which video image data and graphics data are stored.

Table 36.29 Frame Buffer Transfer Mode

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR1	SC_RES_BST_MD	0	Transfer Burst Length for Frame Buffer Writing 0: 32-byte 1: 128-byte

(2) Frame Buffer Write Control

Frame buffer writing is enabled or disabled.

Table 36.30 Frame Buffer Writing Control

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR5	SC_RES_WENB	0	Frame Buffer Write Enable After making the setting to enable writing, writing starts from the second frame. 0: Frame buffer writing is disabled. 1: Frame buffer writing is enabled.

(3) Frame Buffer Writing Rate Selection

A frame buffer writing rate can be selected from among 1/1, 1/2, 1/4, and 1/8 the vertical frequency of the input signal.

When 1/2, 1/4, or 1/8 is selected, either the top or bottom field can be selected for writing.

Table 36.31 Frame Buffer Write Control

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR5	SC_RES_FS_RATE [1:0]	0	Writing Rate Sets the frame buffer writing rate to the vertical frequency of the input signal. 0: 1/1 an input signal (The SC_RES_FLD_SEL setting is invalid.) 1: 1/2 an input signal 2: 1/4 an input signal 3: 1/8 an input signal
SC_SCL1_WR5	SC_RES_FLD_SEL	0	Write Field Select 0: Top field 1: Bottom field
SC_SCL1_WR5	SC_RES_INTER	1	Field Operating Mode Select 0: Progressive 1: Interlace

(4) Frame Buffer Write Addresses

Frame buffer addresses are specified using the base address, line offset address, frame offset address, data size of a line, and the number of lines in a frame. When an interlaced video image is input, the top and bottom field data can be separately stored in the frame buffer.

The SC_RES_BASE[31:0], SC_RES_LN_OFF[14:0], and SC_RES_FLM_OFF[22:0] bits should be set in 32-byte units (the lower five bits should be fixed to 0).

For 128-byte transfer, bits [6:5] in the address control registers should be fixed to 0 since addresses should be specified in 128-byte units.

For the data size of a line and the number of lines in a frame, the relevant register values set for the scaling-down control block are used.

Table 36.32 Frame Buffer Write Address Control

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR1	SC_RES_TB_ADD_MOD	0	Top and Bottom Data Write Address Specification Method 0: A write address is specified in common for top and bottom data. 1: Separate write addresses are specified for top and bottom data.
SC_SCL1_WR2	SC_RES_BASE [31:0]	0	Frame Buffer Base Address Sets the start address of the frame buffer to store the frame data for the top field when SC_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC_RES_TB_ADD_MOD = 0. For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.
SC_SCL1_WR8	SC_RES_BASE_B [31:0]	0	Frame Buffer Base Address for Bottom Sets the start address of the frame buffer to store the frame data for the bottom field when SC_RES_TB_ADD_MOD = 1. For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.
SC_SCL1_WR3	SC_RES_LN_OFF [14:0]	2048	Frame Buffer Line Offset Address Sets the line offset address for calculating the line start address for the top field when SC_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC_RES_TB_ADD_MOD = 0. Line 0: SC_RES_BASE Line 1: SC_RES_BASE + SC_RES_LN_OFF × 1 : Line n: SC_RES_BASE + SC_RES_LN_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.
SC_SCL1_WR9	SC_RES_LN_OFF_B [14:0]	2048	Frame Buffer Line Offset Address for Bottom Sets the line offset address for calculating the line start address for the bottom field when SC_RES_TB_ADD_MOD = 1. Line 0: SC_RES_BASE_B Line 1: SC_RES_BASE_B + SC_RES_LN_OFF_B × 1 : Line n: SC_RES_BASE_B+ SC_RES_LN_OFF_B × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.
SC_SCL1_WR4	SC_RES_FLM_OFF [22:0]	524288	Frame Buffer Frame Offset Address Sets the frame offset address for calculating the start address of each frame for the top field when SC_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC_RES_TB_ADD_MOD = 0. Buffer 0: SC_RES_BASE Buffer 1: SC_RES_BASE + SC_RES_FLM_OFF × 1 : Buffer n: SC_RES_BASE + SC_RES_FLM_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.

Table 36.32 Frame Buffer Write Address Control

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR10	SC_RES_FLM_OFF_B [22:0]	524288	Frame Buffer Frame Offset Address for Bottom Sets the frame offset address for calculating the start address of each frame for the bottom field when SC_RES_TB_ADD_MOD = 1. Buffer 0: SC_RES_BASE_B Buffer 1: SC_RES_BASE_B + SC_RES_FLM_OFF_B × 1 : Buffer n: SC_RES_BASE_B + SC_RES_FLM_OFF_B × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.

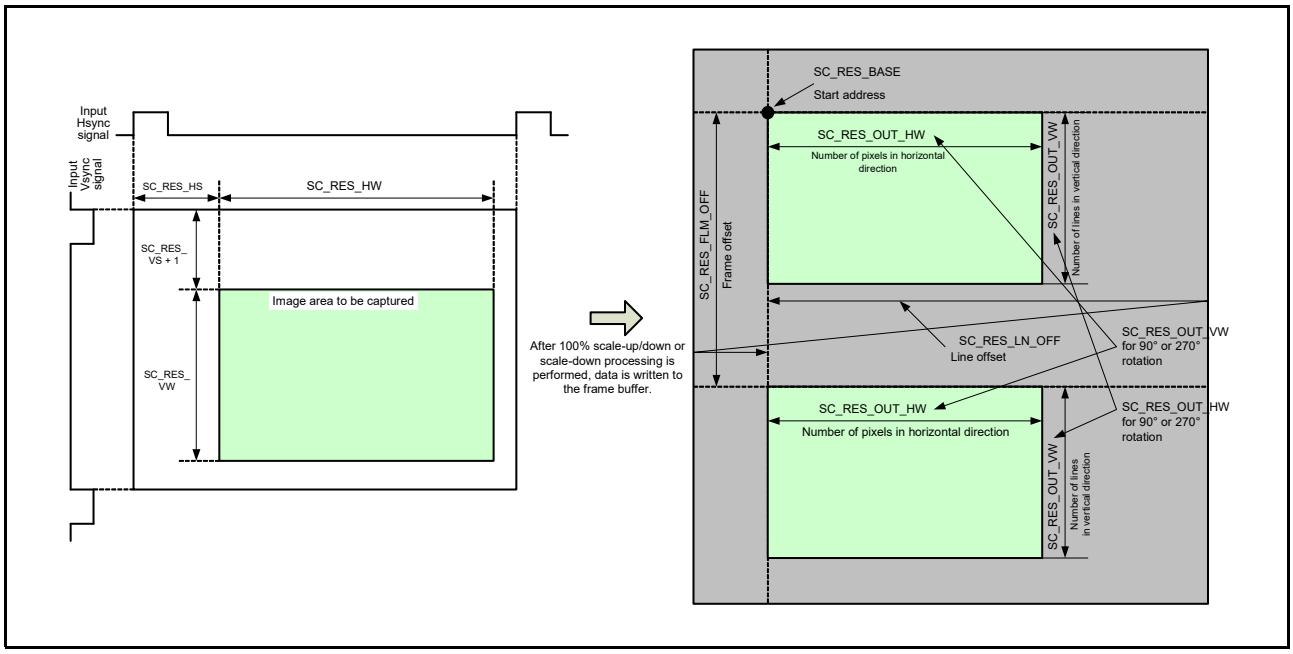


Figure 36.14 Data Arrangement in Frame Buffer

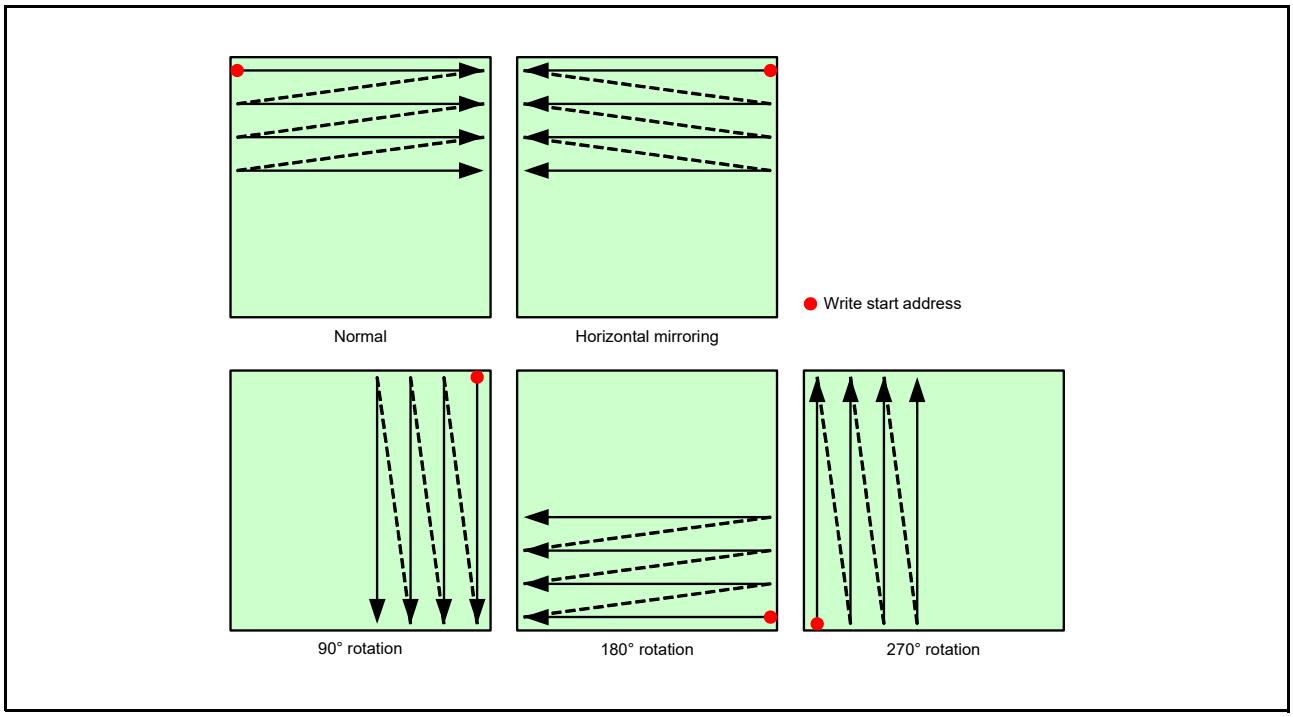


Figure 36.15 Data Arrangement in Frame Buffer in Various Writing Modes

(5) Frame Buffer Management

The scaling control block can handle multiple frames as the frame buffer.

Data is written to the buffer in cyclic mode according to the number of frames specified by the SC_RES_FLM_NUM bits.

For rotation, the SC_RES_FLM_NUM bits should be set to two or more frames.

To use the frame buffer as the ring buffer in line mode, the SC_RES_FLM_NUM bits should be set to 0 (1 frame) and the SC_RES_LOOP bit to 1.

Table 36.33 Frame Buffer Write Control

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR3	SC_RES_FLM_ NUM[9:0]	1	Number of Frames of Buffer to be Written to Sets the number of frames for the top field when SC_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC_RES_TB_ADD_MOD = 0 Number of frames defined by SC_RES_FLM_NUM + 1 are used.
SC_SCL1_WR9	SC_RES_FLM_ NUM_B[9:0]	1	Number of Frames of Buffer to be Written to for Bottom Field when SC_RES_TB_ADD_MOD = 1 Number of frames defined by SC_RES_FLM_NUM_B + 1 are used.
SC_SCL1_WR1	SC_RES_LOOP	0	Frame Buffer Write Mode Select 0: Frame mode 1: Line mode (read as ring buffer)
SC_SCL1_WR7	SC_RES_FLM_ CNT[9:0]	—	Frame Number of Frame Being Accessed Frame number of the frame being accessed in the top field when SC_RES_TB_ADD_MOD = 1 or that in the top or bottom field when SC_RES_TB_ADD_MOD = 0.
SC_SCL1_WR11	SC_RES_FLM_ CNT_B[9:0]	—	Frame Number of Frame Being Accessed in Bottom Field Frame number of the frame being accessed in the bottom field when SC_RES_TB_ADD_MOD = 1.

(6) Buffer Overflow Handling

If writing to the frame buffer cannot be completed due to bus-traffic related problems, an overflow interrupt can be output to the interrupt controller.

Table 36.34 Buffer Overflow Detection

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR7	SC_RES_OVERFLOW	—	Line Buffer Overflow Detect 1: Line buffer has overflowed. 0: Line buffer has not overflowed.

(7) Frame Buffer Write End Flag

When writing one frame of data to the frame buffer is completed, a frame buffer write end interrupt can be output to the interrupt controller.

36.1.18 Selecting a Scaling-up Process or Graphics 0 Process

Scaling-up process and graphics 0 process are mutually exclusive and thus frame buffer cannot be read out simultaneously for the processes.

When displaying input video image signals or displaying enlarged graphics, data is read from the frame buffer via the scaling-up control block.

However, graphics can be enlarged and displayed by the scaling-up control block only when the RGB565, RGB888, YCbCr422, or YCbCr444 format is used.

When displaying graphics without enlargement, the data is read from the frame buffer via the graphics 0 processing block.

With the SC_RES_IBUS_SYNC_SEL bit, sync signals for reading out the frame buffer and read size setting bits are selected.

Table 36.35 Selection of Scaling-Up Process and Graphics 0 Process

Type of Output Scaling Display	SC_RES_ IBUS_SYNC_SEL	Sync Signals for Frame Buffer Read	Frame Buffer Read Size Setting Bits	Display Enabling Bits
Input video signal display Enlarged graphics display	0	Output from scaling-up control block	SC_RES_IN_VW SC_RES_IN_HW	SC_RES_P_VS SC_RES_P_VW SC_RES_P_HS SC_RES_P_HW
Graphics display	1	Output from graphics 0 processing block	GR_FLM_LNUM* GR_HW*	GR_GRC_VS GR_GRC_VW GR_GRC_HS GR_GRC_HW

Note: * The value set to the register + 1 is the actual read size.

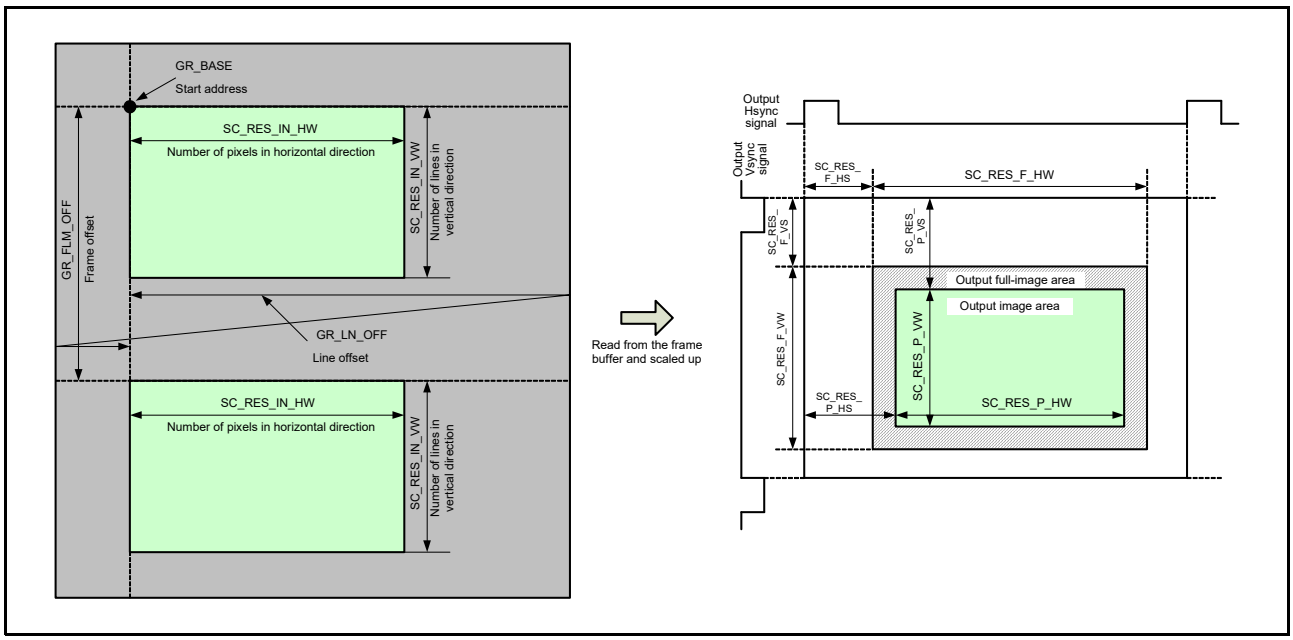


Figure 36.16 Area Setting for Input Video Image Signal Display and Enlarged Graphics Display

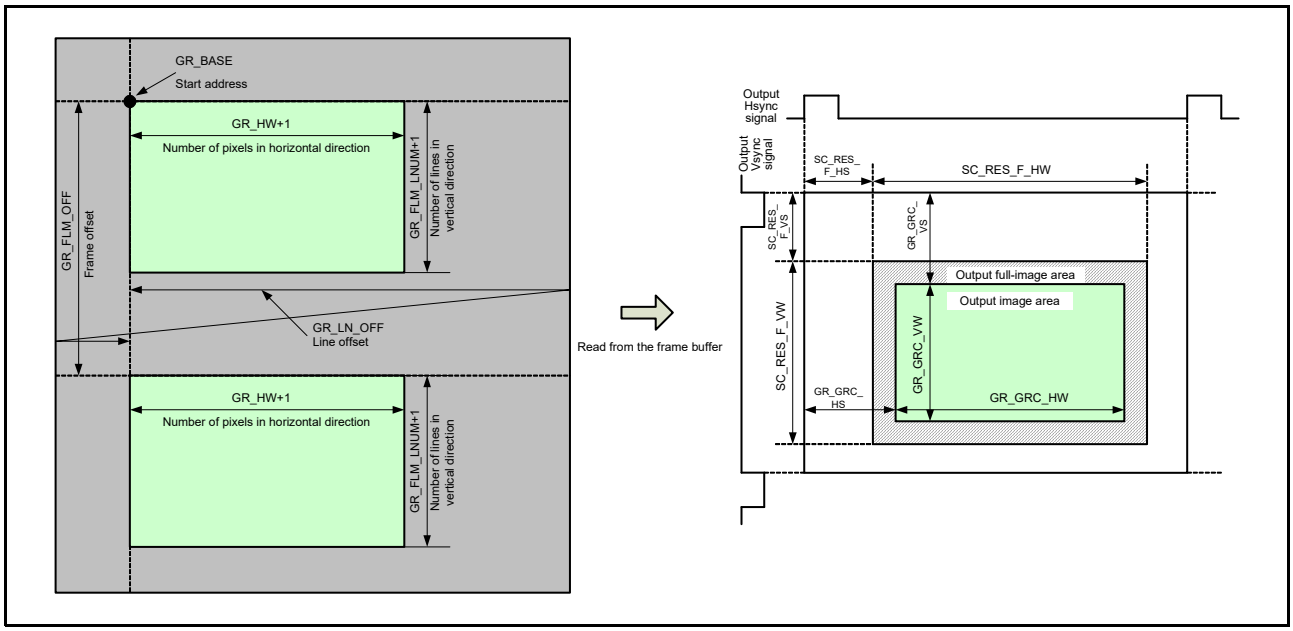


Figure 36.17 Area Setting for Graphics Display

Table 36.36 Scaling-Up Process or Graphics 0 Process Selection

Register Name	Bit Name	Initial Value	Description
SC_SCL0_US8	SC_RES_IBUS_SYNC_SEL	0	Sync Signal Select for Frame Buffer Read Block 0: Sync signals from the scaling-up control block 1: Sync signals from the graphics processing block

The GR_DISP_SEL bits are used to select a display by the scaling-up control block (video image display or enlarged graphics display) or graphics display.

For details on the graphics processing, refer to section 38, Video Display Controller 6 (5): Image Synthesizer.

36.1.19 Selecting Field for Frame Buffer Reading

For the next frame buffer to be read, the top or bottom field can be selected. This field selection is used in the scaling-up control block.

Table 36.37 Field Specification for Frame Buffer Reading

Register Name	Bit Name	Initial Value	Description
GR_FLM1	GR_FLD_SEL	0	Enables or disables top or bottom field selection for the next frame buffer to be read. 0: Field selection is disabled. 1: Top or bottom field can be selected.
GR_FLM3	GR_FLD_NXT	0	Selects the top or bottom field for the next frame buffer. 0: Bottom 1: Top

36.1.20 Pointer Buffer and Frame Buffer Reading Processing

(1) Pointer Buffers

The pointer buffers can be used to control the frame buffer for the input video image. They are mainly used to prevent flicker in the output video image, which occurs when the input and output vertical sync signals are asynchronous.

Four pointer buffers are provided and each pointer buffer has a start address register that shows the start location of the frame buffer and a field information register that shows the current field is the top or bottom field. The four pointer buffers are arranged in a ring structure and a write pointer is provided to indicate the pointer buffer corresponding to the location currently being written to. The location pointed to by the write pointer is being written to, and the corresponding pointer buffer value is undetermined. The value in the pointer buffer corresponding to the location being written to and the value in the write pointer are automatically updated when frame data writing is completed.

When the frame buffer address setting signal is linked with the pointer buffer (GR_FLM_SEL = 3), frame buffer reading can be controlled by using the read pointer that indicates the pointer buffer corresponding to the location being read; the start address and field information of the next frame buffer to be read should be read from the pointer buffer and they should be set in the frame buffer base address and field information. The read pointer value is automatically updated at the rising edge of the vertical sync signal on the reading side.

(2) Write Pointer Control

The write pointer is incremented by one every time frame data writing is completed.

(3) Read Pointer Control

The read pointer is updated at the rising edge of the vertical sync signal on the reading side according to the difference between the read and write pointer values as follows.

(A) When (write pointer value) – (read pointer value) ≤ 1

The read pointer value is not updated (the same frame is displayed continuously).

(B) When (write pointer value) – (read pointer value) = 2

The read pointer is incremented by one with the next updating timing.

(C) When (write pointer value) – (read pointer value) ≥ 3

The read pointer is incremented by two with the next updating timing.

(One frame is skipped.)

(4) Frame Buffer Read Control

(A) When SC_RES_WENB = 0

Frame data is not written to the frame buffer, and the frame buffer is not read.

(B) When frame data writing is terminated with SC_RES_WENB = 1

As the pointer buffer value is determined, the frame buffer is read.

Table 36.38 Frame Buffer Control

Register Name	Bit Name	Initial Value	Description
SC_SCL1_PBUF0	SC_BUF0_ADD	0	Start address of the write buffer pointed to by pointer buffer 0
SC_SCL1_PBUF1	SC_BUF1_ADD	0	Start address of the write buffer pointed to by pointer buffer 1
SC_SCL1_PBUF2	SC_BUF2_ADD	0	Start address of the write buffer pointed to by pointer buffer 2
SC_SCL1_PBUF3	SC_BUF3_ADD	0	Start address of the write buffer pointed to by pointer buffer 3
SC_SCL1_PBUF_FLD	SC_FLD_INF0	0	Top or bottom field information pointed to by pointer buffer 0 0: Bottom 1: Top
SC_SCL1_PBUF_FLD	SC_FLD_INF1	0	Top or bottom field information pointed to by pointer buffer 1 0: Bottom 1: Top
SC_SCL1_PBUF_FLD	SC_FLD_INF2	0	Top or bottom field information pointed to by pointer buffer 2 0: Bottom 1: Top
SC_SCL1_PBUF_FLD	SC_FLD_INF3	0	Top or bottom field information pointed to by pointer buffer 3 0: Bottom 1: Top
SC_SCL1_PBUF_CNT	SC_PBUF_RST	0	Reset Control for the Pointer Buffer 0: Pointer buffer is not reset. 1: Pointer buffer is reset.
SC_SCL1_MON1	SC_PBUF_NUM	0	Write pointer indicating the pointer buffer number corresponding to the location currently being written to.
SC_SCL1_WR5	SC_RES_WENB	0	Frame Buffer Write Enable After making the setting to enable writing, writing starts from the second frame. 0: Writing is disabled. 1: Writing is enabled.
GR_FLM1	GR_FLM_SEL	0	Frame Buffer Address Setting Signal Selection 0: Links to scaling-down process. (This setting is prohibited when separate write addresses are specified for the top and bottom fields; that is, SC_RES_TB_ADD_MOD = 1 in SC_SCL1_WR1.) 1: Selects GR0_FLM_NUM. 2: Links to distortion correction. 3: Links to pointer buffer.
GR_FLM2	GR_BASE	0	Frame Buffer Base Address Sets the start address of the frame buffer where frame data is to be stored. GR_BASE[4:3] and GR_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower three bits should be fixed to 000.

For other frame buffer read operation and graphics processing, refer to section 38, Video Display Controller 6 (5): Image Synthesizer.

36.2 Register Descriptions

Table 36.39 shows the register configuration.

- Symbols used in Register Description:

Initial value: Register value after a reset
—: Undefined value
R/W: Readable/writable. The written value can be read.
R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.
R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.
R: Read-only. The write value should always be 0.
—/W: Write-only. The read value is undefined.

Table 36.39 Register Configuration of the Scaler

Name	Abbreviation	R/W	Initial Value	Address	Access Size
SCL0 register update control register (SC0)	SC0_SCL0_UPDATE	R/WC1	H'0000 0000	H'FCFF 7500	32
Mask control register (SC0)	SC0_SCL0_FRC1	R/W	H'0AF0 0001	H'FCFF 7504	32
Missing Vsync compensation control register (SC0)	SC0_SCL0_FRC2	R/W	H'0E10 0001	H'FCFF 7508	32
Output sync select register (SC0)	SC0_SCL0_FRC3	R/W	H'0000 0001	H'FCFF 750C	32
Free-running period control register (SC0)	SC0_SCL0_FRC4	R/W	H'020C 031F	H'FCFF 7510	32
Output delay control register (SC0)	SC0_SCL0_FRC5	R/W	H'0000 0101	H'FCFF 7514	32
Full-screen vertical size register (SC0)	SC0_SCL0_FRC6	R/W	H'0023 01E0	H'FCFF 7518	32
Full-screen horizontal size register (SC0)	SC0_SCL0_FRC7	R/W	H'0090 0280	H'FCFF 751C	32
Vsync detection register (SC0)	SC0_SCL0_FRC9	R	H'0000 0000	H'FCFF 7524	32
Status monitor 0 register (SC0)	SC0_SCL0_MON0	R	H'0000	H'FCFF 7528	16
Interrupt control register (SC0)	SC0_SCL0_INT	R/W	H'0000	H'FCFF 752A	16
Scaling-down control register (SC0)	SC0_SCL0_DS1	R/W	H'0000 0011	H'FCFF 752C	32
Vertical capture size register (SC0)	SC0_SCL0_DS2	R/W	H'0012 00F0	H'FCFF 7530	32
Horizontal capture size register (SC0)	SC0_SCL0_DS3	R/W	H'00F4 05A0	H'FCFF 7534	32
Horizontal scale down register (SC0)	SC0_SCL0_DS4	R/W	H'1000 2408	H'FCFF 7538	32
Initial vertical phase register (SC0)	SC0_SCL0_DS5	R/W	H'1800 0000	H'FCFF 753C	32
Vertical scaling register (SC0)	SC0_SCL0_DS6	R/W	H'0000 07FC	H'FCFF 7540	32
Scaling-down control block output size register (SC0)	SC0_SCL0_DS7	R/W	H'00F0 0280	H'FCFF 7544	32
Scaling-up control register (SC0)	SC0_SCL0_US1	R/W	H'0000 0011	H'FCFF 7548	32
Output image vertical size register (SC0)	SC0_SCL0_US2	R/W	H'0023 01E0	H'FCFF 754C	32
Output image horizontal size register (SC0)	SC0_SCL0_US3	R/W	H'0090 0280	H'FCFF 7550	32
Scaling-up control block input size register (SC0)	SC0_SCL0_US4	R/W	H'00F0 0280	H'FCFF 7554	32
Horizontal scale up register (SC0)	SC0_SCL0_US5	R/W	H'0000 2408	H'FCFF 7558	32
Horizontal scale up initial phase register (SC0)	SC0_SCL0_US6	R/W	H'1000 0000	H'FCFF 755C	32
Trimming register (SC0)	SC0_SCL0_US7	R/W	H'0000 0000	H'FCFF 7560	32
Frame buffer read select register (SC0)	SC0_SCL0_US8	R/W	H'0000 0000	H'FCFF 7564	32
Background color register (SC0)	SC0_SCL0_OVR1	R/W	H'0080 0080	H'FCFF 756C	32

Table 36.39 Register Configuration of the Scaler

Name	Abbreviation	R/W	Initial Value	Address	Access Size
SCL1 register update control register (SC0)	SC0_SCL1_UPDATE	R/WC1	H'0000 0000	H'FCFF 7580	32
Writing mode register (SC0)	SC0_SCL1_WR1	R/W	H'0000 0000	H'FCFF 7588	32
Write address register 1T (SC0)	SC0_SCL1_WR2	R/W	H'0000 0000	H'FCFF 758C	32
Write address register 2T (SC0)	SC0_SCL1_WR3	R/W	H'0800 0001	H'FCFF 7590	32
Write address register 3T (SC0)	SC0_SCL1_WR4	R/W	H'0008 0000	H'FCFF 7594	32
Frame sub-sampling register (SC0)	SC0_SCL1_WR5	R/W	H'0000 1000	H'FCFF 759C	32
Bit reduction register (SC0)	SC0_SCL1_WR6	R/W	H'0000 0000	H'FCFF 75A0	32
Write detection register (SC0)	SC0_SCL1_WR7	R	H'0000 0000	H'FCFF 75A4	32
Write address register 1B (SC0)	SC0_SCL1_WR8	R/W	H'0000 0000	H'FCFF 75A8	32
Write address register 2B (SC0)	SC0_SCL1_WR9	R/W	H'0800 0001	H'FCFF 75AC	32
Write address register 3B (SC0)	SC0_SCL1_WR10	R/W	H'0008 0000	H'FCFF 75B0	32
Write detection register B (SC0)	SC0_SCL1_WR11	R	H'0000 0000	H'FCFF 75B4	32
Status monitor 1 register (SC0)	SC0_SCL1_MON1	R	H'0000 0000	H'FCFF 75B8	32
Pointer buffer 0 register (SC0)	SC0_SCL1_PBUF0	R	H'0000 0000	H'FCFF 75BC	32
Pointer buffer 1 register (SC0)	SC0_SCL1_PBUF1	R	H'0000 0000	H'FCFF 75C0	32
Pointer buffer 2 register (SC0)	SC0_SCL1_PBUF2	R	H'0000 0000	H'FCFF 75C4	32
Pointer buffer 3 register (SC0)	SC0_SCL1_PBUF3	R	H'0000 0000	H'FCFF 75C8	32
Pointer buffer and field information register (SC0)	SC0_SCL1_PBUF_FLD	R	H'0000 0000	H'FCFF 75CC	32
Pointer buffer control register (SC0)	SC0_SCL1_PBUF_CNT	R/W	H'0000 0000	H'FCFF 75D0	32
Graphics 0 register update control register	GR0_UPDATE	R/WC1	H'0000 0000	H'FCFF 7600	32
Frame buffer read control register (graphics 0)	GR0_FLM_RD	R/W	H'0000 0000	H'FCFF 7604	32
Frame buffer control register 1 (graphics 0)	GR0_FLM1	R/W	H'0000 0000	H'FCFF 7608	32
Frame buffer control register 2 (graphics 0)	GR0_FLM2	R/W	H'0000 0000	H'FCFF 760C	32
Frame buffer control register 3 (graphics 0)	GR0_FLM3	R/W	H'0800 0001	H'FCFF 7610	32
Frame buffer control register 4 (graphics 0)	GR0_FLM4	R/W	H'0008 0000	H'FCFF 7614	32
Frame buffer control register 5 (graphics 0)	GR0_FLM5	R/W	H'0000 03FF	H'FCFF 7618	32
Frame buffer control register 6 (graphics 0)	GR0_FLM6	R/W	H'8000 0000	H'FCFF 761C	32
Alpha blending control register 1 (graphics 0)	GR0_AB1	R/W	H'0000 0000	H'FCFF 7620	32
Alpha blending control register 2 (graphics 0)	GR0_AB2	R/W	H'0000 0000	H'FCFF 7624	32
Alpha blending control register 3 (graphics 0)	GR0_AB3	R/W	H'0000 0000	H'FCFF 7628	32
Alpha blending control register 7 (graphics 0)	GR0_AB7	R/W	H'00FF 0000	H'FCFF 7638	32
Alpha blending control register 8 (graphics 0)	GR0_AB8	R/W	H'0000 0000	H'FCFF 763C	32
Alpha blending control register 9 (graphics 0)	GR0_AB9	R/W	H'0000 0000	H'FCFF 7640	32

Table 36.39 Register Configuration of the Scaler

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Alpha blending control register 10 (graphics 0)	GR0_AB10	R/W	H'0000 0000	H'FCFF 7644	32
Alpha blending control register 11 (graphics 0)	GR0_AB11	R/W	H'0000 0000	H'FCFF 7648	32
Background color control register (graphics 0)	GR0_BASE	R/W	H'0000 8080	H'FCFF 764C	32
CLUT table control register (graphics 0)	GR0_CLUT	R/W	H'0000 0000	H'FCFF 7650	32

36.2.1 SCL0 Register Update Control Register (SC0_SCL0_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SC0_SCL0_VEN_D	SC0_SCL0_VEN_C	—	—	—	SC0_SCL0_UPDATE	—	—	—	SC0_SCL0_VEN_B	—	—	—	SC0_SCL0_VEN_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/WC1	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	SC0_SCL0_VEN_D	0	R/WC1	Scaling-Up Control and Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
12	SC0_SCL0_VEN_C	0	R/WC1	Scaling-Down Control and Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SC0_SCL0_UPDATE	0	R/WC1	SYNC Control Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_SCL0_VEN_B	0	R/WC1	Synchronization Control and Scaling-up Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_SCL0_VEN_A	0	R/WC1	Scaling-Down Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

36.2.2 Mask Control Register (SC0_SCL0_FRC1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_RES_VMASK[15:0]															
Initial value:	0	0	0	0	1	0	1	0	1	1	1	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_VMASK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	SC0_RES_VMASK [15:0]	2800	R/W	Repeated Vsync Signal Masking Period Sets the repeated Vsync signal masking period beginning at a Vsync signal in terms of 128 pixel-clock periods. Masking period [usec] = SC0_RES_VMASK × 128 ÷ pixel clock frequency [MHz]
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_ VMASK_ON	1	R/W	Repeated Vsync Signal Masking Control 0: Repeated Vsync signal masking control is disabled. 1: Repeated Vsync signal masking control is enabled.

Note: This register is updated when the SC0_SCL0_UPDATE bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.3 Missing Vsync Compensation Control Register (SC0_SCL0_FRC2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_RES_VLACK[15:0]															
Initial value:	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_VLACK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	SC0_RES_ VLACK [15:0]	3600	R/W	Missing-Sync Compensating Pulse Output Wait Time Sets the wait time before outputting a missing-sync compensating pulse after a Vsync signal. Wait time [usec] = SC0_RES_VLACK × 128 ÷ pixel clock frequency [MHz]
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_ VLACK_ON	1	R/W	Missing Vsync Signal Compensation 0: Compensation of missing Vsync signals is disabled. 1: Compensation of missing Vsync signals is enabled.

Note: This register is updated when the SC0_SCL0_UPDATE bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.4 Output Sync Select Register (SC0_SCL0_FRC3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_VS_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_VS_SEL	1	R/W	Vsync Signal Output Select 0: Externally input Vsync signal 1: Internally generated free-running Vsync signal

Note: This register is updated when the SC0_SCL0_UPDATE bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.5 Free-Running Period Control Register (SC0_SCL0_FRC4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_FV[10:0]										
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_FH[10:0]-										
Initial value:	0	0	0	0	0	0	1	1	0	0	0	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_FV [10:0]	524	R/W	Free-Running Vsync Period Setting Free-running Vsync period = (SC0_RES_FV + 1) × horizontal period [usec]
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_FH [10:0]	799	R/W	Hsync Period Setting Hsync period [usec] = (SC0_RES_FH +1) ÷ pixel clock frequency [MHz]

Note: This register is updated when the SC0_SCL0_UPDATE bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.6 Output Delay Control Register (SC0_SCL0_FRC5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SC0_RES_FLD_DLY_SEL	SC0_RES_VSDLY[7:0]							
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SC0_RES_FLD_DLY_SEL	1	R/W	Field Determination Signal Delay Control 0: No delay 1: Delay of one vertical cycle
7 to 0	SC0_RES_VSDLY[7:0]	1	R/W	Vsync Signal Delay Control Adjusts the Vsync signal delay in the output Hsync period units. Vsync signal delay [usec]: SC0_RES_VSDLY × output Hsync period [usec]

Note: This register is updated when the SC0_SCL0_VEN_B bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.7 Full-Screen Vertical Size Register (SC0_SCL0_FRC6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_F_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_F_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_F_VS[10:0]	35	R/W	Vertical Enable Signal Start Position for Full Screen. (VSYNC + V backporch lines) Note: The set value should be four or more (lines). SC0_RES_F_VS + SC0_RES_F_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_F_VW[10:0]	480	R/W	Vertical Enable Signal Width for Full Screen (lines) Note: SC0_RES_F_VS + SC0_RES_F_VW should be equal to or less than 2039 (lines).

Note: This register is updated when the SC0_SCL0_VEN_B bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.8 Full-Screen Horizontal Size Register (SC0_SCL0_FRC7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_F_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_F_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_F_HS[10:0]	144	R/W	Horizontal Enable Signal Start Position for Full Screen. (HSYNC+H backporch pixel-clock cycles) Note: The set value should be 16 or more (clock cycles). SC0_RES_F_HS + SC0_RES_F_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_F_HW[10:0]	640	R/W	Horizontal Enable Signal Width for Full Screen (pixel-clock cycles) Note 1. SC0_RES_F_HS + SC0_RES_F_HW should be equal to or less than 2015 (clock cycles). Note 2. The set value should be equal to (horizontal signal width for full screen + 2) when serial RGB output is selected as an LCD output signal.

Note: This register is updated when the SC0_SCL0_VEN_B bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.9 Vsync Detection Register (SC0_SCL0_FRC9)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_QVLOCK	—	—	—	SC0_RES_QVLACK
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_RES_QVLOCK	0	R	Locked Vsync Signal Detection Flag 1: No repeated or missing Vsync signal input has been detected for four or more vertical periods. 0: Repeated or missing Vsync signal input has been detected.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_QVLACK	0	R	Missing Vsync Signal Detection Flag 1: Missing Vsync signal input has been detected. 0: No missing Vsync signal input has been detected.

36.2.10 Status Monitor 0 Register (SC0_SCL0_MON0)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_LIN_STAT[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_LIN_STAT[10:0]	All 0	R	Current location of the image line input to the scaling-down control block.

36.2.11 Interrupt Control Register (SC0_SCL0_INT)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_LINE[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_LINE[10:0]	All 0	R/W	Setting of Interrupt on Image Line Input to Scaling-down Control Block When the location of the image line input to the scaling-down control block matches the SC0_RES_LINE setting, an interrupt signal is output. However, setting this is prohibited in this product.

Note: This register is updated when the SC0_SCL0_VEN_A bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.12 Scaling-Down Control Register (SC0_SCL0_DS1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_DS_V_ON	—	—	—	SC0_RES_DS_H_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_RES_DS_V_ON	1	R/W	Vertical Scale Down On/Off 0: Off 1: On
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_DS_H_ON	1	R/W	Horizontal Scale Down On/Off 0: Off 1: On

Note: This register is updated when the SC0_SCL0_VEN_A bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.13 Vertical Capture Size Register (SC0_SCL0_DS2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_VS [10:0]	18	R/W	Vertical Position Setting for Video Signal Capturing (VSYNC + (V backporch - 1) lines) Note: The set value should be four or more (lines). SC0_RES_VS + SC0_RES_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_VW [10:0]	240	R/W	Vertical Width of Video Signal to be Captured (Lines) Note: SC0_RES_VS + SC0_RES_VW should be equal to or less than 2039 (lines).

Note: This register is updated when the SC0_SCL0_VEN_A bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.14 Horizontal Capture Size Register (SC0_SCL0_DS3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SC0_RES_HW[11:0]											
Initial value:	0	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_HS [10:0]	244	R/W	Horizontal Position Setting for Video Signal Capturing (HSYNC + H backporch video-image clock cycles) Note: The set value should be 16 or more (clock cycles). SC0_RES_HS + SC0_RES_HW should be equal to or less than 4063 (clock cycles).
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	SC0_RES_HW [11:0]	1440	R/W	Horizontal Width of Video Signal to be Captured (Video-image clock cycles) Note: SC0_RES_HS + SC0_RES_HW should be equal to or less than 4063 (clock cycles).

Note: This register is updated when the SC0_SCL0_VEN_A bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.15 Horizontal Scale Down Register (SC0_SCL0_DS4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	SC0_RES_PFIL_SEL	SC0_RES_DS_H_INTERPOTYP	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_DS_H_RATIO[15:0]															
Initial value:	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29	SC0_RES_PFIL_SEL	0	R/W	Prefilter Mode Select for Brightness Signals 0: The prefilter is turned off. 1: The prefilter is turned on. (1/4 + 1/2 + 1/4)
28	SC0_RES_DS_H_INTERPOTYP	1	R/W	Horizontal Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
27 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	SC0_RES_DS_H_RATIO[15:0]	9224	R/W	Horizontal Scale Down Ratio ([15:12]: Integer part, [11:0]: Decimal part) round(SC0_RES_HW ÷ SC0_RES_OUT_HW × 4096) SC0_RES_DS_H_RATIO < 4096: Setting prohibited SC0_RES_DS_H_RATIO = 4096: 100% scale up SC0_RES_DS_H_RATIO > 4096: Scale down

Note: This register is updated when the SC0_SCL0_VEN_A bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.16 Initial Vertical Phase Register (SC0_SCL0_DS5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SC0_RES_V_INTERPOTYP	SC0_RES_TOP_INIPHASE[11:0]											
Initial value:	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SC0_RES_BTM_INIPHASE[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	SC0_RES_V_INTERPOTYP	1	R/W	Vertical Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
27 to 16	SC0_RES_TOP_INIPHASE[11:0]	2048	R/W	Vertical Interpolation Start Phase for Top Field 0 to 4095 (0 to approx. 1.0)
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	SC0_RES_BTM_INIPHASE[11:0]	0	R/W	Vertical Interpolation Start Phase for Bottom Field 0 to 4095 (0 to approx. 1.0)

Note: This register is updated when the SC0_SCL0_VEN_A and SC0_SCL0_VEN_B bits in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) are 1.

36.2.17 Vertical Scaling Register (SC0_SCL0_DS6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_V_RATIO[15:0]															
Initial value:	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	SC0_RES_V_RATIO [15:0]	2044	R/W	Vertical Scale Up/Down Ratio ([15:12]: Integer part, [11:0]: Decimal part) For scale down: round (SC0_RES_VW ÷ SC0_RES_OUT_VW × 4096) For scale up: round (SC0_RES_IN_VW ÷ SC0_RES_P_VW × 4096) SC0_RES_V_RATIO < 4096: Scale up SC0_RES_V_RATIO = 4096: 100% scale up SC0_RES_V_RATIO > 4096: Scale down

Note: These bits updated when the SC0_SCL0_VEN_A and SC0_SCL0_VEN_B bits in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) are 1. Accordingly, even a scaled-up graphics display requires both an input Vsync signal and output Vsync signal.

36.2.18 Scaling-Down Control Block Output Size Register (SC0_SCL0_DS7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_OUT_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_OUT_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_OUT_VW [10:0]	240	R/W	Number of Valid Lines in Vertical Direction Output by Scaling-down Control Block (lines) This bit setting is used for the number of lines to be written to the frame buffer. When SC0_SCL1_WR1.SC0_RES_LOOP is 0 (frame write mode), specify the number of lines for one frame. When SC0_SCL1_WR1.SC0_RES_LOOP is 1 (line write mode), specify the number of lines for repeated write. Note: The SC0_RES_OUT_VW value should be aligned in 4-line units and equal to or smaller than the SC0_RES_VW value.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_OUT_HW [10:0]	640	R/W	Number of Valid Horizontal Pixels Output by Scaling-Down Control Block (video-image clock cycles) Note: The SC0_RES_OUT_HW value should be aligned in 4-pixel units and equal to or smaller than the SC0_RES_HW value.

Note: This register is updated when the SC0_SCL0_VEN_A and SC0_SCL0_VEN_C bits in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) are 1.

36.2.19 Scaling-Up Control Register (SC0_SCL0_US1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_US_V_ON	—	—	—	SC0_RES_US_H_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_RES_US_V_ON	1	R/W	Vertical Scale Up On/Off 0: Off 1: On
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_US_H_ON	1	R/W	Horizontal Scale Up On/Off 0: Off 1: On

Note: This register is updated when the SC0_SCL0_VEN_B bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.20 Output Image Vertical Size Register (SC0_SCL0_US2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_P_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_P_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_P_VS[10:0]	35	R/W	Vertical Enable Signal Start Position for Output Image (VSYNC + V backporch lines) Note: The set value should be four or more (lines). SC0_RES_P_VS + SC0_RES_P_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_P_VW[10:0]	480	R/W	Vertical Enable Signal Width for Output Image (lines) Note: SC0_RES_P_VS + SC0_RES_P_VW should be equal to or less than 2039 (lines).

Note: This register is updated when the SC0_SCL0_VEN_B bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.21 Output Image Horizontal Size Register (SC0_SCL0_US3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_P_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_P_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_P_HS[10:0]	144	R/W	Horizontal Enable Signal Start Position for Output Image (HSYNC+H backporch pixel-clock cycles) Note: The set value should be 16 or more (clock cycles). SC0_RES_P_HS + SC0_RES_P_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_P_HW[10:0]	640	R/W	Horizontal Enable Signal Width for Output Image (pixel-clock cycles) Note: SC0_RES_P_HS + SC0_RES_P_HW should be equal to or less than 2015 (clock cycles).

Note: This register is updated when the SC0_SCL0_VEN_B bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.22 Scaling-Up Control Block Input Size Register (SC0_SCL0_US4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_IN_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_IN_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_IN_VW[10:0]	240	R/W	Number of Valid Lines in Vertical Direction Input to Scaling-up Control Block (lines)
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_IN_HW[10:0]	640	R/W	Number of Valid Horizontal Pixels Input to Scaling-up Control Block (pixel-clock cycles)

Note: This register is updated when the SC0_SCL0_VEN_B and SC0_SCL0_VEN_D bits in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) are 1.

36.2.23 Horizontal Scale Up Register (SC0_SCL0_US5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_US_H_RATIO[15:0]															
Initial value:	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	SC0_RES_US_H_RATIO [15:0]	9224	R/W	Horizontal Scale Up Ratio ([15:12]: Integer part, [11:0]: Decimal part) round (SC0_RES_IN_HW ÷ SC0_RES_P_HW × 4096) SC0_RES_US_H_RATIO < 4096: Scale up SC0_RES_US_H_RATIO = 4096: 100% scale up SC0_RES_US_H_RATIO > 4096: Setting prohibited

Note: This register is updated when the SC0_SCL0_VEN_B bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.24 Horizontal Scale Up Initial Phase Register (SC0_SCL0_US6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SC0_RES_US_H_INTERPOTYP	SC0_RES_US_HT_INIPHASE[11:0]											
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SC0_RES_US_HB_INIPHASE[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	SC0_RES_US_H_INTERPOTYP	1	R/W	Horizontal Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
27 to 16	SC0_RES_US_HT_INIPHASE[11:0]	0	R/W	Horizontal Interpolation Start Phase for Top Field 0 to 4095 (0 to approx. 1.0)
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	SC0_RES_US_HB_INIPHASE[11:0]	0	R/W	Horizontal Interpolation Start Phase for Bottom Field 0 to 4095 (0 to approx. 1.0)

Note: This register is updated when the SC0_SCL0_VEN_B bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.25 Trimming Register (SC0_SCL0_US7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_HCUT[7:0]								SC0_RES_VCUT[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	SC0_RES_HCUT[7:0]	0	R/W	Horizontal Amount of Cut-off Post-Scaling Image (Right and Left Parts) Sets the number of pixel-clock cycles.
7 to 0	SC0_RES_VCUT[7:0]	0	R/W	Vertical Amount of Cut-off Post-Scaling Image (Upper and Lower Parts) Sets the number of lines.

Note: This register is updated when the SC0_SCL0_VEN_B bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.26 Frame Buffer Read Select Register (SC0_SCL0_US8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_IBUS_SYNC_SEL	—	—	—	SC0_RES_DISP_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_RES_IBUS_SYNC_SEL	0	R/W	Sync Signal Select for Frame Buffer Read Block 0: Sync signals from the scaling-up control block 1: Sync signals from the graphics processing block
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_DISP_ON	0	R/W	Post-Scaling Image Frame Display On/Off 0: Frame display on 1: Frame display off

Note: SC0_RES_IBUS_SYNC_SEL is updated when the SC0_SCL0_VEN_B and SC0_SCL0_VEN_D bits in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) are 1.
SC0_RES_DISP_ON is updated when the SC0_SCL0_VEN_B bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.27 Background Color Register (SC0_SCL0_OVR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	SC0_RES_BK_COL_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_BK_COL_G[7:0]								SC0_RES_BK_COL_B[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	SC0_RES_BK_CLO_R[7:0]	128	R/W	Background Color Setting R/Cr Signal R:8 bits; unsigned (0 to 255 [LSB]) Cr:8 bits; 128 offset binary; unsigned (0 to 255 [LSB])
15 to 8	SC0_RES_BK_COL_G[7:0]	0	R/W	Background Color Setting G/Y Signal G/Y: 8 bits; unsigned (0 to 255 [LSB])
7 to 0	SC0_RES_BK_COL_B[7:0]	128	R/W	Background Color Setting B/Cb Signal B:8 bits; unsigned (0 to 255 [LSB]) Cb:8 bits; 128 offset binary; unsigned (0 to 255 [LSB])

Note: This register is updated when the SC0_SCL0_VEN_B bit in the SC0_SCL0 register update control register (SC0_SCL0_UPDATE) is 1.

36.2.28 SCL1 Register Update Control Register (SC0_SCL1_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	SC0_SCL1_UPDATE_B	—	—	—	SC0_SCL1_UPDATE_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC0_SCL1_VEN_B	—	—	—	SC0_SCL1_VEN_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	SC0_SCL1_UPDATE_B	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SC0_SCL1_UPDATE_A	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated.
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_SCL1_VEN_B	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_SCL1_VEN_A	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

36.2.29 Writing Mode Register (SC0_SCL1_WR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_WRSWA [2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SC0_RES_TB_ADD_MOD	SC0_RES_DS_WR_MD [2:0]			SC0_RES_MD [1:0]		SC0_RES_LOOP	SC0_RES_BST_MD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	SC0_RES_WRSWA[2:0]	All 0	R/W	8-Bit, 16-Bit, or 32-Bit Swap Setting These bits control swapping in frame buffer writing as follows. Bit 0 0: Swapped in 8-bit units. 1: Not swapped in 8-bit units. Bit 1 0: Swapped in 16-bit units. 1: Not swapped in 16-bit units. Bit 2 0: Swapped in 32-bit units. 1: Not swapped in 32-bit units. According to the setting of these bits, data is swapped as follows. Each number in parentheses ((1) to (8)) indicates 8-bit data. 000: (1) (2) (3) (4) (5) (6) (7) (8) [Not swapped] 001: (2) (1) (4) (3) (6) (5) (8) (7) [Swapped in 8-bit units] 010: (3) (4) (1) (2) (7) (8) (5) (6) [Swapped in 16-bit units] 011: (4) (3) (2) (1) (8) (7) (6) (5) [Swapped in 16-bit units + 8-bit units] 100: (5) (6) (7) (8) (1) (2) (3) (4) [Swapped in 32-bit units] 101: (6) (5) (8) (7) (2) (1) (4) (3) [Swapped in 32-bit units + 8-bit units] 110: (7) (8) (5) (6) (3) (4) (1) (2) [Swapped in 32-bit units + 16-bit units] 111: (8) (7) (6) (5) (4) (3) (2) (1) [Swapped in 32-bit units + 16-bit units + 8-bit units] Note: When YCbCr422 or RGB565 is selected as a frame buffer video-signal writing format, these bits should be set to 000 [Not swapped].
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	SC0_RES_TB_ADD_MOD	0	R/W	Top and Bottom Data Write Address Specification Method 0: A write address is specified in common for top and bottom data. 1: Separate write addresses are specified for top and bottom data.
6 to 4	SC0_RES_DS_WR_MD [2:0]	0	R/W	Frame Buffer Writing Mode for Image Processing 0: Normal 1: Horizontal mirroring 2: 90° rotation 3: 180° rotation 4: 270° rotation 5 to 7: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
3, 2	SC0_RES_MD [1:0]	0	R/W	Frame Buffer Video-Signal Writing Format 0: YCbCr422 (16 bits) 1: RGB565 (16 bits) 2: RGB888 (24 (32) bits) 3: YCbCr444 (24 (32) bits)
1	SC0_RES_ LOOP	0	R/W	Frame Buffer Write Mode Select 0: Frame mode 1: Line mode (read as ring buffer)
0	SC0_RES_ BST_MD	0	R/W	Transfer Burst Length for Frame Buffer Writing 0: 32-byte 1: 128-byte

Note: SC0_RES_LOOP and SC0_RES_BST_MD are updated when the SC0_SCL1_VEN_B bit in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) is 1.
SC0_RES_TB_ADD_MOD, SC0_RES_DS_WR_MD, and SC0_RES_MD are updated when the SC0_SCL1_VEN_A and SC0_SCL1_VEN_B bits in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) are 1.
SC0_RES_WRSWA is updated when the SC0_SCL1_UPDATE_A bit in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) is 1.

36.2.30 Write Address Register 1T (SC0_SCL1_WR2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_RES_BASE[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_BASE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC0_RES_ BASE [31:0]	0	R/W	Frame Buffer Base Address Sets the start address of the frame buffer to store the frame data for the top field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 0. For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.

Note: This register is updated when the SC0_SCL1_VEN_B bit in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) is 1.

36.2.31 Write Address Register 2T (SC0_SCL1_WR3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	SC0_RES_LN_OFF[14:0]														
Initial value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SC0_RES_FLM_NUM[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	SC0_RES_LN_OFF [14:0]	2048	R/W	Frame Buffer Line Offset Address Sets the line offset address for calculating the line start address for the top field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 0. Line 0: SC0_RES_BASE Line 1: SC0_RES_BASE + SC0_RES_LN_OFF × 1 : Line n: SC0_RES_BASE + SC0_RES_LN_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SC0_RES_FLM_NUM [9:0]	1	R/W	Number of Frames of Buffer to be Written to Sets the number of frames for the top field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 0 Number of frames defined by SC0_RES_FLM_NUM + 1 are used.

Note: This register is updated when the SC0_SCL1_VEN_B bit in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) is 1.

36.2.32 Write Address Register 3T (SC0_SCL1_WR4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	SC0_RES_FLM_OFF[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_FLM_OFF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	SC0_RES_FLM_OFF [22:0]	524288	R/W	Frame Buffer Frame Offset Address Sets the frame offset address for calculating the start address of each frame for the top field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 0. Buffer 0: SC0_RES_BASE Buffer 1: SC0_RES_BASE + SC0_RES_FLM_OFF × 1 : Buffer n: SC0_RES_BASE + SC0_RES_FLM_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.

Note: This register is updated when the SC0_SCL1_VEN_B bit in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) is 1.

36.2.33 Frame Sub-Sampling Register (SC0_SCL1_WR5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SC0_RES_INTER	—	—	SC0_RES_FS_RATE[1:0]		—	—	—	SC0_RES_FLD_SEL	—	—	—	SC0_RES_WENB
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	SC0_RES_INTER	1	R/W	Field Operating Mode Select 0: Progressive 1: Interlace
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	SC0_RES_FS_RATE[1:0]	0	R/W	Writing Rate Sets the frame buffer writing rate to the vertical frequency of the input signal. 0: 1/1 an input signal (The SC0_RES_FLD_SEL setting is invalid.) 1: 1/2 an input signal 2: 1/4 an input signal 3: 1/8 an input signal
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_RES_FLD_SEL	0	R/W	Write Field Select 0: Top field 1: Bottom field
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_WENB	0	R/W	Frame Buffer Write Enable After making the setting to enable writing, writing starts from the second frame. 0: Frame buffer writing is disabled. 1: Frame buffer writing is enabled.

Note: SC0_RES_INTER, SC0_RES_FS_RATE[1:0], and SC0_RES_FLD_SEL are updated when the SC0_SCL1_VEN_A bit in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) is 1. SC0_RES_WENB is updated when the SC0_SCL1_VEN_A and SC0_SCL1_VEN_B bits in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) are 1.

36.2.34 Bit Reduction Register (SC0_SCL1_WR6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_DTH_ON	—	—	—	SC0_RES_BITDEC_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_RES_DTH_ON	0	R/W	Dither Correction On/Off 0: Off (rounded off) 1: On (2 × 2 dither pattern)
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_BITDEC_ON	0	R/W	Bit Reduction On/Off 0: Off 1: On

Note: This register is updated when the SC0_SCL1_VEN_A bit in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) is 1.

36.2.35 Write Detection Register (SC0_SCL1_WR7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_OVERFLOW
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SC0_RES_FLM_CNT[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SC0_RES_OVERFLOW	0	R	Line Buffer Overflow Detect 1: Line buffer has overflowed. 0: Line buffer has not overflowed.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SC0_RES_FLM_CNT[9:0]	0	R	Frame Number of Frame Being Accessed Frame number of the frame being accessed in the top field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1 or that in the top or bottom field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 0.

36.2.36 Write Address Register 1B (SC0_SCL1_WR8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_RES_BASE_B[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_BASE_B[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC0_RES_BASE_B [31:0]	0	R/W	Frame Buffer Base Address for Bottom Sets the start address of the frame buffer to store the frame data for the bottom field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1. For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.

Note: This register is updated when the SC0_SCL1_VEN_B bit in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) is 1.

36.2.37 Write Address Register 2B (SC0_SCL1_WR9)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	SC0_RES_LN_OFF_B[14:0]														
Initial value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SC0_RES_FLM_NUM_B[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	SC0_RES_LN_OFF_B [14:0]	2048	R/W	Frame Buffer Line Offset Address for Bottom Sets the line offset address for calculating the line start address for the bottom field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1. Line 0: SC0_RES_BASE_B Line 1: SC0_RES_BASE_B + SC0_RES_LN_OFF_B × 1 : Line n: SC0_RES_BASE + SC0_RES_LN_OFF_B × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SC0_RES_FLM_NUM_B [9:0]	1	R/W	Number of Frames of Buffer to be Written to for Bottom Field Number of frames defined by SC0_RES_FLM_NUM_B + 1 are used when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1.

Note: This register is updated when the SC0_SCL1_VEN_B bit in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) is 1.

36.2.38 Write Address Register 3B (SC0_SCL1_WR10)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	SC0_RES_FLM_OFF_B[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_FLM_OFF_B[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	SC0_RES_FLM_OFF_B[22:0]	524288	R/W	Frame Buffer Frame Offset Address for Bottom Sets the frame offset address for calculating the start address of each frame for the bottom field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1. Buffer 0: SC0_RES_BASE_B Buffer 1: SC0_RES_BASE_B + SC0_RES_FLM_OFF_B × 1 : Buffer n: SC0_RES_BASE_B + SC0_RES_FLM_OFF_B × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.

Note: This register is updated when the SC0_SCL1_VEN_B bit in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) is 1.

36.2.39 Write Detection Register B (SC0_SCL1_WR11)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SC0_RES_FLM_CNT_B[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SC0_RES_FLM_CNT_B[9:0]	0	R	Frame Number of Frame Being Accessed in Bottom Field Frame number of the frame being accessed in the bottom field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1.

36.2.40 Status Monitor 1 Register (SC0_SCL1_MON1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SC0_PBUF_NUM[1:0]		—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	SC0_PBUF_NUM[1:0]	All 0	R	Write pointer indicating the pointer buffer number corresponding to the location currently being written to.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

36.2.41 Pointer Buffer 0 Register (SC0_SCL1_PBUF0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_PBUF0_ADD[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_PBUF0_ADD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC0_PBUF0_ADD[31:0]	All 0	R	Start address of the write buffer pointed to by pointer buffer 0.

36.2.42 Pointer Buffer 1 Register (SC0_SCL1_PBUF1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_PBUF1_ADD[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_PBUF1_ADD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC0_PBUF1_ ADD[31:0]	All 0	R	Start address of the write buffer pointed to by pointer buffer 1.

36.2.43 Pointer Buffer 2 Register (SC0_SCL1_PBUF2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_PBUF2_ADD[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_PBUF2_ADD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC0_PBUF2_ ADD[31:0]	All 0	R	Start address of the write buffer pointed to by pointer buffer 2.

36.2.44 Pointer Buffer 3 Register (SC0_SCL1_PBUF3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_PBUF3_ADD[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_PBUF3_ADD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC0_PBUF3_ ADD[31:0]	All 0	R	Start address of the write buffer pointed to by pointer buffer 3.

36.2.45 Pointer Buffer and Field Information Register (SC0_SCL1_PBUF_FLD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	SC0_FLD_INF3	—	—	—	—	—	—	—	SC0_FLD_INF2
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SC0_FLD_INF1	—	—	—	—	—	—	—	SC0_FLD_INF0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	SC0_FLD_INF3	0	R	Top or bottom field information of the frame buffer pointed to by pointer buffer 3. 0: Bottom 1: Top
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SC0_FLD_INF2	0	R	Top or bottom field information of the frame buffer pointed to by pointer buffer 2. 0: Bottom 1: Top
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SC0_FLD_INF1	0	R	Top or bottom field information of the frame buffer pointed to by pointer buffer 1. 0: Bottom 1: Top
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_FLD_INF0	0	R	Top or bottom field information of the frame buffer pointed to by pointer buffer 0. 0: Bottom 1: Top

36.2.46 Pointer Buffer Control Register (SC0_SCL1_PBUF_CNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC0_PBUF_RST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SC0_PBUF_RST	0	R/W	Reset Control for Pointer Buffer 0: Pointer buffer is not reset. 1: Pointer buffer is reset.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: This register is updated when the SC0_SCL1_UPDATE_B bit in the SC0_SCL1 register update control register (SC0_SCL1_UPDATE) is 1.

36.2.47 Graphics 0 Register Update Control Register (GR0_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GR0_UPDATE	—	—	—	GR0_P_VEN	—	—	—	GR0_IBUS_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR0_UPDATE	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR0_P_VEN	0	R/WC1	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR0_IBUS_VEN	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

36.2.48 Frame Buffer Read Control Register (Graphics 0) (GR0_FLM_RD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR0_R_ENB
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR0_R_ENB	0	R/W	Frame Buffer Read Enable 0: Frame buffer reading is disabled. 1: Frame buffer reading is enabled.

Note: This register is updated when the GR0_IBUS_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.49 Frame Buffer Control Register 1 (Graphics 0) (GR0_FLM1)

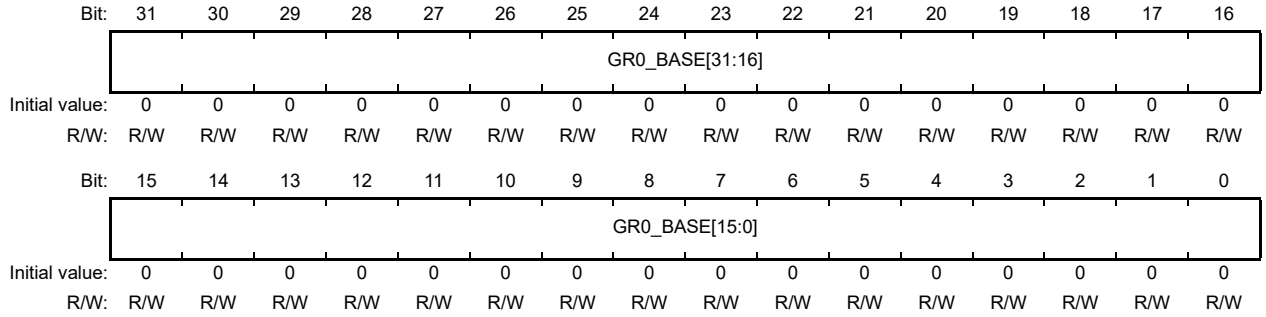
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_FLD_SEL	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR0_LN_OFF_DIR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR0_FLM_SEL[1:0]		—	—	—	GR0_IMR_FLM_INV	—	—	—	GR0_BST_MD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GR0_FLD_SEL	0	R/W	Enables or disables top or bottom field selection for the next frame buffer to be read. 0: Field selection is disabled. 1: Top or bottom field can be selected.
30 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR0_LN_OFF_DIR	0	R/W	Selects the line offset address direction of the frame buffer. 0: Increments the address by the line offset address. 1: Decrements the address by the line offset address.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	GR0_FLM_SEL[1:0]	0	R/W	Selects a frame buffer address setting signal. 0: Links to scaling-down process. (This setting is prohibited when separate write addresses are specified for the top and bottom fields; that is, SC0_RES_TB_ADD_MOD = 1 in SC0_SCL1_WR1.) 1: Links to distortion correction. 2: Links to distortion correction. 3: Links to pointer buffer.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR0_IMR_FLM_INV	0	R/W	Sets the frame buffer number for distortion correction. 0: Does not replace the numbers of the frames to be read. 1: Replaces the numbers of the frames to be read.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR0_BST_MD	0	R/W	Frame Buffer Burst Transfer Mode 0: 32-byte transfer 1: 128- byte transfer

Note: GR0_FLD_SEL is updated when the GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.
GR0_LN_OFF_DIR and GR0_IMR_FLM_INV are updated when the GR0_IBUS_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.
GR0_FLM_SEL is updated when the GR0_P_VEN and GR0_IBUS_VEN bits in the graphics 0 register update control register (GR0_UPDATE) are 1.
GR0_BST_MD is updated when the GR0_IBUS_VEN and GR0_P_VEN bits in the graphics 0 register update control register (GR0_UPDATE) are 1.

36.2.50 Frame Buffer Control Register 2 (Graphics 0) (GR0_FLM2)



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GR0_BASE [31:0]	0	R/W	Frame Buffer Base Address Sets the start address of the frame buffer where frame data is to be stored. GR0_BASE[4:3] and GR0_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower three bits should be fixed to 000.

Note: This register is updated when the GR0_IBUS_VEN and GR0_P_VEN bits in the graphics 0 register update control register (GR0_UPDATE) are 1.

36.2.51 Frame Buffer Control Register 3 (Graphics 0) (GR0_FLM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_FLD_NXT	GR0_LN_OFF[14:0]														
Initial value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR0_FLM_NUM[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GR0_FLD_NXT	0	R/W	Top or Bottom Field Selection for Next Frame Buffer 0: Bottom 1: Top
30 to 16	GR0_LN_OFF[14:0]	2048	R/W	Frame Buffer Line Offset Address Sets the line offset address for calculating the start address of each line. Line 0: GR0_BASE Line 1: GR0_BASE + GR0_LN_OFF × 1 : Line n: GR0_BASE + GR0_LN_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	GR0_FLM_NUM[9:0]	1	R/W	Frame Number of Frame Buffer Manually set the frame number when GR0_FLM_SEL = 1.

Note: GR0_FLD_NXT is updated when the GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.
GR0_LN_OFF[14:0] and GR0_FLM_NUM[9:0] are updated when the GR0_IBUS_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.52 Frame Buffer Control Register 4 (Graphics 0) (GR0_FLM4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	GR0_FLM_OFF[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_FLM_OFF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	GR0_FLM_OFF[22:0]	524288	R/W	Frame Buffer Frame Offset Address Specifies the frame offset address used for calculating the start address of each frame buffer when more than one buffer is used. Buffer 0: GR0_BASE Buffer 1: GR0_BASE + GR0_FLM_OFF × 1 : Buffer n: GR0_BASE + GR0_FLM_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.

Note: This register is updated when the GR0_IBUS_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.53 Frame Buffer Control Register 5 (Graphics 0) (GR0_FLM5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR0_FLM_LNUM[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR0_FLM_LOOP[10:0]										
Initial value:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR0_FLM_LNUM[10:0]	0	R/W	Sets number of lines in a frame Number of lines is (GR0_FLM_LNUM + 1).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR0_FLM_LOOP[10:0]	1023	R/W	Number of lines when reading the addresses repeatedly by returning to the start address after reaching the end address. (GR0_FLM_LOOP + 1) lines are read.

Note: This register is updated when the GR0_IBUS_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.54 Frame Buffer Control Register 6 (Graphics 0) (GR0_FLM6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_FORMAT[3:0]				—	GR0_HW[10:0]										
Initial value:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_YCC_SWAP[2:0]			GR0_RDSWA[2:0]			—	GR0_CNV444_MD	—	—	GR0_STA_POS[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	GR0_FORMAT [3:0]	8	R/W	Sets the format of the frame buffer read signal. 0: RGB565 1: RGB888 2: αRGB1555 3: αRGB4444 4: αRGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8: YCbCr422 9: YCbCr444 10: RGBα5551 11: RGBα8888 12 to 15: Setting prohibited
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26 to 16	GR0_HW [10:0]	0	R/W	Sets the width of the horizontal valid period. The width is (GR0_HW + 1) pixels. Note: The set value should be equal to or more than two.
15 to 13	GR0_YCC_SWAP[2:0]	0	R/W	Controls swapping of data read from buffer in the YCbCr422 format. 0: Cb/Y0/Cr/Y1 1: Y0/Cb/Y1/Cr 2: Cr/Y0/Cb/Y1 3: Y0/Cr/Y1/Cb 4: Y1/Cr/Y0/Cb 5: Cr/Y1/Cb/Y0 6: Y1/Cb/Y0/Cr 7: Cb/Y1/Cr/Y0

Bit	Bit Name	Initial Value	R/W	Description
12 to 10	GR0_ RDSWA [2:0]	0	R/W	8-Bit, 16-Bit, or 32-Bit Swap Setting These bits control swapping in frame buffer reading as follows. Bit 0 0: Swapped in 8-bit units. 1: Not swapped in 8-bit units. Bit 1 0: Swapped in 16-bit units. 1: Not swapped in 16-bit units. Bit 2 0: Swapped in 32-bit units. 1: Not swapped in 32-bit units. According to the setting of these bits, data is swapped as follows. Each number in parentheses ((1) to (8)) indicates 8-bit data. 000: (1) (2) (3) (4) (5) (6) (7) (8) [Not swapped] 001: (2) (1) (4) (3) (6) (5) (8) (7) [Swapped in 8-bit units] 010: (3) (4) (1) (2) (7) (8) (5) (6) [Swapped in 16-bit units] 011: (4) (3) (2) (1) (8) (7) (6) (5) [Swapped in 16-bit units + 8-bit units] 100: (5) (6) (7) (8) (1) (2) (3) (4) [Swapped in 32-bit units] 101: (6) (5) (8) (7) (2) (1) (4) (3) [Swapped in 32-bit units + 8-bit units] 110: (7) (8) (5) (6) (3) (4) (1) (2) [Swapped in 32-bit units + 16-bit units] 111: (8) (7) (6) (5) (4) (3) (2) (1) [Swapped in 32-bit units + 16-bit units + 8-bit units]
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	GR0_ CNV444_ MD	0	R/W	Sets the interpolation mode for YCbCr422 to YCbCr444 conversion. 0: Hold interpolation 1: Average interpolation
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	GR0_STA_ POS[5:0]	0	R/W	Sets the amount of data to be skipped through. Specifically data amount equal to the amount indicated by GR0_STA_POS is skipped from the start of the line.

Note: GR0_YCC_SWAP, GR0_CNV444, and GR0_STA_POS are updated when GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.
GR0_ RDSWA is updated when the GR0_UPDATE bit in the graphics 0 register update control register (GR0_UPDATE) is 1.
GR0_FORMAT and GR0_HW are updated when GR0_IBUS_VEN and GR0_P_VEN bits in the graphics 0 register update control register (GR0_UPDATE) are 1.

36.2.55 Alpha Blending Control Register 1 (Graphics 0) (GR0_AB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	GR0_GRC_DISP_ON	—	—	GR0_DISP_SEL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR0_GRC_DISP_ON	0	R/W	Turns on/off frame-line display of the graphics image area. 0: Frame-line display off 1: Frame-line display on
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	GR0_DISP_SEL[1:0]	0	R/W	Selects the graphics display mode. 0: Background color display (GR0_BASE) 1: Lower-layer graphics display When displaying video image or enlarged graphics, select this setting. 2: Current graphics display When displaying graphics, select this setting. 3: Blended display of lower-layer graphics and current graphics* Note: * Select this setting whenever chroma-key processing is to proceed. Since only current graphics are to be displayed by chroma-key processing, set the α values for both pixels to be subject to chroma-keying and pixels not to be subject to chroma-keying to 255.

Note: This register is updated when GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.56 Alpha Blending Control Register 2 (Graphics 0) (GR0_AB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR0_GRC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR0_GRC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR0_GRC_VS[10:0]	0	R/W	Vertical Start Position of Graphics Image Area. Note: The set value should be four or more (lines). GR0_GRC_VS + GR0_GRC_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR0_GRC_VW[10:0]	0	R/W	Vertical Width of Graphics Image Area.

Note: This register is updated when the GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.57 Alpha Blending Control Register 3 (Graphics 0) (GR0_AB3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR0_GRC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR0_GRC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR0_GRC_HS[10:0]	0	R/W	Horizontal Start Position of Graphics Image Area. Note: The set value should be 16 or more (clock cycles). GR0_GRC_HS + GR0_GRC_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR0_GRC_HW[10:0]	0	R/W	Horizontal Width of Graphics Image Area. Note: For displaying an image with 1- or 2-pixel horizontal width, set GR0_HW to 2 and GR0_GRC_HW to 1 (1 pixel) or 2 (2 pixels).

Note: This register is updated when the GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.58 Alpha Blending Control Register 7 (Graphics 0) (GR0_AB7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR0_CK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR0_CK_ON	0	R/W	CLUT-Index/RGB-Index Chroma-Key Processing On/Off 0: Off 1: On

Note: This register is updated when the GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.59 Alpha Blending Control Register 8 (Graphics 0) (GR0_AB8)

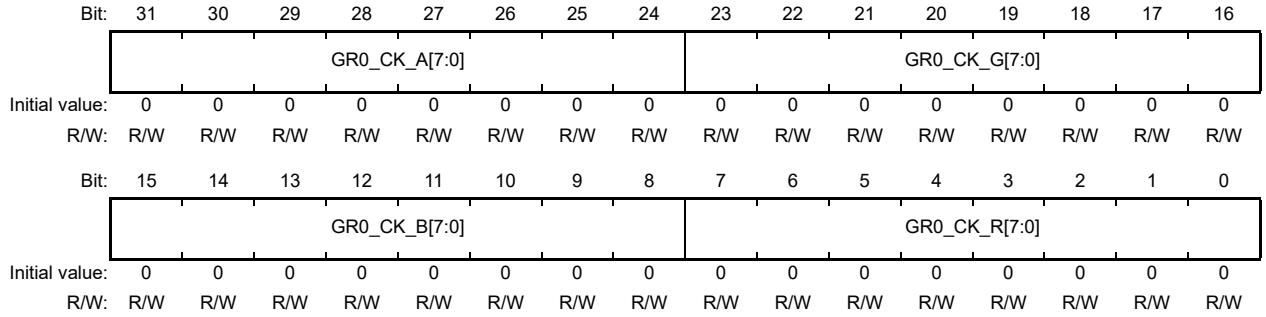
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_CK_KCLUT[7:0]								GR0_CK_KG[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_CK_KB[7:0]								GR0_CK_KR[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR0_CK_KCLUT[7:0]	0	R/W	CLUT Signal for CLUT-Index Chroma-Key Processing CLUT: Unsigned 8 bits (0 to 255 [LSB])
23 to 16	GR0_CK_KG[7:0]	0	R/W	G Signal for RGB-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR0_CK_KB[7:0]	0	R/W	B Signal for RGB-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR0_CK_KR[7:0]	0	R/W	R Signal for RGB-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when the GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.60 Alpha Blending Control Register 9 (Graphics 0) (GR0_AB9)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR0_CK_A [7:0]	0	R/W	Replaced Alpha Signal after RGB-Index Chroma-Key Processing α: Unsigned 8 bits (0 to 255 [LSB]) Note: These bits should always be set to 255 to display the current graphics only.
23 to 16	GR0_CK_G [7:0]	0	R/W	Replaced G Signal after RGB-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR0_CK_B [7:0]	0	R/W	Replaced B Signal after RGB-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR0_CK_R [7:0]	0	R/W	Replaced R Signal after RGB-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when the GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.61 Alpha Blending Control Register 10 (Graphics 0) (GR0_AB10)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_A0[7:0]								GR0_G0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_B0[7:0]								GR0_R0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR0_A0 [7:0]	0	R/W	CLUT1 α0 Signal Replaced with α signal when in the CLUT1 format and CLUT1 = 0. Replaced with α signal when in the αRGB1555/RGBα5551 format and α = 0. Note: These bits should always be set to 255 to display the current graphics only.
23 to 16	GR0_G0 [7:0]	0	R/W	CLUT1 G0 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 0.
15 to 8	GR0_B0 [7:0]	0	R/W	CLUT1 B0 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 0.
7 to 0	GR0_R0 [7:0]	0	R/W	CLUT1 R0 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 0.

Note: This register is updated when the GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.62 Alpha Blending Control Register 11 (Graphics 0) (GR0_AB11)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_A1[7:0]								GR0_G1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_B1[7:0]								GR0_R1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR0_A1 [7:0]	0	R/W	CLUT1 α1 Signal Replaced with α signal when in the CLUT1 format and CLUT1 = 1. Replaced with α signal when in the αRGB1555/RGBα5551 format and α = 1. Note: These bits should always be set to 255 to display the current graphics only.
23 to 16	GR0_G1 [7:0]	0	R/W	CLUT1 G1 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 1.
15 to 8	GR0_B1 [7:0]	0	R/W	CLUT1 B1 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 1.
7 to 0	GR0_R1 [7:0]	0	R/W	CLUT1 R1 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 1.

Note: This register is updated when the GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.63 Background Color Control Register (Graphics 0) (GR0_BASE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR0_BASE_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_BASE_B[7:0]								GR0_BASE_R[7:0]							
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR0_BASE_ G[7:0]	0	R/W	Background Color G/Y Signal G/Y: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR0_BASE_ B[7:0]	128	R/W	Background Color B/Cb Signal B: Unsigned 8 bits (0 to 255 [LSB]) Cb: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])
7 to 0	GR0_BASE_ R[7:0]	128	R/W	Background Color R/Cr Signal R: Unsigned 8 bits (0 to 255 [LSB]) Cr: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])

Note: This register is updated when the GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.2.64 CLUT Table Control Register (Graphics 0) (GR0_CLUT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR0_CLT_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR0_CLT_SEL	0	R/W	CLUT Table Select Signal 0: Selects CLUT table 0. Referring to the CLUT table 0 value to expand to αRGB8888 The CPU side can read-access or write-access to the CLUT table 1. 1: Selects CLUT table 1. Referring to the CLUT table 1 value to expand to αRGB8888 The CPU side can read-access or write-access to the CLUT table 0.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: This register is updated when the GR0_P_VEN bit in the graphics 0 register update control register (GR0_UPDATE) is 1.

36.3 Usage Method

36.3.1 Scaling Setting Example for 525i Video Input and VGA-Size (640 x 480) Video Output

(1) Angles of View for Input and Output

This section describes an example of setting the signals of the input and output angles of view shown in Table 36.40.

Here, the over-scan rate is assumed to be 100%.

Table 36.40 Input and Output Angles for 525i Video Input and VGA-Size (640 x 480) Video Output

Input Signal	Output Signal	Signal Format	Rotation	Buffer Planes	Scaling Filter
1440 × 240	640 × 480	YCbCr	Normal	Two planes	2-tap linear

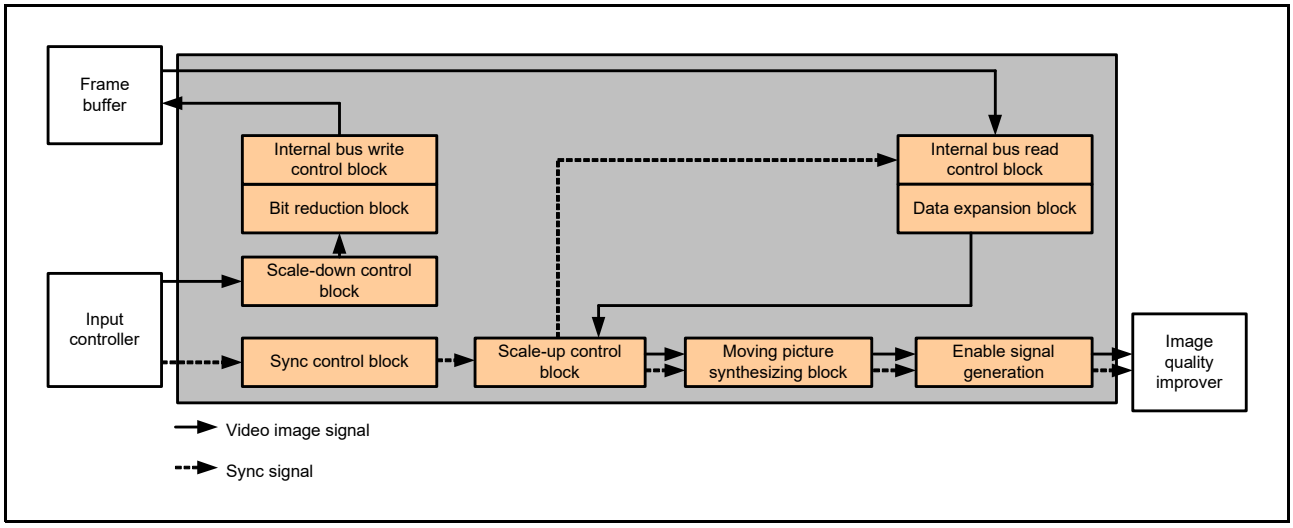


Figure 36.18 Signal Paths for Displaying Input Video Image

(2) Horizontal Scaling (Horizontal Scale Down, Scaling Filter: 2-Tap Linear)

The scaling rate for folding can be calculated as shown below.

RATIO_org = round (1440 ÷ 640 × 4096) = 9216

σ = (9216 × (640 − 1) − (1440 − 1) × 4096) ÷ (640 − 1) = −8.01

Horizontal scaling ratio = roundup (9216 − (−8.01)) = 9225

(3) Vertical Scaling (Vertical Scale Up, Scaling Filter: 2-Tap Linear)

The scaling rate for folding can be calculated as shown below.

RATIO_org = round (240 ÷ 480 × 4096) = 2048

σ = (2048 × (480 − 1) − (240 − 1) × 4096) ÷ (480 − 1) = 4.27

Vertical scaling ratio = round (2048 − (4.07)) = 2044

(4) Setting Frame Buffer Access Area

Since video data is written to the frame buffer after scaled down, the write size is 640 × 240 pixels.

The frame buffer area required is 640 pixels or more for line offset and line offset × 240 pixels or more for frame offset.

Here, the frame buffer work area is assumed to be 1024 × 256 pixels.

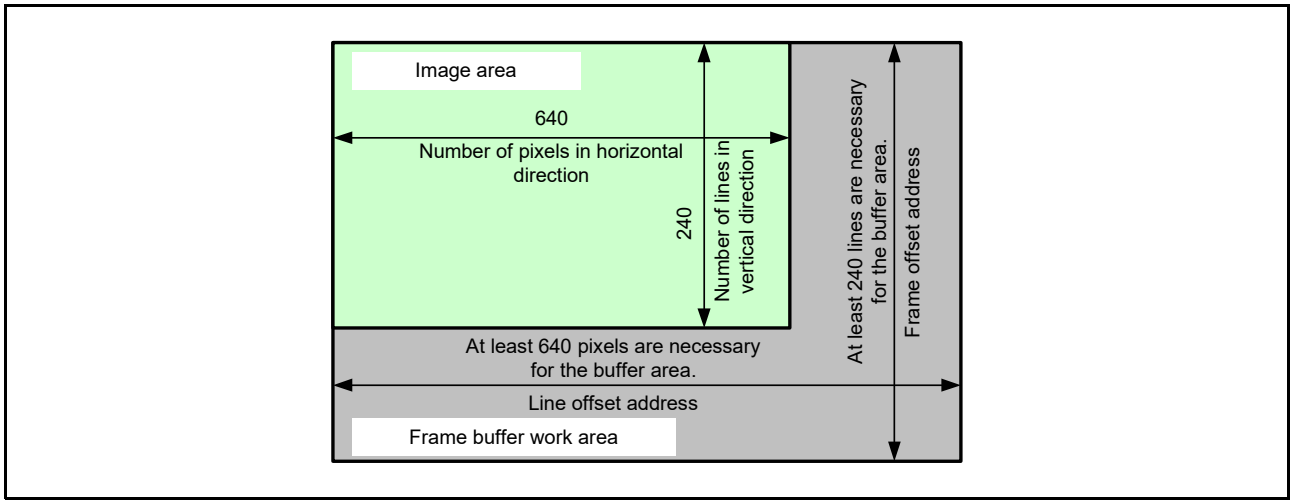


Figure 36.19 Frame Buffer Access Area Setting

Since the frame buffer is accessed in 64-bit units, YCbCr422 (16 bits) is accessed in 4-pixel units.

The line offset address values to be set are:

SC_RES_LN_OFF[14:0] = 1024 × 2 = 2048
GR_LN_OFF[14:0] = 1024 × 2 = 2048

The frame offset address values to be set are:

SC_RES_FLM_OFF[22:0] = SC_RES_LN_OFF[14:0] × 256 = 524288
GR_FLM_OFF[22:0] = GR_LN_OFF[14:0] × 256 = 524288

(5) Register Setting Example

Table 36.41 Register Setting Example for 525i Video Input and VGA-Size Video Output

Register Name	Bit Name	Settings	Remarks
Synchronization Control			
SC_SCL0_FRC3	SC_RES_VS_SEL	0	External Vsync selected
SC_SCL0_FRC4	SC_RES_FH[10:0]	799	Horizontal period width of output signal (period width = set value + 1)
Size of Angle of View			
SC_SCL0_DS2	SC_RES_VS[10:0]	15	Vertical capture start position of input signal
SC_SCL0_DS2	SC_RES_VW[10:0]	240	Vertical capture width of input signal
SC_SCL0_DS3	SC_RES_HS[10:0]	244	Horizontal capture start position of input signal
SC_SCL0_DS3	SC_RES_HW[11:0]	1440	Horizontal capture width of input signal
SC_SCL0_FRC6	SC_RES_F_VS[10:0]	35	Vertical valid start position of full screen
SC_SCL0_FRC6	SC_RES_F_VW[10:0]	480	Vertical valid width of full screen
SC_SCL0_FRC7	SC_RES_F_HS[10:0]	144	Horizontal valid start position of full screen
SC_SCL0_FRC7	SC_RES_F_HW[10:0]	640	Horizontal valid width of full screen
SC_SCL0_US2	SC_RES_P_VS[10:0]	35	Vertical valid start position of output image
SC_SCL0_US2	SC_RES_P_VW[10:0]	480	Vertical valid width of output image
SC_SCL0_US3	SC_RES_P_HS[10:0]	144	Horizontal valid start position of output image
SC_SCL0_US3	SC_RES_P_HW[10:0]	640	Horizontal valid width of output image
Scaling Setting			
SC_SCL0_DS4	SC_RES_DS_H_RATIO [15:0]	9224	Horizontal scaling-down because SC_RES_DS_H_RATIO is equal to or larger than 4096
SC_SCL0_DS1	SC_RES_DS_H_ON	1	Horizontal scaling-down on
SC_SCL0_US1	SC_RES_US_H_ON	0	Horizontal scaling-up off
SC_SCL0_US5	SC_RES_US_H_RATIO [15:0]	4096	Horizontal scaling-up off because SC_RES_US_H_RATIO is equal to or larger than 4096
SC_SCL0_DS1	SC_RES_DS_V_ON	0	Vertical scaling-down off
SC_SCL0_US1	SC_RES_US_V_ON	1	Vertical scaling-up on
SC_SCL0_DS6	SC_RES_V_RATIO[15:0]	2044	Vertical scaling-up because SC_RES_V_RATIO is smaller than 4096
SC_SCL0_DS7	SC_RES_OUT_VW[10:0]	240	Vertical valid input width because the vertical scaling-down function is off
SC_SCL0_DS7	SC_RES_OUT_HW[10:0]	640	Horizontal image size after horizontal scaling-down
SC_SCL0_US4	SC_RES_IN_VW[10:0]	240	Vertical width of frame buffer read
SC_SCL0_US4	SC_RES_IN_HW[10:0]	640	Horizontal width of frame buffer read
IP Conversion Setting			
SC_SCL0_DS5	SC_RES_TOP_ INIPHASE[11:0]	2048	Top field adjusted by 0.5-line phase
SC_SCL0_DS5	SC_RES_BTM_ INIPHASE[11:0]	0	No phase adjustment for bottom field
SC_SCL0_FRC5	SC_RES_FLD_DLY_SEL	1	IP conversion with two planes of frame buffer used for vertical scaling-up
Frame Buffer Write Setting			
SC_SCL1_WR1	SC_RES_DS_WR_MD [2:0]	0	Normal write mode for rotation control
SC_SCL1_WR1	SC_RES_MD[1:0]	0	Frame buffer write format YCbCr422 (16 bits)
SC_SCL1_WR2	SC_RES_BASE[31:0]	0	Frame buffer write start address (0 in setting example)
SC_SCL1_WR3	SC_RES_LN_OFF[14:0]	2048	Frame buffer write line offset

Table 36.41 Register Setting Example for 525i Video Input and VGA-Size Video Output

Register Name	Bit Name	Settings	Remarks
SC_SCL1_WR3	SC_RES_FLM_NUM[9:0]	1	Two planes of frame buffer used
SC_SCL1_WR4	SC_RES_FLN_OFF[22:0]	524288	Frame buffer write frame offset
SC_SCL1_WR5	SC_RES_WENB	1	Frame buffer write enabled
Frame Buffer Read Setting			
GR_FLM1	GR_FLM_SEL[1:0]	0	Frame number for frame buffer write output
GR_FLM2	GR_BASE[31:0]	0	Conforming to frame buffer write setting
GR_FLM3	GR_LN_OFF[14:0]	2048	Conforming to frame buffer write setting
GR_FLM4	GR_FLM_OFF[22:0]	524288	Conforming to frame buffer write setting
GR_FLM6	GR_FORMAT[3:0]	8	Frame buffer read format YCbCr422
GR_FLM_RD	GR_R_ENB	1	Frame buffer read enabled
GR_FLM6	GR_CNV444_MD	1	Mean value interpolation in YCbCr422→ YCbCr444 conversion
Scaling-up Selection			
SC_SCL0_US8	SC_RES_IBUS_SYNC_SEL	0	Scaled-up video signal displayed
GR_AB1	GR_DISP_SEL[1:0]	1	Scaling display selected

36.3.2 Scaling Setting Example for Graphics Display

(1) Angle of View for Graphics Display

This section describes an example of setting the signals of the input and output angles of view shown in Table 36.42.

Table 36.42 Input and Output Angles of View for Graphics Display

Graphics Size	Output Signal	Graphics Signal Format
640 × 480	640 × 480	RGB888

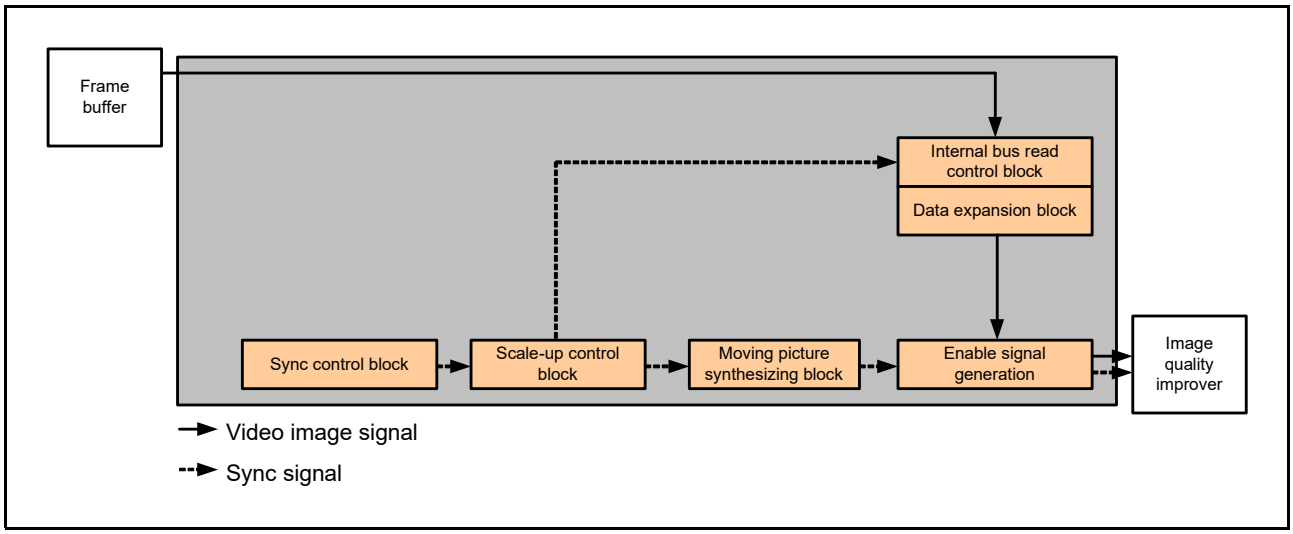


Figure 36.20 Signal Paths for Graphics Display

(2) Setting Frame Buffer Access Area

In the frame buffer in which graphics data is stored, graphics data needs to be expanded in the area of 640 × 480 pixels or larger.

Here, the frame buffer area in which graphics data is expanded is assumed to be 640 × 480 pixels.

Since the frame buffer is accessed in 64-bit units, RGB888 (32 bits) is accessed in 2-pixel units.

The line offset address values to be set are:

GR_LN_OFF[14:0] = 640 × 4 = 2560

The frame offset address values to be set are:

GR_FLM_OFF[22:0] = GR_LN_OFF[14:0] × 480 = 1228800

(3) Register Setting Example

Table 36.43 Register Setting Example for Graphics Display

Register Name	Bit Name	Settings	Remarks
Synchronization Control			
SC_SCL0_FRC3	SC_RES_VS_SEL	1	Free-running Vsync selected (when the appropriate input signal is available, an external sync can also be selected)
SC_SCL0_FRC4	SC_RES_FV[10:0]	524	Vertical period width of output signal (period width = set value + 1)
SC_SCL0_FRC4	SC_RES_FH[10:0]	799	Horizontal period width of output signal (period width = set value + 1)
Size of Angle of View			
SC_SCL0_FRC6	SC_RES_F_VS[10:0]	35	Vertical valid start position of full screen
SC_SCL0_FRC6	SC_RES_F_VW[10:0]	480	Vertical valid width of full screen
SC_SCL0_FRC7	SC_RES_F_HS[10:0]	144	Horizontal valid start position of full screen
SC_SCL0_FRC7	SC_RES_F_HW[10:0]	640	Horizontal valid width of full screen
GR_AB2	GR_GRC_VS[10:0]	35	Vertical valid start position of graphics output
GR_AB2	GR_GRC_VW[10:0]	480	Vertical valid width of graphics output
GR_AB3	GR_GRC_HS[10:0]	144	Horizontal valid start position of graphics output
GR_AB3	GR_GRC_HW[10:0]	640	Horizontal valid width of graphics output
Frame Buffer Read Setting			
GR_FLM1	GR_FLM_SEL[1:0]	1	Frame number setting with register
GR_FLM3	GR_FLM_NUM[9:0]	0	Frame number of frame buffer (0 in setting example)
GR_FLM5	GR_FLM_LNUM[9:0]	479	Number of graphics lines (number of lines = set value + 1)
GR_FLM6	GR_HW[9:0]	639	Horizontal valid width of graphics (valid width = set value + 1)
GR_FLM2	GR_BASE[31:0]	0	Conforming to graphics expansion setting (0 in setting example)
GR_FLM3	GR_LN_OFF[14:0]	2560	Conforming to graphics expansion setting
Frame Buffer Read Setting			
GR_FLM4	GR_FLM_OFF[22:0]	1228800	Conforming to graphics expansion setting
GR_FLM6	GR_FORMAT[3:0]	1	Frame buffer read format RGB888
GR_FLM_RD	GR_R_ENB	1	Frame buffer read enabled
Scaling-up Selection			
SC_SCL0_US8	SC_RES_IBUS_SYNC_SEL	1	Graphics output displayed
GR_AB1	GR_DISP_SEL[1:0]	2	Graphics display selected

36.3.3 Scaling Setting Example for Scaled-up Graphics Display

(1) Angles of View for Input and Output

This section describes an example of setting the signals of the input and output angles of view shown in Table 36.44.

Table 36.44 Input and Output Angles of View for Scaled-up Graphics Display

Graphics Size	Output Signal	Graphics Signal Format
640 × 480	800 × 600	RGB565

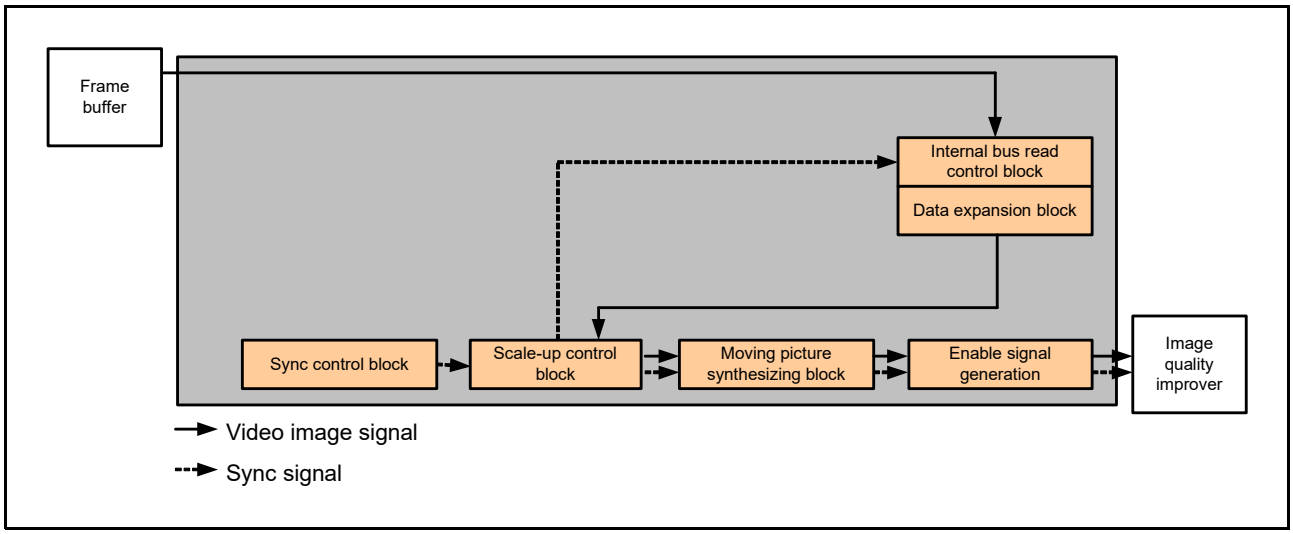


Figure 36.21 Signal Paths for Scaled-up Graphics Display

(2) Horizontal Scaling (Horizontal Scale Up, Scaling Filter: 2-Tap Linear)

The scaling rate for folding can be calculated as shown below.

RATIO_org = round (640 ÷ 800 × 4096) = 3277
σ = (3277 × (800 − 1) − (640 − 1) × 4096) ÷ (800 − 1) = 1.23
Horizontal scaling ratio = round (3277 − (1.23)) = 3276

(3) Vertical Scaling (Vertical Scale Up, Scaling Filter: 2-Tap Linear)

The scaling rate for folding can be calculated as shown below.

RATIO_org = round (480 ÷ 600 × 4096) = 3277
σ = (3277 × (600 − 1) − (480 − 1) × 4096) ÷ (600 − 1) = 1.57
Vertical scaling ratio = round (3277 − (1.57)) = 3275

(4) Setting Frame Buffer Access Area

In the frame buffer in which graphics data is stored, graphics data needs to be expanded in the area of 640 × 480 pixels or larger.

Here, the frame buffer area in which graphics data is expanded is assumed to be 640 × 480 pixels.

Since the frame buffer is accessed in 64-bit units, RGB565 (16 bits) is accessed in 4-pixel units.

The line offset address values to be set are:

GR_LN_OFF[14:0] = 640 × 2 = 1280

The frame offset address values to be set are:

GR_FLM_OFF[22:0] = GR_LN_OFF[14:0] × 480 = 614400

(5) Register Setting Example

Table 36.45 Register Setting Example for Scaled-up Graphics Display

Register Name	Bit Name	Settings	Remarks
Synchronization Control			
SC_SCL0_FRC3	SC_RES_VS_SEL	1	Free-running Vsync selected (when the appropriate input signal is available, an external sync can also be selected)
SC_SCL0_FRC4	SC_RES_FV[10:0]	668	Vertical period width of output signal (period width = set value + 1)
SC_SCL0_FRC4	SC_RES_FH[10:0]	1040	Horizontal period width of output signal (period width = set value + 1)
Size of Angle of View			
SC_SCL0_FRC6	SC_RES_F_VS[10:0]	27	Vertical valid start position of full screen
SC_SCL0_FRC6	SC_RES_F_VW[10:0]	600	Vertical valid width of full screen
SC_SCL0_FRC7	SC_RES_F_HS[10:0]	216	Horizontal valid start position of full screen
SC_SCL0_FRC7	SC_RES_F_HW[10:0]	800	Horizontal valid width of full screen
SC_SCL0_US2	SC_RES_P_VS[10:0]	27	Vertical valid start position of image output
SC_SCL0_US2	SC_RES_P_VW[10:0]	600	Vertical valid width of image output
SC_SCL0_US3	SC_RES_P_HS[10:0]	216	Horizontal valid start position of image output
SC_SCL0_US3	SC_RES_P_HW[10:0]	800	Horizontal valid width of image output
Scaling Setting			
SC_SCL0_US5	SC_RES_US_H_RATIO[15:0]	3276	Horizontal scaling-up because SC_RES_US_H_RATIO is smaller than 4096
SC_SCL0_DS6	SC_RES_V_RATIO[15:0]	3275	Vertical scaling-up because SC_RES_V_RATIO is smaller than 4096
SC_SCL0_US1	SC_RES_US_H_ON	1	Horizontal scaling-up on
SC_SCL0_US1	SC_RES_US_V_ON	1	Vertical scaling-up on
SC_SCL0_US4	SC_RES_IN_VW[10:0]	480	Vertical width of frame buffer read
SC_SCL0_US4	SC_RES_IN_HW[10:0]	640	Horizontal width of frame buffer read
Frame Buffer Read Setting			
GR_FLM1	GR_FLM_SEL[1:0]	1	Frame number setting with register
GR_FLM3	GR_FLM_NUM[9:0]	0	Frame number of frame buffer (0 in setting example)
GR_FLM2	GR_BASE[31:0]	0	Conforming to graphics expansion setting (0 in setting example)
GR_FLM3	GR_LN_OFF[14:0]	1280	Conforming to graphics expansion setting
GR_FLM4	GR_FLM_OFF[22:0]	614400	Conforming to graphics expansion setting
GR_FLM6	GR_FORMAT[3:0]	0	Frame buffer read format RGB565
GR_FLM_RD	GR_R_ENB	1	Frame buffer read enabled
Scaling-up Selection			
SC_SCL0_US8	SC_RES_IBUS_SYNC_SEL	0	Scaled-up video signal displayed
GR_AB1	GR_DISP_SEL[1:0]	1	Scaling display selected

37. Video Display Controller 6 (4): Image Quality Improver

37.1 Image Quality Improver

37.1.1 Overview of Functions

The image quality improver subjects scaled YCbCr signals to black stretching, LTI/sharpness processing, and GBR conversion by using a color matrix.

The image quality improver does not act on RGB signals.

Figure 37.1 is a functional block diagram of the image quality improver. Image quality improver 0 is connected to scaler 0.

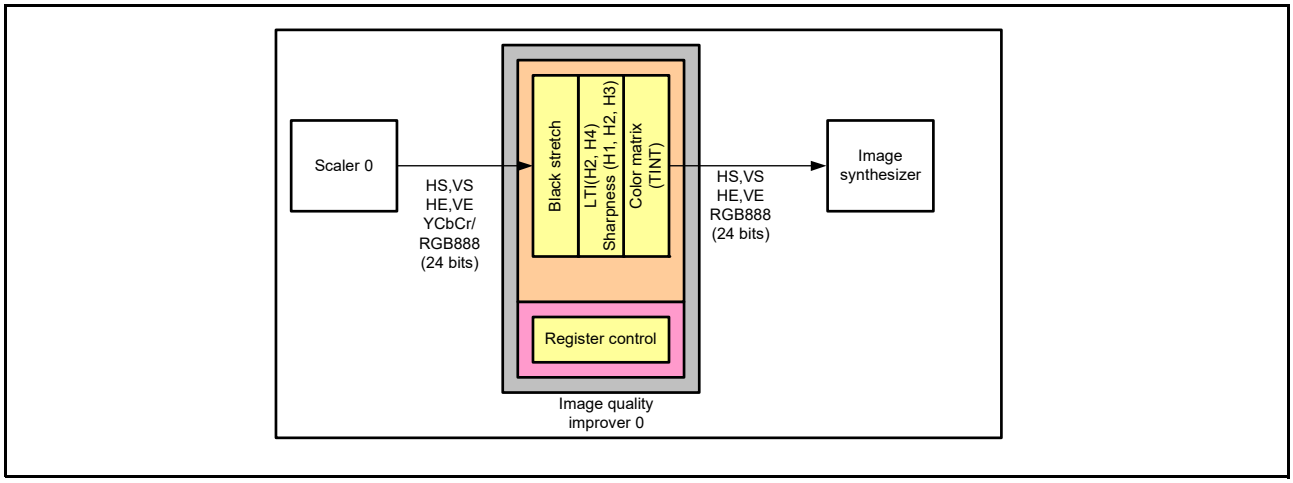


Figure 37.1 Functional Block Diagram of Image Quality Improver

37.1.2 Register Update Control

The control register for image quality improver controls the update timing entirely by vertical synchronous signals.

The vertical synchronous signal launched after the update control register is set to 1 is reflected in various registers, following which the update control register is automatically cleared to 0.

Note that registers for the improver can be identified by the number in the register name like ADJ0_xxxx. In the sections except for Register Description, however, the number is omitted like ADJ_xxxx for convenience sake.

Table 37.1 Register Update Control

Register Name	Bit Name	Initial Value	Description
ADJ_UPDATE	ADJ_VEN	0	Image Quality Improver Register Update 0: Register is not updated. 1: Register is updated by launch of vertical synchronous signal.

37.1.3 Black Stretch

Black stretch refers to the black stretch correction of the Y signal of the input video signal of YCbCr format.

Correction of the Y signal is done by adjusting the time constant, depth (gain), and start point.

Figure 37.2 is a drawing illustrating black stretch correction.

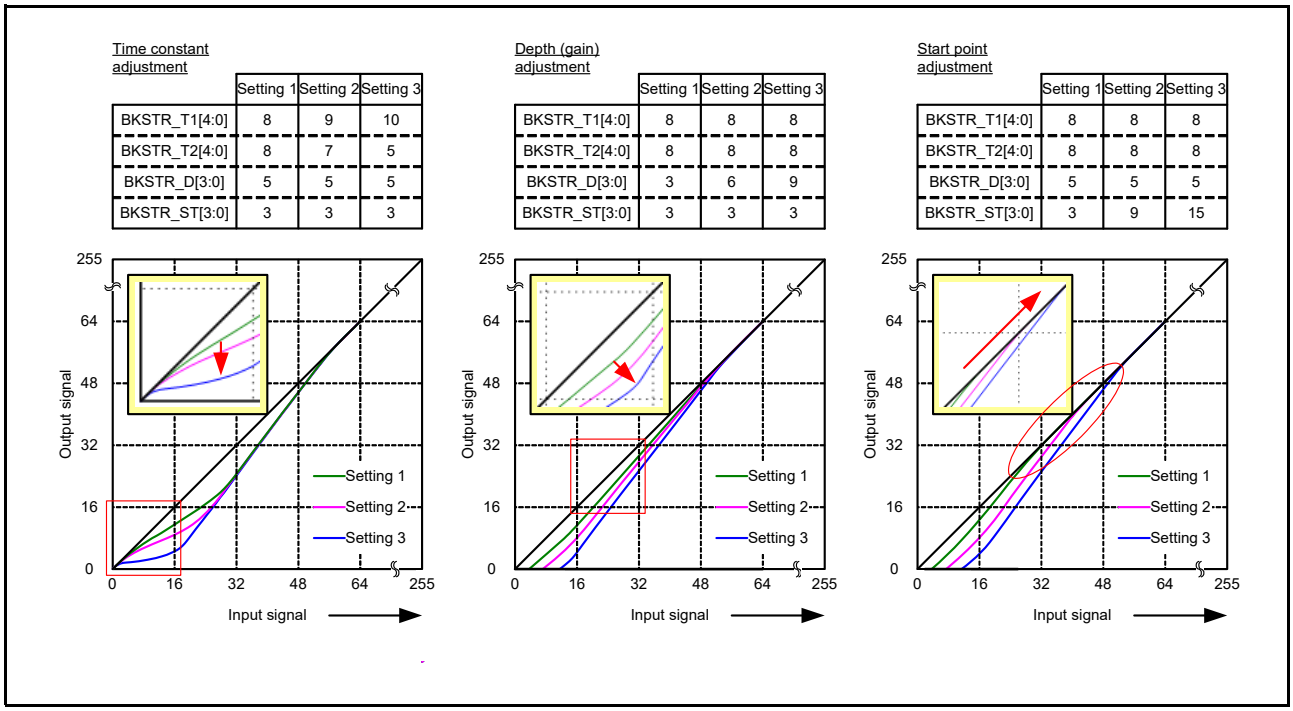


Figure 37.2 Black Stretch Correction (With Sample Settings)

Table 37.2 Black Stretch Control

Register Name	Bit Name	Initial Value	Description
ADJ_BKSTR_SET	BKSTR_ON	0	Black Stretch On/Off Control 0: Black Stretch Off 1: Black Stretch On
ADJ_BKSTR_SET	BKSTR_ST[3:0]	0	Black Stretch Start Point 0 (low) to 15 (high)
ADJ_BKSTR_SET	BKSTR_T1[4:0]	0	Black Stretch Time Constant (T1) 0 (small) to 31 (large)
ADJ_BKSTR_SET	BKSTR_T2[4:0]	0	Black Stretch Time Constant (T2) 0 (small) to 30 (large), 31: Setting prohibited
ADJ_BKSTR_SET	BKSTR_D[3:0]	0	Black Stretch Depth 0 (shallow) to 15 (deep)

37.1.4 Enhancer

The enhancer subjects the scaled Y signal input to transient improvement (LTI) and sharpness processing in the horizontal direction.

(1) Enhancer Area Specification

The operating area of the enhancer is specified with reference to the rising edges of the Hsync signal and Vsync signal. ENH_HS should be set to four or greater clocks, and ENH_VS should be set to two or greater lines. Figure 37.3 shows enhancer area setting.

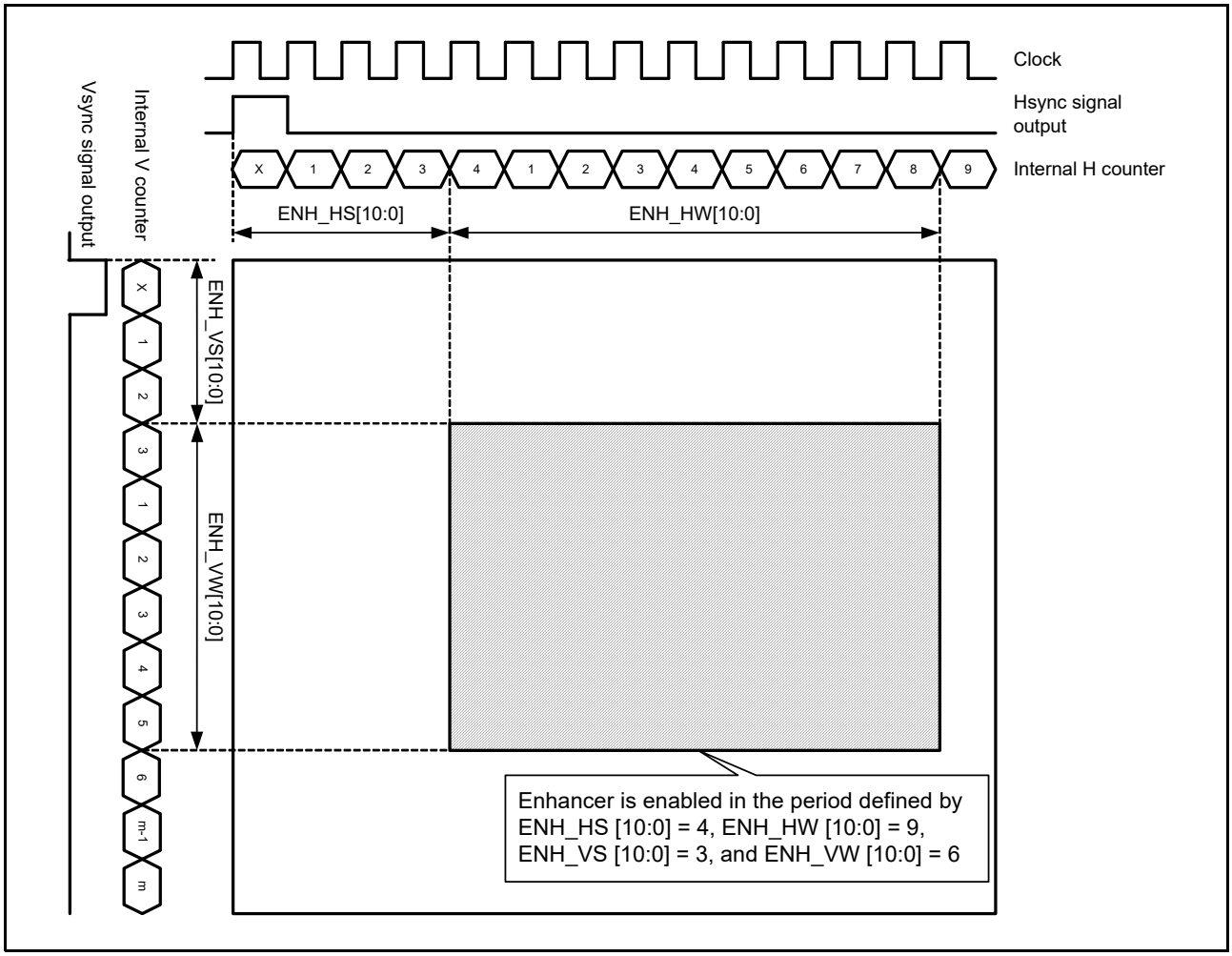


Figure 37.3 Period when Enhancer is Enabled

Setting ENH_DISP_ON to 1 displays the enhancer-enabled area with frame lines.

Table 37.3 Enhancer Area Control

Register Name	Bit Name	Initial Value	Description
ADJ_ENH_TIM1	ENH_MD	1	Operating Mode 0: RGB mode 1: YCbCr mode
ADJ_ENH_TIM2	ENH_VS[10:0]	0	Start Position of Vertical Valid Image Area in Enhancer-Enabled Area Note: Set to 2 or greater lines.
ADJ_ENH_TIM2	ENH_VW[10:0]	0	Width of Vertical Valid Image Area in Enhancer-Enabled Area
ADJ_ENH_TIM3	ENH_HS[10:0]	0	Start Position of Horizontal Valid Image Area in Enhancer-Enabled Area Note: Set to 4 or greater clocks.
ADJ_ENH_TIM3	ENH_HW[10:0]	0	Width of Horizontal Valid Image Area in Enhancer-Enabled Area
ADJ_ENH_TIM1	ENH_DISP_ON	0	Frame Line Display in Enhancer-Enabled Area 0: Off 1: On

(2) LTI (Luminance Transient Improvement)

The enhancer subjects the Y signal input to transient improvement in the horizontal direction.

Transient improvement of the blanking signal is turned off.

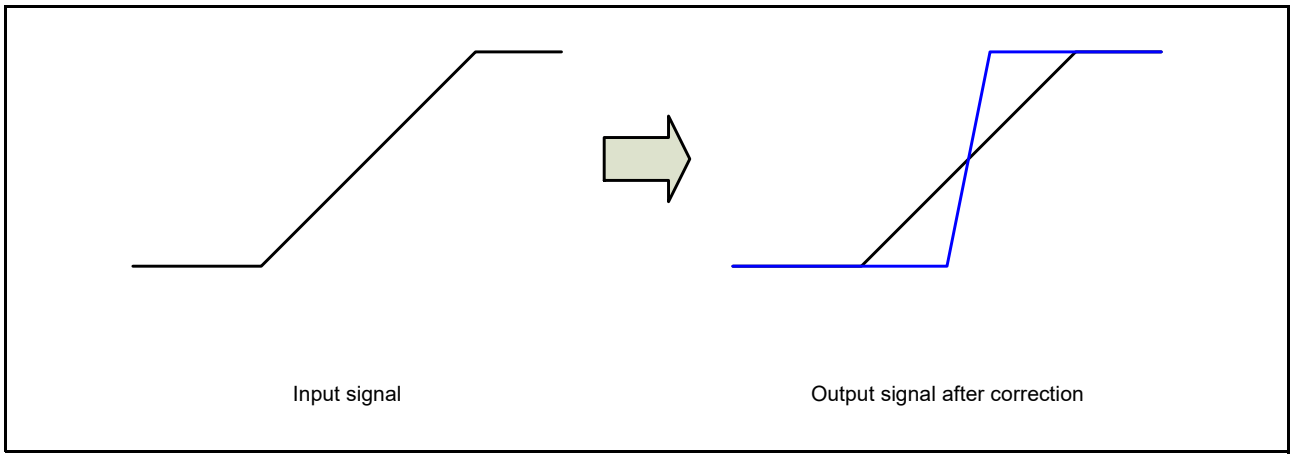


Figure 37.4 LTI Correction

After edge detection of the image, the LTI can be independently controlled in the two horizontal bands.

In LTI, the median filter is inserted after edge detection of the image.

In LTI (H4), the reference pixels of the median filter can be selected.

However, under normal operations, half the tap data (second adjacent pixel) at edge detection is used as reference.

Table 37.4 Reference Pixel Table for LTI

LTI Band	Reference Pixel for Edge Detection	LPF Application	Median Filter Reference Pixels
Horizontal LTI (H2)	Second adjacent pixel used as reference	LPF not applied or LPF (1,2,1)	Adjacent pixel used as reference
Horizontal LTI (H4)	Fourth adjacent pixel used as reference	LPF (1,2,1)	Adjacent pixel or second adjacent pixel used as reference

In LTI, the detection result can be subjected to a coring process.

The core value set in the register is subtracted from the edge detection result, and LTI correction is performed on the coring output after subtraction.

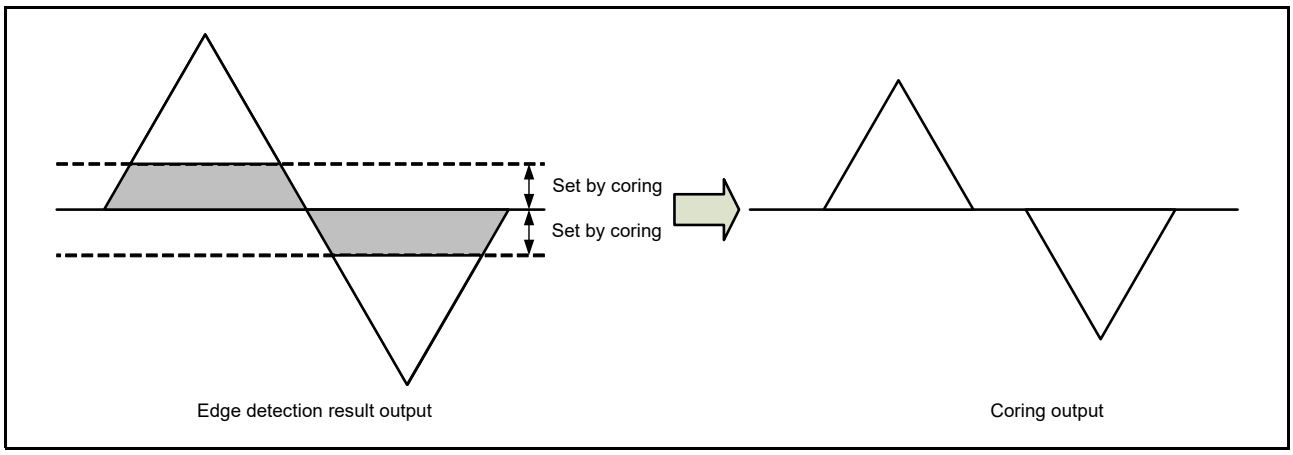


Figure 37.5 LTI Coring

Table 37.5 LTI Control

Register Name	Bit Name	Initial Value	Description
ADJ_ENH_LTI1	LTI_H_ON	0	LTI On/Off Control 0: LTI off 1: LTI on
ADJ_ENH_LTI1	LTI_H2_INC_ZERO [7:0]	10	Median Filter LTI Correction Threshold LTI correction is disabled when: |right TAP value – center TAP value| < LTI1_H2_INC_ZERO or |left TAP value – center TAP value| < LTI1_H2_INC_ZERO.
ADJ_ENH_LTI1	LTI_H2_LPF_SEL	0	LPF Selection for Folding Prevention Before H2 Edge Detection 0: LPF not selected 1: LPF selected
ADJ_ENH_LTI1	LTI_H2_GAIN[7:0]	0	LTI Edge Amplitude Value Gain 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times)
ADJ_ENH_LTI1	LTI_H2_CORE[7:0]	0	LTI Coring (Maximum core value of 255) Amplitude smaller than or equal to the value of LTI_H2_CORE is cored from the edge amplitude value. (A core value setting of 128 remains unchanged.)
ADJ_ENH_LTI2	LTI_H4_INC_ZERO[7:0]	10	Median Filter LTI Correction Threshold LTI correction is disabled when: |right TAP value – center TAP value| < LTI1_H4_INC_ZERO or |left TAP value – center TAP value| < LTI1_H4_INC_ZERO.

Table 37.5 LTI Control

Register Name	Bit Name	Initial Value	Description
ADJ_ENH_LTI2	LTI_H4_MEDIAN_TAP_SEL	0	Median Filter Reference Pixel Select 0: Second adjacent pixel selected as reference 1: Adjacent pixel selected as reference
ADJ_ENH_LTI2	LTI_H4_GAIN[7:0]	0	LTI Edge Amplitude Value Gain 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times)
ADJ_ENH_LTI2	LTI_H4_CORE[7:0]	0	LTI Coring (Maximum core value of 255) Amplitude less than or equal to the value of LTI_H4_CORE is cored from the edge amplitude value. (A core value setting of 128 remains unchanged.)

(3) Sharpness Process

The enhancer performs edge enhancement on the Y signal input by adding overshoot and undershoot to the original signal. Edge enhancement of the blanking signal is turned off.

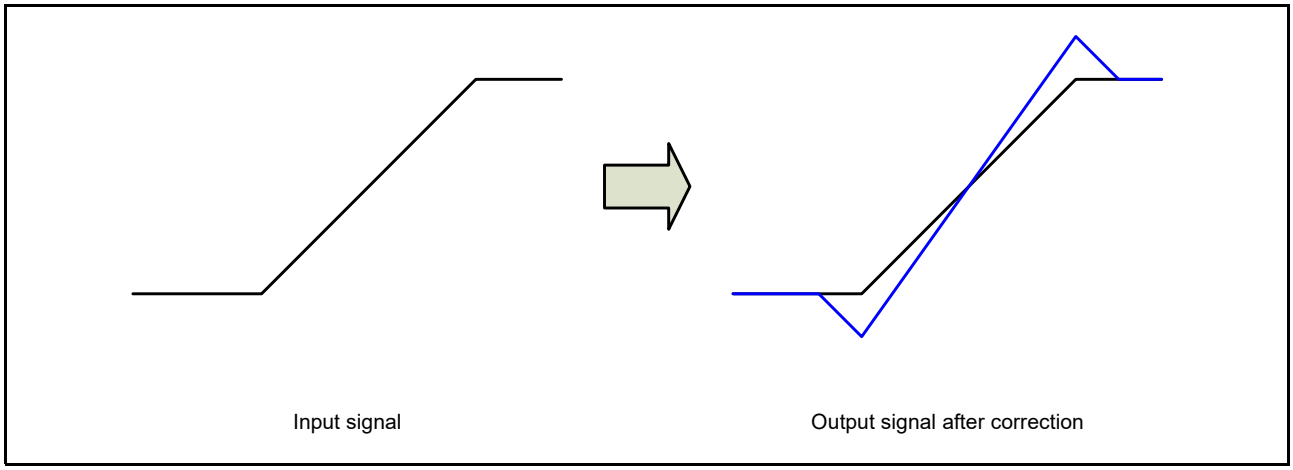


Figure 37.6 Sharpness Correction

After edge detection of the image, the sharpness can be independently controlled in the three horizontal bands.

In horizontal sharpness, a 3-tap low-pass filter (LPF) is inserted before edge detection to prevent folding. The LPF can be turned on or off by register setting.

Table 37.6 Reference Pixel Table for Sharpness

Sharpness Band	Reference Pixel for Edge Detection	LPF Application
Horizontal sharpness (H1)	Adjacent pixel used as reference	LPF not applied
Horizontal sharpness (H2)	Second adjacent pixel used as reference	LPF not applied or LPF (1,2,1)
Horizontal sharpness (H3)	Third adjacent pixel used as reference	LPF (1,2,1)

The edge amplitude of the edge to be enhanced is adjusted according to the value of SHP_CORE.

Edge enhancement is accomplished when the edge detection result of the image is greater than the value of SHP_CORE.

In edge enhancement, a correction value is output by multiplying (edge amplitude value - SHP_CORE) by sharpness gain.

Sharpness is turned off when the edge detection result of the image is smaller than the value of SHP_CORE.

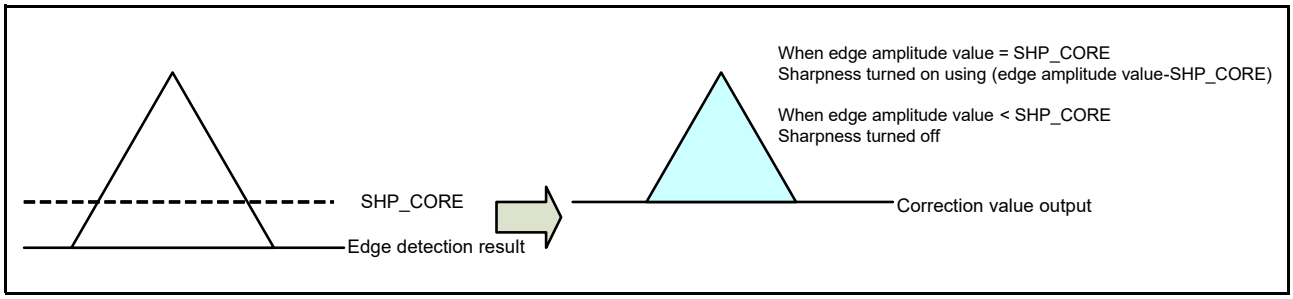


Figure 37.7 Sharpness Characteristics

Table 37.7 Sharpness Control

Register Name	Bit Name	Initial Value	Description
ADJ_ENH_SHP1	SHP_H_ON	0	Sharpness On/Off Control 0: Horizontal sharpness off 1: Horizontal sharpness on
ADJ_ENH_SHP3	SHP_H2_LPF_SEL	0	LPF Selection for Folding Prevention Before H2 Edge Detection 0: LPF not selected 1: LPF selected
ADJ_ENH_SHP2	SHP_H1_CLIP_O[7:0]	0	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H1_CLIP_O
ADJ_ENH_SHP2	SHP_H1_CLIP_U[7:0]	0	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H1_CLIP_U
ADJ_ENH_SHP2	SHP_H1_GAIN_O[7:0]	0	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H1_GAIN_O × (edge amplitude value – SHP_H1_CORE)
ADJ_ENH_SHP2	SHP_H1_GAIN_U[7:0]	0	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H1_GAIN_U × (edge amplitude value – SHP_H1_CORE)
ADJ_ENH_SHP1	SHP_H1_CORE[6:0]	0	Active Sharpness Range Edge amplitude value ≥ SHP_H1_CORE: Sharpness processing on Edge amplitude value < SHP_H1_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.
ADJ_ENH_SHP4	SHP_H2_CLIP_O[7:0]	0	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H2_CLIP_O
ADJ_ENH_SHP4	SHP_H2_CLIP_U[7:0]	0	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H2_CLIP_U
ADJ_ENH_SHP4	SHP_H2_GAIN_O[7:0]	0	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H2_GAIN_O × (edge amplitude value – SHP_H2_CORE)
ADJ_ENH_SHP4	SHP_H2_GAIN_U[7:0]	0	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H2_GAIN_U × (edge amplitude value – SHP_H2_CORE)
ADJ_ENH_SHP3	SHP_H2_CORE[6:0]	0	Active Sharpness Range Edge amplitude value ≥ SHP_H2_CORE: Sharpness processing on Edge amplitude value < SHP_H2_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.
ADJ_ENH_SHP6	SHP_H3_CLIP_O[7:0]	0	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H3_CLIP_O
ADJ_ENH_SHP6	SHP_H3_CLIP_U[7:0]	0	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H3_CLIP_U
ADJ_ENH_SHP6	SHP_H3_GAIN_O[7:0]	0	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H3_GAIN_O × (edge amplitude value – SHP_H3_CORE)
ADJ_ENH_SHP6	SHP_H3_GAIN_U[7:0]	0	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H3_GAIN_U × (edge amplitude value – SHP_H3_CORE)
ADJ_ENH_SHP5	SHP_H3_CORE[6:0]	0	Active Sharpness Range Edge amplitude value ≥ SHP_H3_CORE: Sharpness processing on Edge amplitude value < SHP_H3_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.

37.1.5 Color Matrix

Color matrix is performed by adjusting the offset of each input signal and nine-axis gain. This allows YCbCr to GBR conversion.

(1) GBR to GBR Conversion

YGIN_A = YGIN + ADJ_MTX_YG − 128

CBBIN_A = CBBIN + ADJ_MTX_B − 128

CRRIN_A = CRRIN + ADJ_MTX_R − 128

YGOUT = (ADJ_MTX_GG × YGIN_A + ADJ_MTX_GB × CBBIN_A + ADJ_MTX_GR × CRRIN_A) ÷ 256

CBBOUT = (ADJ_MTX_BG × YGIN_A + ADJ_MTX_BB × CBBIN_A + ADJ_MTX_BR × CRRIN_A) ÷ 256

CRROUT = (ADJ_MTX_RG × YGIN_A + ADJ_MTX_RB × CBBIN_A + ADJ_MTX_RR × CRRIN_A) ÷ 256

(2) YCbCr to GBR Conversion

YGIN_A = YGIN + ADJ_MTX_YG − 128

CBBIN_A = CBBIN − 128

CRRIN_A = CRRIN − 128

YGOUT = (ADJ_MTX_GG × YGIN_A + ADJ_MTX_GB × CBBIN_A + ADJ_MTX_GR × CRRIN_A) ÷ 256

CBBOUT = (ADJ_MTX_BG × YGIN_A + ADJ_MTX_BB × CBBIN_A + ADJ_MTX_BR × CRRIN_A) ÷ 256

CRROUT = (ADJ_MTX_RG × YGIN_A + ADJ_MTX_RB × CBBIN_A + ADJ_MTX_RR × CRRIN_A) ÷ 256

Table 37.8 Matrix Coefficients (Standard Values) of SMPTE 293M

	YGIN		CBBIN		CRRIN	
	Coefficient	Bit Setting	Coefficient	Bit Setting	Coefficient	Bit Setting
YGOUT	1.000	ADJ_MTX_GG = 256	–0.344	ADJ_MTX_GB = 1960	–0.714	ADJ_MTX_GR = 1865
CBBOUT	1.000	ADJ_MTX_BG = 256	1.772	ADJ_MTX_BB = 454	0.000	ADJ_MTX_BR = 0
CRROUT	1.000	ADJ_MTX_RG = 256	0.000	ADJ_MTX_RB = 0	1.402	ADJ_MTX_RR = 359

Table 37.9 Color Matrix Control

Register Name	Bit Name	Initial Value	Description
ADJ_MTX_MODE	ADJ_MTX_MD[1:0]	2	Operating Mode 0: GBR → GBR 1: Setting prohibited 2: YCbCr → GBR 3: Setting prohibited
ADJ_MTX_YG_ADJ0	ADJ_MTX_YG[7:0]	128	Y/G Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128(0) to 255 (+127) [LSB])
ADJ_MTX_CBB_ADJ0	ADJ_MTX_B[7:0]	128	B Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
ADJ_MTX_CRR_ADJ0	ADJ_MTX_R[7:0]	128	R Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
ADJ_MTX_YG_ADJ0	ADJ_MTX_GG[10:0]	256	Gain Adjustment of Y/G Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_YG_ADJ1	ADJ_MTX_GB[10:0]	1960	Gain Adjustment of Cb/B Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_YG_ADJ1	ADJ_MTX_GR[10:0]	1865	Gain Adjustment of Cr/R Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_CBB_ADJ0	ADJ_MTX_BG[10:0]	256	Gain Adjustment of Y/G Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_CBB_ADJ1	ADJ_MTX_BB[10:0]	454	Gain adjustment of Cb/B signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_CBB_ADJ1	ADJ_MTX_BR[10:0]	0	Gain Adjustment of Cr/R Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_CRR_ADJ0	ADJ_MTX_RG[10:0]	256	Gain Adjustment of Y/G Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_CRR_ADJ1	ADJ_MTX_RB[10:0]	0	Gain Adjustment of Cb/B Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_CRR_ADJ1	ADJ_MTX_RR[10:0]	359	Gain Adjustment of Cr/R Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

37.2 Register Description

Table 37.10 shows the register configuration.

[Symbols used in Register Description]

Initial value: Register value after a reset
—: Undefined value
R/W: Readable/writable. The written value can be read.
R/WC0: Read and write. Bit is initialized if 0 is written, and ignored if 1 is written.
R/WC1: Read and write. Bit is initialized if 1 is written, and ignored if 0 is written.
R: Read-only. The write value should always be 0.
—/W: Write-only. Read value is undefined.

Table 37.10 Image Quality Improver Register Configuration

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Register update control register in image quality improver (image quality improver 0)	ADJ0_UPDATE	R/WC1	H'0000 0000	H'FCFF 7680	32
Black stretch register (image quality improver 0)	ADJ0_BKSTR_SET	R/W	H'0000 0000	H'FCFF 7684	32
Enhancer timing adjustment register 1 (image quality improver 0)	ADJ0_ENH_TIM1	R/W	H'0000 0010	H'FCFF 7688	32
Enhancer timing adjustment register 2 (image quality improver 0)	ADJ0_ENH_TIM2	R/W	H'0023 01E0	H'FCFF 768C	32
Enhancer timing adjustment register 3 (image quality improver 0)	ADJ0_ENH_TIM3	R/W	H'0091 0280	H'FCFF 7690	32
Enhancer sharpness register 1 (image quality improver 0)	ADJ0_ENH_SHP1	R/W	H'0000 0000	H'FCFF 7694	32
Enhancer sharpness register 2 (image quality improver 0)	ADJ0_ENH_SHP2	R/W	H'0000 0000	H'FCFF 7698	32
Enhancer sharpness register 3 (image quality improver 0)	ADJ0_ENH_SHP3	R/W	H'0000 0000	H'FCFF 769C	32
Enhancer sharpness register 4 (image quality improver 0)	ADJ0_ENH_SHP4	R/W	H'0000 0000	H'FCFF 76A0	32
Enhancer sharpness register 5 (image quality improver 0)	ADJ0_ENH_SHP5	R/W	H'0000 0000	H'FCFF 76A4	32
Enhancer sharpness register 6 (image quality improver 0)	ADJ0_ENH_SHP6	R/W	H'0000 0000	H'FCFF 76A8	32
Enhancer LTI register 1 (image quality improver 0)	ADJ0_ENH_LTI1	R/W	H'000A 0000	H'FCFF 76AC	32
Enhancer LTI register 2 (image quality improver 0)	ADJ0_ENH_LTI2	R/W	H'000A 0000	H'FCFF 76B0	32
Matrix mode register in image quality improver (image quality improver 0)	ADJ0_MTX_MODE	R/W	H'0000 0002	H'FCFF 76B4	32
Matrix YG control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_YG_ADJ0	R/W	H'0080 0100	H'FCFF 76B8	32
Matrix YG control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_YG_ADJ1	R/W	H'07A8 0749	H'FCFF 76BC	32
Matrix CBB control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_CBB_ADJ0	R/W	H'0080 0100	H'FCFF 76C0	32
Matrix CBB control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_CBB_ADJ1	R/W	H'01C6 0000	H'FCFF 76C4	32

Table 37.10 Image Quality Improver Register Configuration

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Matrix CRR control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_CRR_ADJ0	R/W	H'0080 0100	H'FCFF 76C8	32
Matrix CRR control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_CRR_ADJ1	R/W	H'0000 0167	H'FCFF 76CC	32

37.2.1 Register Update Control Register in Image Quality Improver (ADJ0_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADJ0_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ADJ0_VEN	0	R/WC1	Image Quality Improver Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

37.2.2 Black Stretch Register (ADJ0_BKSTR_SET)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	BKSTR_ON	BKSTR_ST[3:0]				BKSTR_D[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	BKSTR_T1[4:0]					—	—	—	BKSTR_T2[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	BKSTR_ON	0	R/W	Black Stretch On/Off Control 0: Black stretch off 1: Black stretch on
23 to 20	BKSTR_ST [3:0]	0	R/W	Black Stretch Start Point Setting values: 0 (low) to 15 (high)
19 to 16	BKSTR_D [3:0]	0	R/W	Depth of Black Stretch Setting Values: 0 (shallow) to 15 (deep)
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	BKSTR_T1 [4:0]	0	R/W	Black Stretch Time Constant (T1) Setting Values: 0 (small) to 31 (large)
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	BKSTR_T2 [4:0]	0	R/W	Black Stretch Time Constant (T2) Setting Values: 0 (small) to 31 (large)

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.3 Enhancer Timing Adjustment Register 1 (ADJ0_ENH_TIM1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	ENH_MD	—	—	—	ENH_DISP_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	ENH_MD	1	R/W	Operating Mode 0: RGB mode 1: YCbCr mode
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ENH_DISP_ON	0	R/W	Frame Line Display On/Off of Enhancer-Enabled Area 0: Display off 1: Display on

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.4 Enhancer Timing Adjustment Register 2 (ADJ0_ENH_TIM2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ENH_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ENH_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ENH_VS[10:0]	35	R/W	Start Position of Vertical Valid Image Area in Enhancer-Enabled Area Note: Set to 2 or greater lines.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ENH_VW[10:0]	480	R/W	Width of Vertical Valid Image Area in Enhancer-Enabled Area

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.5 Enhancer Timing Adjustment Register 3 (ADJ0_ENH_TIM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ENH_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ENH_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ENH_HS[10:0]	145	R/W	Start Position of Horizontal Valid Image Area in Enhancer-Enabled Area Note: Set to 4 or greater clocks.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ENH_HW[10:0]	640	R/W	Width of Horizontal Valid Image Area in Enhancer-Enabled Area

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.6 Enhancer Sharpness Register 1 (ADJ0_ENH_SHP1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SHP_H_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SHP_H1_CORE[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SHP_H_ON	0	R/W	Sharpness On/Off Control 0: Horizontal sharpness off 1: Horizontal sharpness on
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	SHP_H1_CORE[6:0]	0	R/W	Active Sharpness Range Edge amplitude value ≥ SHP_H1_CORE: Sharpness processing on Edge amplitude value < SHP_H1_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.7 Enhancer Sharpness Register 2 (ADJ0_ENH_SHP2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SHP_H1_CLIP_O[7:0]								SHP_H1_CLIP_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHP_H1_GAIN_O[7:0]								SHP_H1_GAIN_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	SHP_H1_CLIP_O [7:0]	0	R/W	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H1_CLIP_O
23 to 16	SHP_H1_CLIP_U [7:0]	0	R/W	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H1_CLIP_U
15 to 8	SHP_H1_GAIN_O [7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+approx. 4 times) Sharpness correction value = SHP_H1_GAIN_O × (edge amplitude value – SHP_H1_CORE)
7 to 0	SHP_H1_GAIN_U [7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+approx. 4 times) Sharpness correction value = SHP_H1_GAIN_U × (Edge amplitude value – SHP_H1_CORE)

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.8 Enhancer Sharpness Register 3 (ADJ0_ENH_SHP3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SHP_H2_LPF_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SHP_H2_CORE[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SHP_H2_ LPF_SEL	0	R/W	LPF Selection for Folding Prevention before H2 Edge Detection 0: LPF not selected 1: LPF selected
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	SHP_H2_ CORE[6:0]	0	R/W	Active Sharpness Range Edge amplitude value ≥ SHP_H2_CORE: Sharpness processing on Edge amplitude value < SHP_H2_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.9 Enhancer Sharpness Register 4 (ADJ0_ENH_SHP4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SHP_H2_CLIP_O[7:0]								SHP_H2_CLIP_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHP_H2_GAIN_O[7:0]								SHP_H2_GAIN_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	SHP_H2_ CLIP_O[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H2_CLIP_O
23 to 16	SHP_H2_ CLIP_U[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H2_CLIP_U
15 to 8	SHP_H2_ GAIN_O[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H2_GAIN_O × (edge amplitude value – SHP_H2_CORE)
7 to 0	SHP_H2_ GAIN_U[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H2_GAIN_U × (edge amplitude value – SHP_H2_CORE)

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.10 Enhancer Sharpness Register 5 (ADJ0_ENH_SHP5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SHP_H3_CORE[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	SHP_H3_CORE[6:0]	0	R/W	Active Sharpness Range Edge amplitude value ≥ SHP_H3_CORE: Sharpness processing on Edge amplitude value < SHP_H3_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.11 Enhancer Sharpness Register 6 (ADJ0_ENH_SHP6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SHP_H3_CLIP_O[7:0]								SHP_H3_CLIP_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHP_H3_GAIN_O[7:0]								SHP_H3_GAIN_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	SHP_H3_CLIP_O[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H3_CLIP_O
23 to 16	SHP_H3_CLIP_U[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H3_CLIP_U
15 to 8	SHP_H3_GAIN_O[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H3_GAIN_O × (Edge amplitude value – SHP_H3_CORE)
7 to 0	SHP_H3_GAIN_U[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H3_GAIN_U × (Edge amplitude value – SHP_H3_CORE)

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.12 Enhancer LTI Register 1 (ADJ0_ENH_LTI1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LTI_H_ON	—	—	—	—	—	—	LTI_H2_LPF_SEL	LTI_H2_INC_ZERO[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W:	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTI_H2_GAIN[7:0]								LTI_H2_CORE[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	LTI_H_ON	0	R/W	LTI On/Off Control 0: LTI off 1: LTI on
30 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	LTI_H2_LPF_SEL	0	R/W	LPF Selection for Folding Prevention before H2 Edge Detection 0: LPF not selected 1: LPF selected
23 to 16	LTI_H2_INC_ZERO[7:0]	10	R/W	Median Filter LTI Correction Threshold LTI correction is disabled when | right TAP value – center TAP value | < LTI_H2_INC_ZERO or | left TAP value – center TAP value | < LTI_H2_INC_ZERO
15 to 8	LTI_H2_GAIN[7:0]	0	R/W	LTI Edge Amplitude Value Gain 0 (0 times) to 64 (+ 1 times) to 255 (+ approx. 4 times)
7 to 0	LTI_H2_CORE[7:0]	0	R/W	LTI Coring (Maximum Core value of 255) Amplitude less than or equal to the value of LTI_H2_CORE is cored from the edge amplitude value. (A core value setting of 128 remains unchanged)

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.13 Enhancer LTI Register 2 (ADJ0_ENH_LTI2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	LTI_H4_MEDIAN_TAP_SEL	LTI_H4_INC_ZERO[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTI_H4_GAIN[7:0]								LTI_H4_CORE[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	LTI_H4_MEDIAN_TAP_SEL	0	R/W	Median Filter Reference Pixel Select 0: Second adjacent pixel selected as reference 1: Adjacent pixel selected as reference
23 to 16	LTI_H4_INC_ZERO[7:0]	10	R/W	Median Filter LTI Correction Threshold LTI correction is disabled when | right TAP value – center TAP value | < LTI_H4_INC_ZERO or | left TAP value – center TAP value | < LTI_H4_INC_ZERO
15 to 8	LTI_H4_GAIN[7:0]	0	R/W	LTI Edge Amplitude Value Gain 0 (0 times) to 64 (+ 1 times) to 255 (+ approx. 4 times)
7 to 0	LTI_H4_CORE[7:0]	0	R/W	LTI Coring (Maximum Core value of 255) Amplitude less than or equal to the value of LTI_H4_CORE is cored from the edge amplitude value (A core value setting of 128 remains unchanged)

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.14 Matrix Mode Register in Image Quality Improver (ADJ0_MTX_MODE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADJ0_MTX_MD[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	ADJ0_MTX_MD[1:0]	2	R/W	Operating Mode 0: GBR → GBR 1: Setting prohibited 2: YCbCr → GBR 3: Setting prohibited

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.15 Matrix YG Control Register 0 in Image Quality Improver (ADJ0_MTX_YG_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADJ0_MTX_YG[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ0_MTX_GG[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	ADJ0_MTX_YG[7:0]	128	R/W	Y/G Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ0_MTX_GG[10:0]	256	R/W	Gain Adjustment of Y/G Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.16 Matrix YG Control Register 1 in Image Quality Improver (ADJ0_MTX_YG_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ADJ0_MTX_GB[10:0]										
Initial value:	0	0	0	0	0	1	1	1	1	0	1	0	1	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ0_MTX_GR[10:0]										
Initial value:	0	0	0	0	0	1	1	1	0	1	0	0	1	0	0	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ADJ0_MTX_GB [10:0]	1960	R/W	Gain Adjustment of Cb/B Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ0_MTX_GR [10:0]	1865	R/W	Gain Adjustment of Cr/R Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.17 Matrix CBB Control Register 0 in Image Quality Improver (ADJ0_MTX_CBB_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADJ0_MTX_B[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ0_MTX_BG[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	ADJ0_MTX_B [7:0]	128	R/W	B Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ0_MTX_BG [10:0]	256	R/W	Gain Adjustment of Y/G Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.18 Matrix CBB Control Register 1 in Image Quality Improver (ADJ0_MTX_CBB_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ADJ0_MTX_BB[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ0_MTX_BR[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ADJ0_MTX_BB [10:0]	454	R/W	Gain Adjustment of Cb/B Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ0_MTX_BR [10:0]	0	R/W	Gain Adjustment of Cr/R Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.19 Matrix CRR Control Register 0 in Image Quality Improver (ADJ0_MTX_CRR_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADJ0_MTX_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ0_MTX_RG[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	ADJ0_MTX_R [7:0]	128	R/W	R Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ0_MTX_RG [10:0]	256	R/W	Gain Adjustment of Y/G Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.2.20 Matrix CRR Control Register 1 in Image Quality Improver (ADJ0_MTX_CRR_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ADJ0_MTX_RB[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ0_MTX_RR[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	1	1	0	0	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ADJ0_MTX_RB [10:0]	0	R/W	Gain Adjustment of Cb/B Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ0_MTX_RR [10:0]	359	R/W	Gain Adjustment of Cr/R Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ0_VEN in ADJ0_UPDATE is 1.

37.3 Usage Method

37.3.1 Black Stretch Usage Method

The degree of black stretch can be adjusted by setting the depth (BKSTR_D[3:0]) and the start point (BKSTR_ST[3:0]) of the black stretch. The variation in the black stretch time axis can be adjusted by setting the time constant (BKSTR_T1[4:0] and BKSTR_T2[4:0]). By setting the time constant, changes that occur abruptly due to swapping of the scene can be controlled.

Table 37.11 Black Stretch Setting Register

Register Name	Bit Name	Set Value
ADJ_BKSTR_SET	BKSTR_ON	When black stretch is on: 1
ADJ_BKSTR_SET	BKSTR_D[3:0]	Set the depth of black stretch. The depth increases as the value becomes larger.
ADJ_BKSTR_SET	BKSTR_ST[3:0]	Set the start point of black stretch. The stretching area becomes larger as the value becomes larger.
ADJ_BKSTR_SET	BKSTR_T1[4:0]	Set the time constant of black stretch in the positive direction. The changes are more delayed as the value becomes larger.
ADJ_BKSTR_SET	BKSTR_T2[4:0]	Set the time constant of black stretch in the negative direction. The changes are more delayed as the value becomes larger.

Note: ADJ_VEN in ADJ_UPDATE should be set to 1 after setting the registers.

37.3.2 LTI Processing of Enhancer

Figure 37.8 shows an example of LTI adjustment.

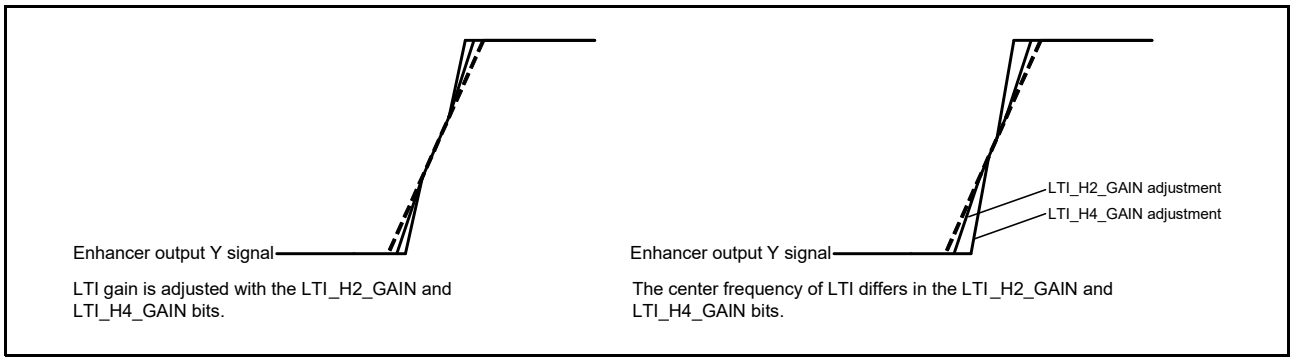


Figure 37.8 Example of LTI Adjustment

37.3.3 Sharpness Processing of Enhancer

Figure 37.9 shows an example of sharpness adjustment.

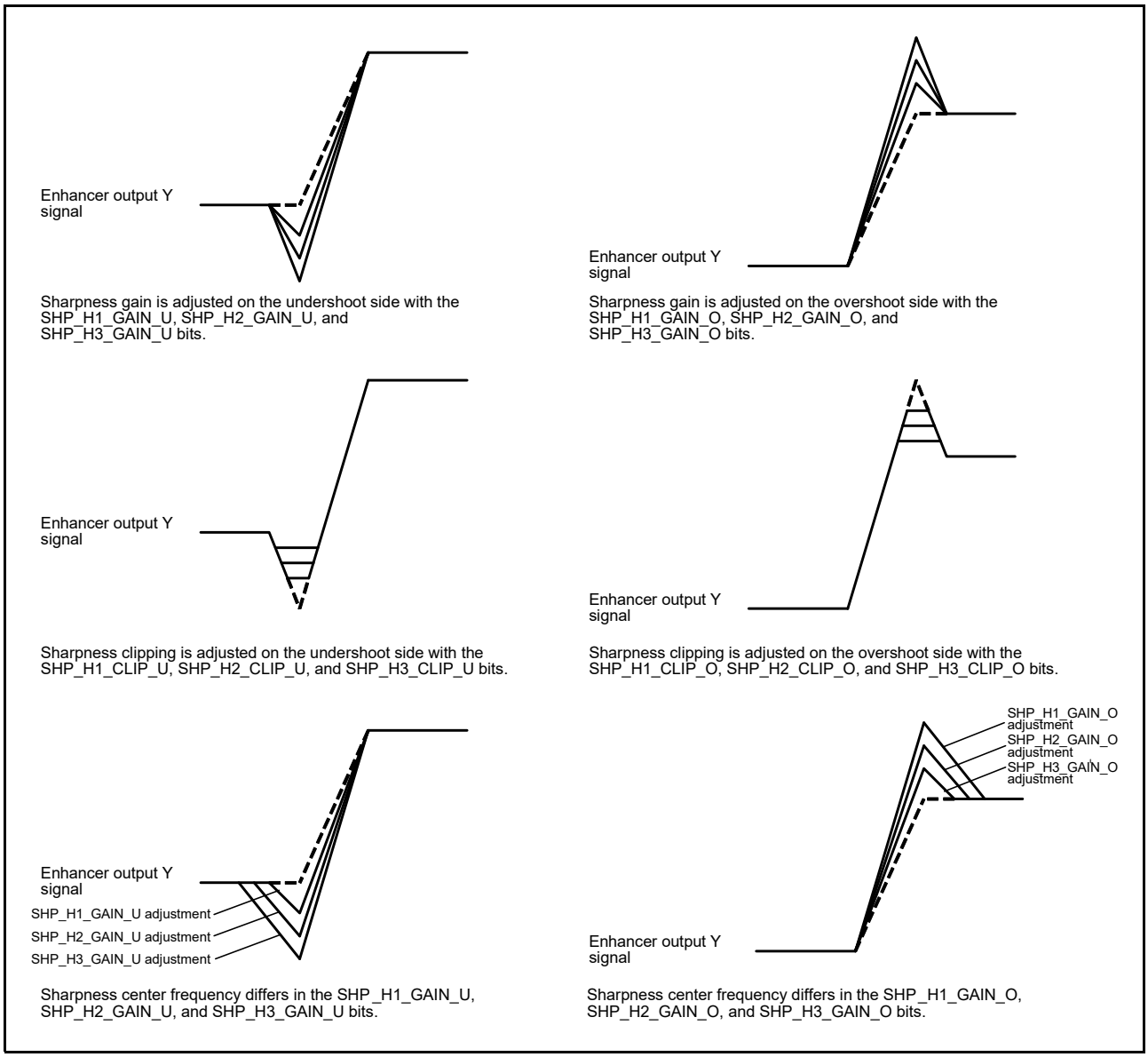


Figure 37.9 Example of Sharpness Adjustment

37.3.4 Setting Method for Color Matrix Data Conversion

GBR signals are assumed to be input to the circuit subsequent to the image quality improver; therefore, the output from the color matrix circuit should be in the GBR format.

Table 37.12 shows an example of GBR conversion setting.

Table 37.12 Recommended Setting Values for Matrix Conversion

		GBR to GBR Conversion	YCBCR to GBR Conversion
Register Name	Bit Name	Recommended Values	Recommended Values
ADJ_MTX_MODE	ADJ_MTX_MD[1:0]	0	2
ADJ_MTX_YG_ADJ0	ADJ_MTX_YG[7:0]	128	128
ADJ_MTX_CBB_ADJ0	ADJ_MTX_B[7:0]	128	128
ADJ_MTX_CRR_ADJ0	ADJ_MTX_R[7:0]	128	128
ADJ_MTX_YG_ADJ0	ADJ_MTX_GG[10:0]	256	256
ADJ_MTX_YG_ADJ1	ADJ_MTX_GB[10:0]	0	1960
ADJ_MTX_YG_ADJ1	ADJ_MTX_GR[10:0]	0	1865
ADJ_MTX_CBB_ADJ0	ADJ_MTX_BG[10:0]	0	256
ADJ_MTX_CBB_ADJ1	ADJ_MTX_BB[10:0]	256	454
ADJ_MTX_CBB_ADJ1	ADJ_MTX_BR[10:0]	0	0
ADJ_MTX_CRR_ADJ0	ADJ_MTX_RG[10:0]	0	256
ADJ_MTX_CRR_ADJ1	ADJ_MTX_RB[10:0]	0	0
ADJ_MTX_CRR_ADJ1	ADJ_MTX_RR[10:0]	256	359

Note: ADJ_VEN in ADJ_UPDATE should be set to 1 after setting the registers.

38. Video Display Controller 6 (5): Image Synthesizer

38.1 Image Synthesizer

38.1.1 Overview of Functions

The image synthesizer reads graphics data from the frame buffer and displays the synthesized image on the screen.

One video plane + two graphics planes, or three graphics planes can be selected for synthesis.

RGB565, RGB888, αRGB1555, αRGB4444, αRGB8888, RGBα5551, RGBα8888, CLUT8, CLUT4, CLUT1, YCbCr422 (for the graphics 0 process), and YCbCr444 (for the graphics 0 process) formats can be used for graphics data, and RGB565, RGB888, YCbCr422, and YCbCr444 formats for video data.

On each of the graphics planes, background color, lower-layer graphics, current graphics, or blended image (for the graphics 2 and 3 processes) of lower-layer graphics and current graphics can be displayed.

It is recommend that the frame buffer is placed in the on-chip large-capacity RAM. Both the on-chip large-capacity RAM and an external SDRAM can be used for the frame buffer, but the bus bandwidth might become short and display might not be possible if an external SDRAM is used for the frame buffer.

The functional block diagram of the image synthesizer is shown below.

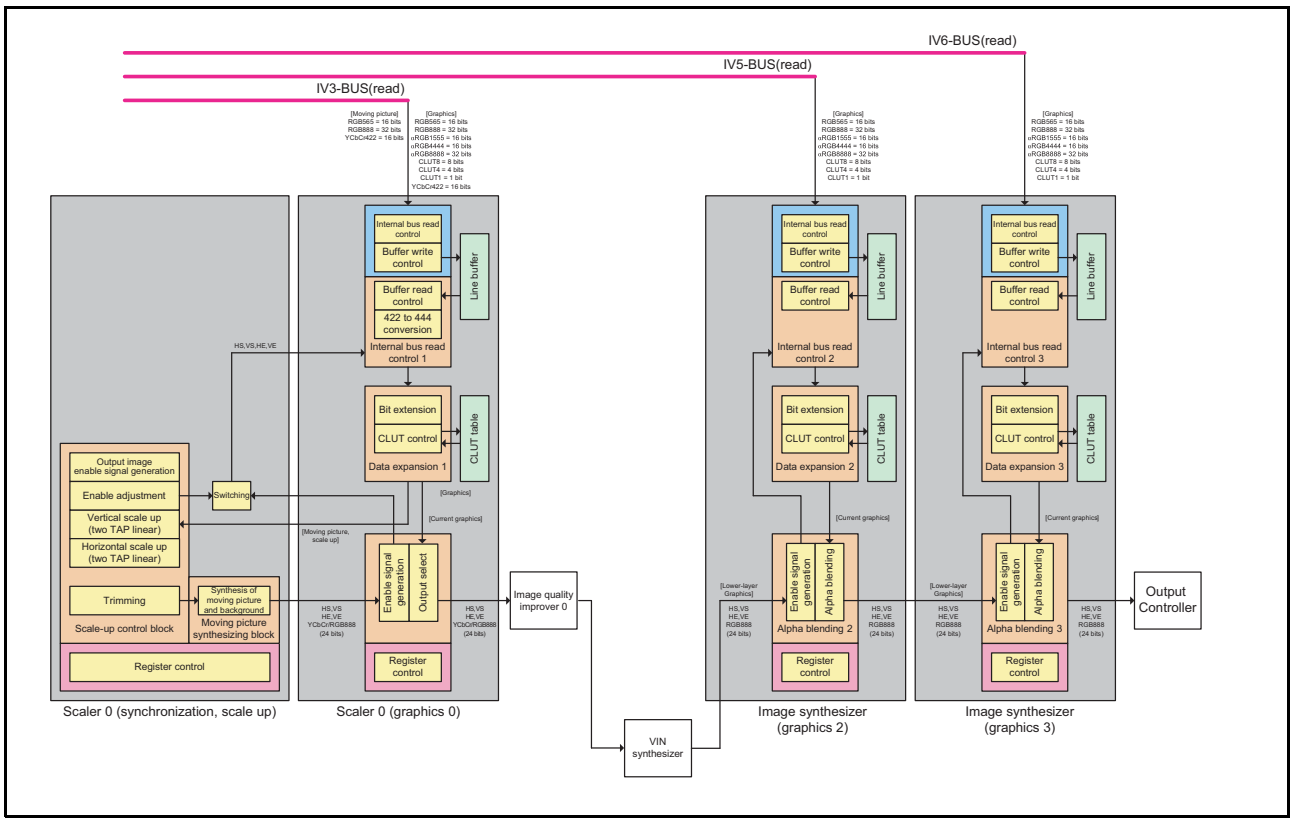


Figure 38.1 Functional Block Diagram of Image Synthesizer

38.1.2 Graphics Data Read Control

Graphics data read can be controlled for the three processes: the graphics 0 process in the scaler 0, and the graphics 2 and 3 processes in the image synthesizer.

The register bits of each process can be identified by the number in the register name like GR0_xxxx, GR2_xxxx, and GR3_xxxx, respectively. In the sections except for Register Descriptions, however, the number is omitted like GR_xxxx for convenience sake.

In the VIN synthesizer, graphics data read is not controlled. In this manual, the name is omitted like GR_xxxx for convenience sake. The synthesizer does not have the read control register (GR_FLM).

(1) Updating Registers

The Vsync signal is used to control the update timing of the registers for graphics display and frame buffer read control, except for some of the registers.

After 1 is set to the bits in the update control register, the contents of the relevant registers are actually modified at the rising edge of the Vsync signal, when the update control register is automatically cleared to 0.

Table 38.1 Register Update Control

Register Name	Bit Name	Initial Value	Description
GR_UPDATE	GR_UPDATE	0	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated.
GR_UPDATE	GR_P_VEN	0	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GR_UPDATE	GR_IBUS_VEN	0	Frame Buffer Read Register Update * 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

Note: * This bit is not supported for the VIN synthesizer.

(2) Frame Buffer Burst Transfer Mode

Either 32-byte or 128-byte transfer mode can be selected for accessing the frame buffer in which video data and graphics data are stored.

Table 38.2 Frame Buffer Burst Transfer Mode

Register Name	Bit Name	Initial Value	Description
GR_FLM1	GR_BST_MD	0	Frame Buffer Burst Transfer Mode 0: 32-byte transfer 1: 128- byte transfer

(3) Frame Buffer Control Mode

More than one frame of data is read from the frame buffer.

For graphics data, set the GR_FLM_SEL[1:0] bits to 1, and set the specific display frame number with the GR_FLM_NUM[9:0] bits. For video data, select a mode with the GR_FLM_SEL[1:0] bits depending on the writing process used; the quantity of the frames used for video data is set in the writing process block.

Table 38.3 Frame Buffer Control Mode

Register Name	Bit Name	Initial Value	Description
GR_FLM1	GR_FLM_SEL[1:0]	0	Frame Buffer Address Setting Signal Select 0: Control linked to scaling-down process, or frame 0 selected.*1 1: Register GR_FLM_NUM selected. 2: Control linked to distortion correction, or frame 0 selected.*2 3: Control linked to pointer buffer, or setting prohibited.*3
GR_FLM3	GR_FLM_NUM[9:0]	0	Frame Number of Frame Buffer Manually set the frame number when GR_FLM_SEL = 1.

Note 1. For the graphics 0 process, frame buffer control links to the scaling-down process. For the graphics 2 and 3 processes, frame 0 is selected.
Note 2. For the graphics 0 process, frame buffer control links to distortion correction. For the graphics 2 and 3 processes, frame 0 is selected.
Note 3. For the graphics 0 process, frame buffer control links to the pointer buffer. For the graphics 2 and 3 processes, setting is prohibited.

(4) Frame Buffer Read Control

The following bit enables or disables read access to the frame buffer.

Table 38.4 Frame Buffer Read Control

Register Name	Bit Name	Initial Value	Description
GR_FLM_RD	GR_R_ENB	0	Frame Buffer Read Enable 0: Disables read access to the frame buffer. 1: Enables read access to the frame buffer.

(5) Distortion Correction Frame Buffer Control

Two frames (frames 0 and 1) are used for distortion correction, and the frame numbers to be read by the image renderer are set.
The frame numbers to be read (frames 0 and 1) can be switched by setting the GR_IMR_FLM_INV bit.
This bit is enabled only when the GR_FLM_SEL bits are set to 2.

Table 38.5 Frame Buffer Size

Register Name	Bit Name	Initial Value	Description
GR1_FLM1	GR1_IMR_FLM_INV	0	Sets the frame buffer number for distortion correction.* 0:Does not switch the frame numbers to be read. 1:Switches the frame numbers to be read.

Note: * This function is supported for the graphics 0 process only.

(6) Frame Buffer Size

The following bits set the size of the frame buffer to be read.

The numbers of horizontal pixels and of lines in the vertical direction are set with the GR_HW[10:0] and GR_FLM_LNUM[10:0] bits, respectively.

Table 38.6 Frame Buffer Size

Register Name	Bit Name	Initial Value	Description
GR_FLM6	GR_HW[10:0]	0	Sets the width of the horizontal valid period. The width is (GR_HW + 1) pixels. Note: Set to 2 or greater.
GR_FLM5	GR_FLM_LNUM[10:0]	0	Sets the number of lines in a frame The number of lines is (GR_FLM_LNUM + 1).

(7) Calculating Addresses in Frame Buffer

The data area in the frame buffer is defined using the addresses specified by GR_BASE[31:0], GR_LN_OFF[14:0], and GR_FLM_OFF[22:0] bits and the display frame number.

The GR_LN_OFF[14:0] and GR_FLM_OFF[22:0] bits should be set in units of 32/128 bytes (the lower 5/7 bits should be fixed to 0).

The GR_BASE[31:0] bits should be set in units of 64 bits to set the display data start position (the lower three bits should be fixed).

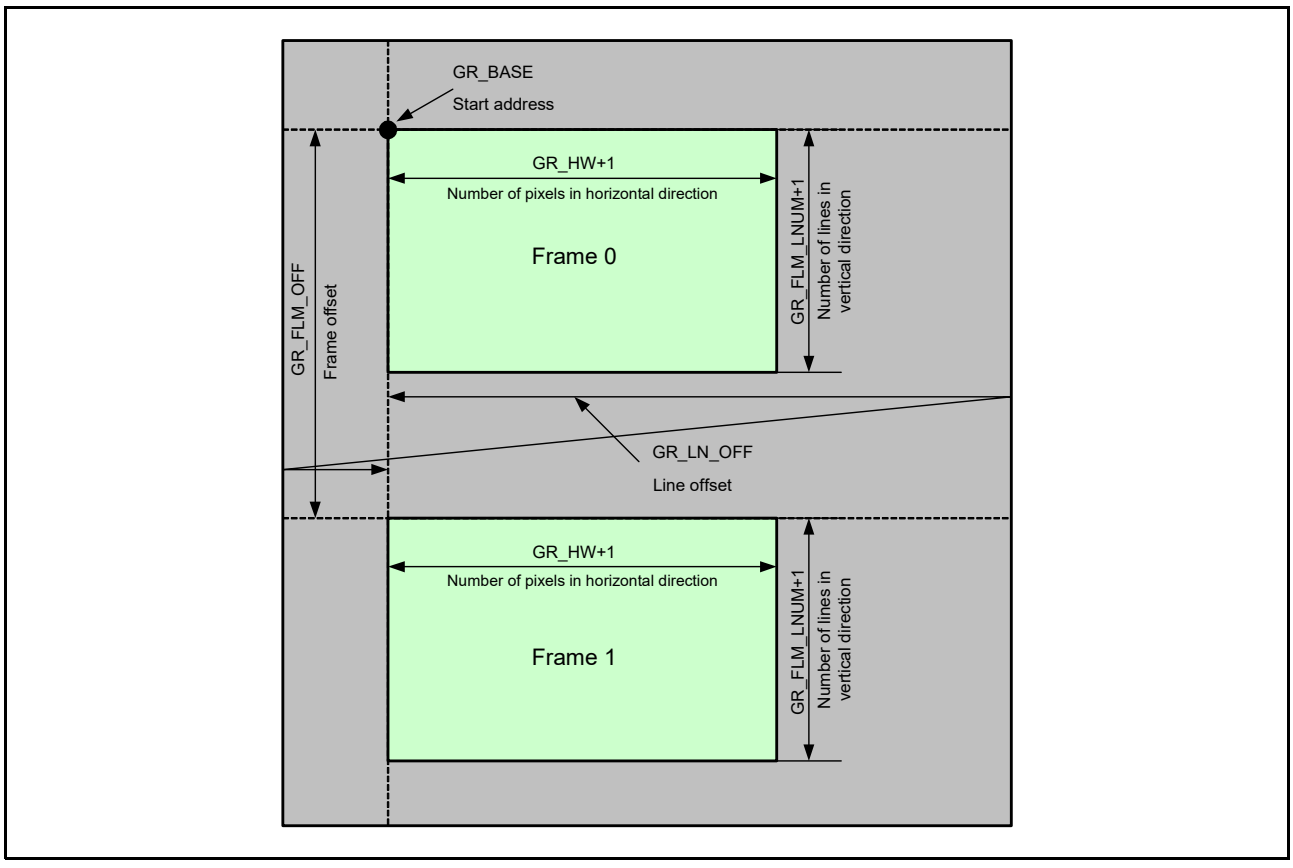


Figure 38.2 Data Arrangement in Frame Buffer

Table 38.7 Calculation of Addresses in Frame Buffer

Register Name	Bit Name	Initial Value	Description
GR_FLM2	GR_BASE[31:0]	0	Frame Buffer Base Address Sets the start address of the frame buffer where frame data is to be stored. GR_BASE[4:3] and GR_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower 3 bits should be set to 000.
GR_FLM3	GR_LN_OFF[14:0]	0	Frame Buffer Line Offset Address Sets the line offset address for calculating the start address of each line. Line 0:GR_BASE Line 1:GR_BASE + GR_LN_OFF × 1 : Line n: GR_BASE + GR_LN_OFF × n For 32-byte transfer, the lower 5 bits should be fixed to 0_0000. For 128-byte transfer, the lower 7 bits should be fixed to 000_0000.
GR_FLM4	GR_FLM_OFF[22:0]	0	Frame Buffer Frame Offset Address (lower) Specifies the frame offset address used for calculating the start address of each frame buffer when more than one buffer is used. Buffer 0: GR_BASE Buffer 1: GR_BASE + GR_FLM_OFF × 1 : Buffer n: GR_BASE + GR_FLM_OFF × n For 32-byte transfer, the lower 5 bits should be fixed to 0_0000. For 128-byte transfer, the lower 7 bits should be fixed to 000_0000.

(8) Setting Frame Buffer Size Smaller than One Frame

Frame buffer size can be set in one-line units.

When the number of lines set with the GR_FLM_LOOP[10:0] bits is smaller than the value of the GR_FLM_LNUM[10:0] bits, data is again read from the start address of the frame buffer after the number of lines set with the (GR_FLM_LOOP[10:0] + 1) bits have been read.

Table 38.8 Setting of Frame Buffer Size Smaller than One Frame

Register Name	Bit Name	Initial Value	Description
GR_FLM5	GR_FLM_LOOP[10:0]	1023	Number of lines when reading the addresses repeatedly by returning to the start address after reaching the end address. The number of lines is (GR_FLM_LOOP + 1).

(9) Line Offset Control for Frame Buffer

The following bit sets the line offset address direction of the frame buffer.

Table 38.9 Line Offset Address Direction Control for Frame Buffer

Register Name	Bit Name	Initial Value	Description
GR_FLM1	GR_LN_OFF_DIR	0	Selects the line offset address direction of the frame buffer. 0:Increments the address by the line offset address. 1:Decrements the address by the line offset address.

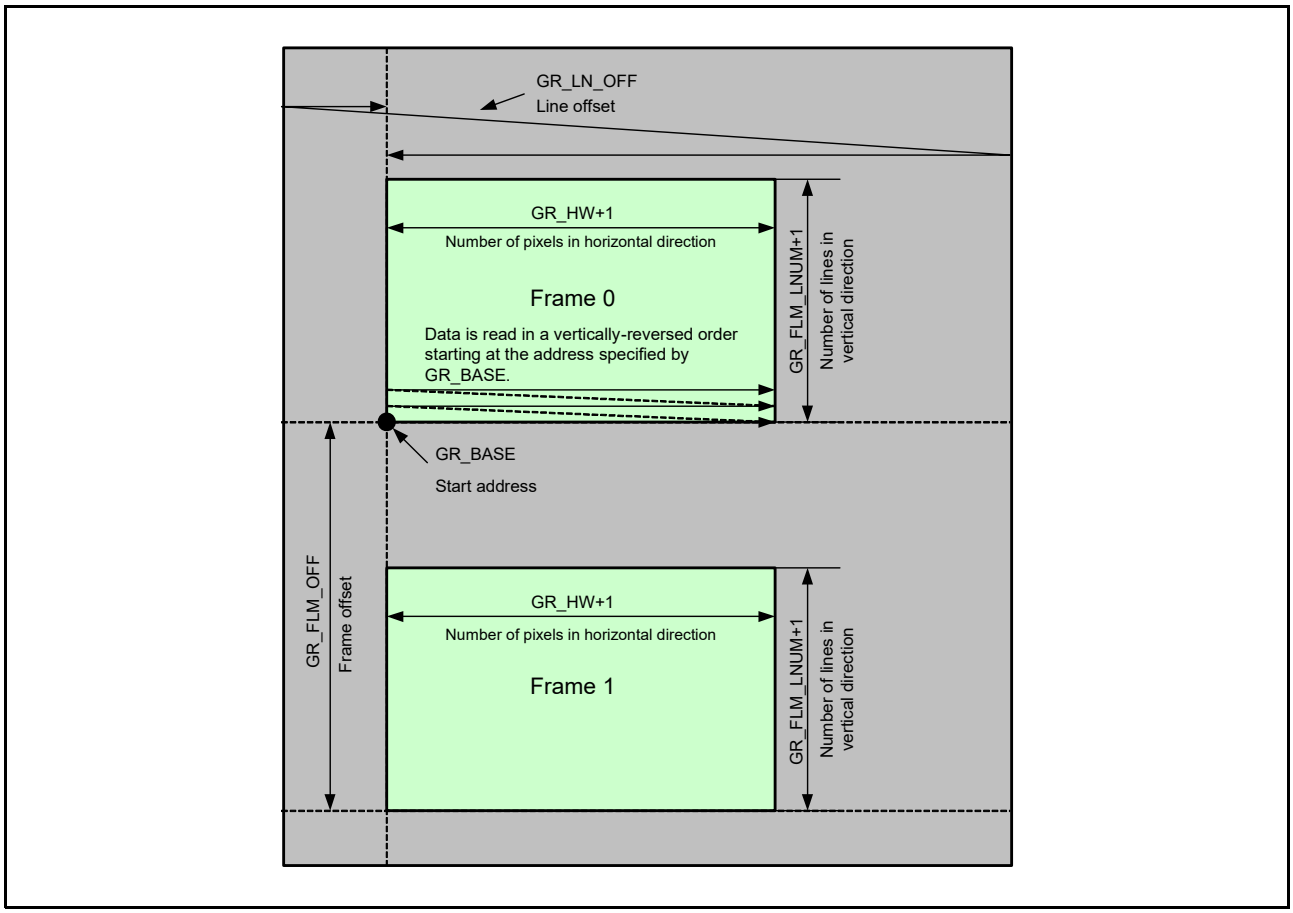


Figure 38.3 Data Arrangement with Line Offset and Decrement Control

(10) Selecting Format of Frame Buffer Read Signal

Signal formats RGB565, RGB888, αRGB1555, αRGB4444, αRGB8888, RGBα5551, RGBα8888, CLUT8, CLUT4 and CLUT1 are supported for the graphics 0, 2, and 3 processes. The YCbCr422 and YCbCr444 formats are also supported for the graphics 0 process.

The GR_FORMAT[3:0] bits select a signal format.

Table 38.10 Format Selection for Frame Buffer Read Signal

Register Name	Bit Name	Initial Value	Description
GR_FLM6	GR_FORMAT[3:0]	0	Sets the format of the frame buffer read signal. 0: RGB565 1: RGB888 2: αRGB1555 3: αRGB4444 4: αRGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8: YCbCr422 or setting prohibited * 9: YCbCr444 or setting prohibited * 10: RGBα5551 11: RGBα8888 12 to 15: Setting prohibited

Note: * Setting this value selects YCbCr422 and YCbCr444 for the graphics 0 process, and is prohibited for the graphics 2 and 3 processes.

(11) Endian Control

In the frame buffer, data is handled in 64-bit units, and endian of the data to be read can be controlled by setting the GR_RDSWA[2:0] bits. Bit 0 of these bits indicates whether 8-bit data is swapped. Bit 1 indicates whether 16-bit data is swapped. Bit 2 indicates whether 32-bit data is swapped. In the YCbCr422 format, data can be arranged with the GR_YCC_SWAP[2:0] bits.

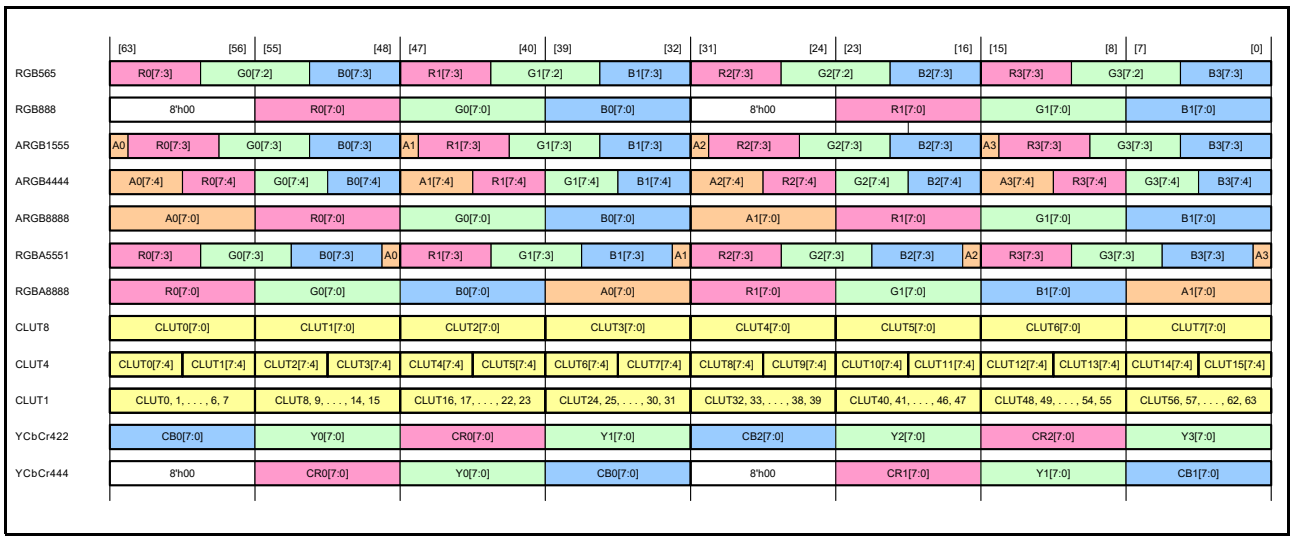


Figure 38.4 Data Arrangement with Endian Control Disabled (GR_RDSWA = 000)

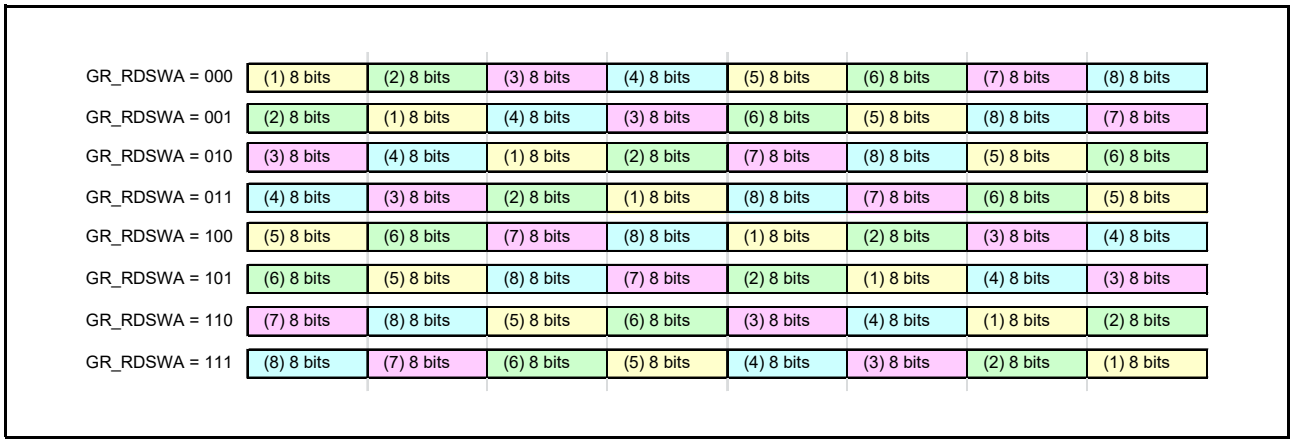


Figure 38.5 Data Arrangement with Endian Control Enabled

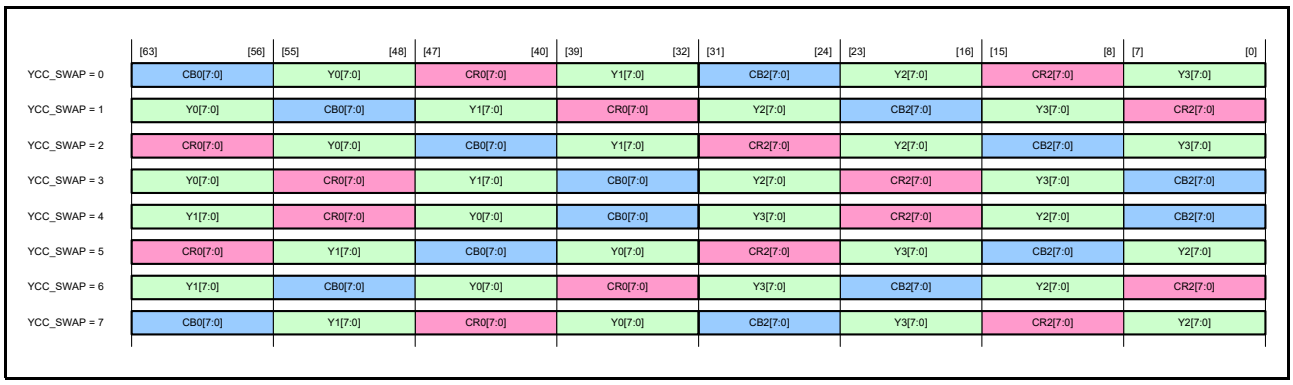


Figure 38.6 YCbCr422 Data Arrangement with Swapping Enabled

Table 38.11 Endian Control

Register Name	Bit Name	Initial Value	Description
GR_FLM6	GR_RDSWA[2:0]	0	Sets 8-, 16-, and 32-bit swap. These three bits specify the method for swapping the bits of frame buffer read data as follows. Bit 0 0: 8 bits are not swapped. 1: 8 bits are swapped. Bit 1 0: 16 bits are not swapped. 1: 16 bits are swapped. Bit 2 0: 32 bits are not swapped. 1: 32 bits are swapped. When eight bits are put together, they are swapped as follows. Each of (1) to (8) indicates eight-bit data. 000: (1) (2) (3) (4) (5) (6) (7) (8) [No swap] 001: (2) (1) (4) (3) (6) (5) (8) (7) [8-bit swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [16-bit swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [16-bit swap + 8-bit swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [32-bit swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [32-bit swap + 8-bit swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [32-bit swap + 16-bit swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [32-bit swap + 16-bit swap + 8-bit swap]
GR_FLM6	GR_YCC_SWAP [2:0]	0	Controls swapping of data read from buffer in the YCbCr422 format. * 0: Cb/Y0/Cr/Y1 1: Y0/Cb/Y1/Cr 2: Cr/Y0/Cb/Y1 3: Y0/Cr/Y1/Cb 4: Y1/Cr/Y0/Cb 5: Cr/Y1/Cb/Y0 6: Y1/Cb/Y0/Cr 7: Cb/Y1/Cr/Y0

Note: * These bits are supported for the graphics 0 process only.

(12) Display Start Pixel Setting for Read Data

When a horizontal offset is applied to display the image data in the frame buffer, the display start pixel is set with the GR_BASE[31:0] and GR_STA_POS[5:0] bits. Calculation of the values for the GR_BASE[31:0] and GR_STA_POS[5:0] bits depends on the signal format. The display start pixel can be calculated with the formulas in the table below, where H_OFF is a horizontal offset from the display start pixel.

Table 38.12 Calculation of Display Start Position for Various Signal Formats

Signal Format of Video/Graphics	Number of Bits per Pixel	Calculation Formula *1
RGB888 αRGB8888, RGBα8888 YCbCr422*2 YCbCr444*3	32	GR_BASE[31:3] = t (H_OFF ÷ 2) GR_STA_POS[5:0] = mod (H_OFF ÷ 2)
RGB565 αRGB1555, RGBα5551 αRGB4444	16	GR_BASE[31:3] = int (H_OFF ÷ 4) GR_STA_POS[5:0] = mod (H_OFF ÷ 4)
CLUT8	8	GR_BASE[31:3] = int (H_OFF ÷ 8) GR_STA_POS[5:0] = mod (H_OFF ÷ 8)
CLUT4	4	GR_BASE[31:3] = int (H_OFF ÷ 16) GR_STA_POS[5:0] = mod (H_OFF ÷ 16)
CLUT1	1	GR_BASE[31:3] = int (H_OFF ÷ 64) GR_STA_POS[5:0] = mod (H_OFF ÷ 64)

Note 1. The functions int() and mod() output a quotient and a remainder, respectively.
Note 2. The YCbCr422 format is not supported for the graphics 2 and 3 processes.
In the YCbCr422 format, 32 bits are used for two pixels (Cb, Y0, Cr, and Y1 components). Therefore, the start position is controlled in units of 32 bits.
Note 3. The YCbCr444 format is not supported for the graphics 2 and 3 processes.

Table 38.13 Setting of Display Start Pixel of Read Data

Register Name	Bit Name	Initial Value	Description
GR_FLM6	GR_STA_POS[5:0]	0	Sets the amount of data to be skipped through. Specifically data amount equal to the amount indicated by GR_STA_POS is skipped from the start of the line.
GR_FLM2	GR_BASE[31:0]	0	Frame Buffer Base Address Sets the start address of the frame buffer where frame data is to be stored. GR_BASE[4:3] and GR_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower 3 bits should be fixed to 000.

(13) YCbCr422 to YCbCr444 Conversion

Data format for the graphics 0 process is converted from YCbCr422 to YCbCr444.

This function is not supported for the graphics 2 and 3 processes.

Table 38.14 YCbCr422 to YCbCr444 Conversion

Register Name	Bit Name	Initial Value	Description
GR_FLM6	GR_CNV444_MD	0	Sets the interpolation mode for YCbCr422 to YCbCr444 conversion. * 0: Hold interpolation 1: Average interpolation

Note: * This bit is not provided for the graphics 2 and 3 processes, for which the YCbCr422 format is not supported.

(14) Bit Extension

When the value of the GR_FORMAT[3:0] bits is 0 to 3, the RGB565, RGB888, αRGB1555, and αRGB4444 formats are converted to the αRGB8888 format. When the value of the GR_FORMAT[3:0] bits is 10, the RGBα5551 format is converted to the RGBα8888 format. The RGBα5551 to RGBα8888 format conversion is omitted because it differs from the αRGB5551 to αRGB8888 format conversion only in the position of α.

- RGB565 to αRGB8888 Format Conversion

  After conversion, α[7:0] is fixed to 255.

  After conversion, R[7:0] = R[4:0] × 263 ÷ 32 (round off to an integer), approximation of #R[4:0] × 255 ÷ 31

  After conversion G[7:0] = G[5:0] × 259 ÷ 64 (round off to an integer), approximation of #G[5:0] × 255 ÷ 63

  After conversion, B[7:0] = B[4:0] × 263 ÷ 32 (round off to an integer), approximation of #B[4:0] × 255 ÷ 31

- RGB888 to αRGB8888 Format Conversion

  After conversion, α[7:0] is fixed to 255.

- αRGB1555 to αRGB8888 Format Conversion

  After conversion, α[7:0] is GR_A1 when α input is 1, and GR_A0 when 0.

  After conversion, R[7:0] = R[4:0] × 263 ÷ 32 (round off to an integer), approximation of #R[4:0] × 255 ÷ 31

  After conversion, G[7:0] = G[4:0] × 263 ÷ 32 (round off to an integer), approximation of #G[4:0] × 255 ÷ 31

  After conversion, B[7:0] = B[4:0] × 263 ÷ 32 (round off to an integer), approximation of #B[4:0] × 255 ÷ 31

- αRGB4444 to αRGB8888 Format Conversion

  After conversion, α[7:0] = α[3:0] × 17

  After conversion, R[7:0] = R[3:0] × 17

  After conversion, G[7:0] = G[3:0] × 17

  After conversion, B[7:0] = B[3:0] × 17

(15) Buffer Underflow Processing

When data read from the frame buffer cannot be completed due to bus-traffic related problems, an underflow interrupt signal is output.

38.1.3 Setting Graphics Display Area

The graphics display area is set with the GR_GRC_HS[10:0], GR_GRC_HW[10:0], GR_GRC_VS[10:0], and GR_GRC_VW[10:0] bits based on the rising edges of the Hsync and Vsync signals.

Figure 38.7 shows the graphics display area.

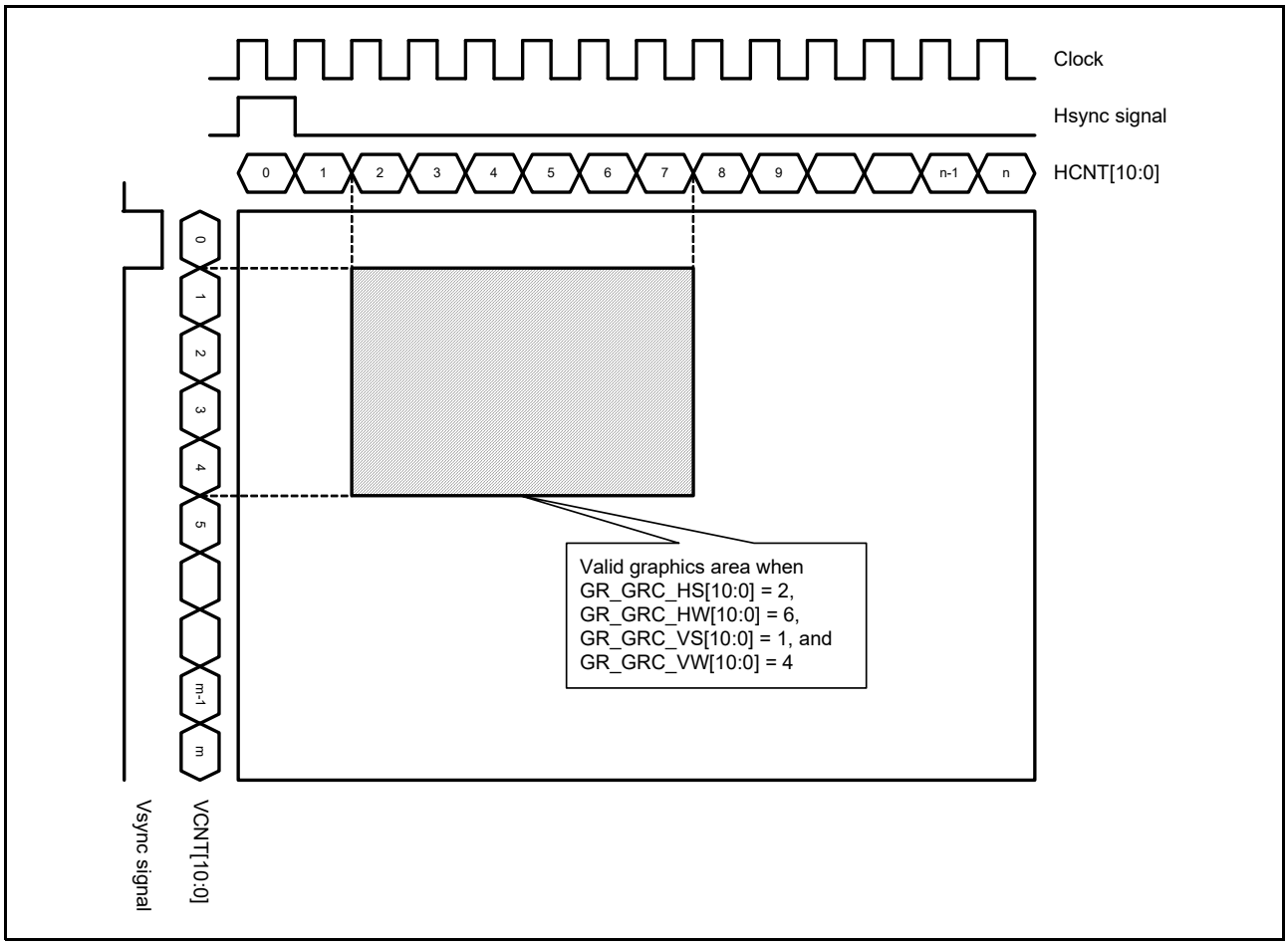


Figure 38.7 Graphics Display Area

The frame line of the graphics area can be displayed by setting the GR_GRC_DISP_ON bit to 1.

Table 38.15 Graphics Image Area Setting

Register Name	Bit Name	Initial Value	Description
GR_AB3	GR_GRC_HS[10:0]	0	Sets the horizontal start position of the graphics image area. Note: Set to 16 or greater clocks and the result of GR_GRC_HS + GR_GRC_HW should be smaller than or equal to 2015 clocks.
GR_AB3	GR_GRC_HW[10:0]	0	Sets the horizontal width of the graphics image area. Note: For displaying an image with 1- or 2-pixel horizontal width, set GR_HW to 2 and GR_GRC_HW to 1 (1-pixel) or 2 (2-pixel).
GR_AB2	GR_GRC_VS[10:0]	0	Sets the vertical start position of the graphics image area. Note: Set to 4 or greater lines and the result of GR_GRC_VS + GR_GRC_VW should be smaller than or equal to 2039 lines.
GR_AB2	GR_GRC_VW[10:0]	0	Sets the vertical width of the graphics image area.
GR_AB1	GR_GRC_DISP_ON	0	Turns on/off frame-line display of the graphics image area. 0: Frame-line display off 1: Frame-line display on

38.1.4 Interrupt Generation at Specified Line

An interrupt signal can be generated at the line specified with the GR_LINE[10:0] bits.

Table 38.16 Interrupt Generation at Specified Line

Register Name	Bit Name	Initial Value	Description
GR_CLUT_INT	GR_LINE[10:0]	0	Line Interrupt Set * When the number of lines matches the value of the GR_LINE bits, an interrupt signal is output. This function is supported for the graphics 3 process only. This function supported for the graphics 3 process is enabled even when the graphics 3 process is not used.

Note: * This function is supported for the graphics 3 process only; these bits are not supported for the graphics 0 and 2 processes.

38.1.5 Formats of Frame Buffer Read Signals and Corresponding Alpha Blending Types

Setting the GR_FORMAT[3:0] bits selects the format of the signal read from the frame buffer.

Table 38.17 shows the signal formats and the corresponding alpha blending types. The priority of the alpha value is: alpha blending in rectangular area > chroma-key processing > alpha blending in pixel units.

Table 38.17 Formats of Frame Buffer Read Signal and Corresponding Alpha Blending Types

GR_FORMAT[3:0]	Signal Format	Alpha Blending in Rectangular Area	RGB-Index Chroma-Key Processing	CLUT-Index Chroma-Key Processing	Alpha Blending in Pixel Units
0	RGB565	Supported	Supported *1	Not supported	Not supported *2
1	RGB888	Supported	Supported	Not supported	Not supported *2
2	αRGB1555	Supported	Supported *1*3	Not supported	Supported *3
3	αRGB4444	Supported	Supported *1	Not supported	Supported
4	αRGB8888	Supported	Supported	Not supported	Supported
5	CLUT8	Supported	Not supported	Supported	Supported
6	CLUT4	Supported	Not supported	Supported	Supported
7	CLUT1	Supported *4	Not supported	Supported *4	Supported *4
8	YCbCr422	Not supported *5	Not supported *5	Not supported *5	Not supported *5
9	YCbCr444	Not supported *5	Not supported *5	Not supported *5	Not supported *5
10	RGBα5551	Supported	Supported *1*3	Not supported	Supported *3
11	RGBα8888	Supported	Supported	Not supported	Supported

Note 1. When each color component of the RGB signal read from the frame buffer is not 8 bits, it is converted to 8 bits by calculation in RGB-index chroma-key processing. (See section 38.1.2 (14) Bit Extension.)
Note 2. Since α value is 255, the current graphics is always displayed.
Note 3. α value for data read from the frame buffer is specified with one bit. This one-bit signal selects one of the two registers, each of which holds an 8-bit α value.
Note 4. CLUT value for the frame buffer signal is specified with one bit. This one-bit signal selects one of the two registers, each of which holds the α, G, B, and R values (8 bits for each value). The CLUT table is not referenced.
Note 5. YCbCr422 and YCbCr444 are supported for the graphics 0 process, but any type of blending and chroma-key processing cannot be used.

38.1.6 Display Selection

The GR_DISP_SEL[1:0] bits select the graphics to be displayed from the background color, the lower-layer graphics, the current graphics, and the blended image of the lower-layer graphics and the current graphics. For blending, alpha blending in a rectangular area, multiplication with current alpha at alpha blending in a rectangular area, RGB-index chroma-key processing, CLUT-index chroma-key processing, alpha blending in one-pixel units, or premultiplication at alpha blending in one-pixel units can be selected.
This function is not supported for the VIN synthesizer.

Table 38.18 shows the settings for various display types.

Table 38.18 Settings for Various Display Types

GR_DISP_SEL [1:0]	GR_ARC_ON	GR_CK_ON	GR_ARC_MUL	GR_ACALC_MD	Processing for Graphics Area	Processing for the Area outside the Graphics Area
0	—	—	—	—	Background color	Background color
1	—	—	—	—	Lower-layer graphics	Lower-layer graphics
2	—	—	—	—	Current graphics	Background color
3	1	—	0	0	Alpha blending in a rectangular area*1	Lower-layer graphics
3	1	—	0	1	Setting prohibited	
3	1	—	1	0	Multiplication with current alpha at alpha blending in a rectangular area*2	Lower-layer graphics
3	1	—	1	1	Multiplication with current alpha at alpha blending in a rectangular area with alpha premultiplied*2	Lower-layer graphics
3	0	1	—	—	RGB-index or CLUT-index chroma-key processing	Lower-layer graphics
3	0	0	—	0	Alpha blending in one-pixel units*2	Lower-layer graphics
3	0	0	—	1	Premultiplication at alpha blending in one-pixel units*2	Lower-layer graphics

Note 1. The alpha blending in a rectangular area is not supported for the graphics 0 process in the scaler.
Note 2. The multiplication with current alpha at alpha blending in a rectangular area and alpha blending function in one-pixel units are supported for the graphics 2 and 3 processes only.

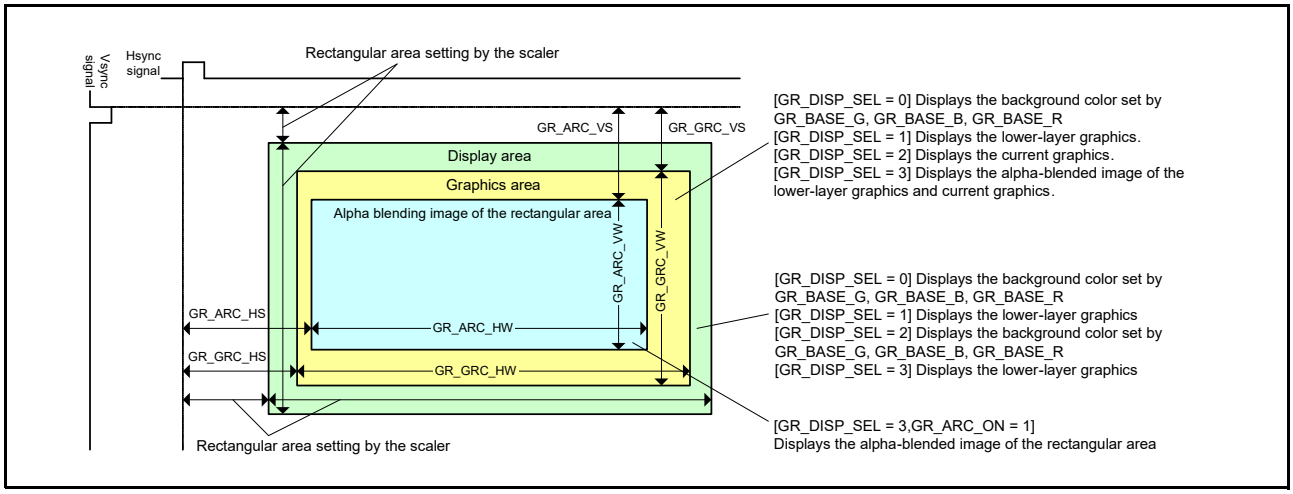


Figure 38.8 Graphic Display Types

Figure 38.9 shows the graphics planes displayed when the GR_DISP_SEL bits are set to 3.

For correspondence between the lower-layer graphics and the current graphics, see Figure 38.1.

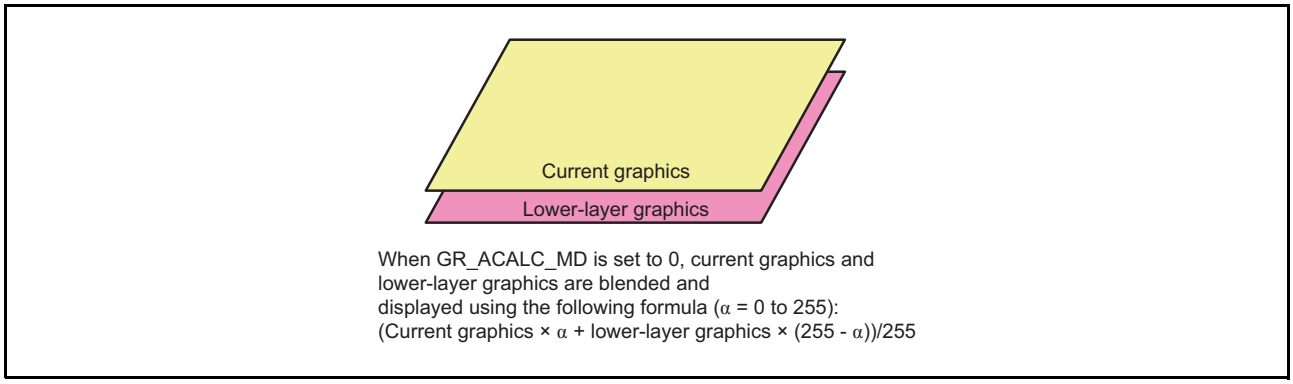


Figure 38.9 Graphics Planes with GR_DISP_SEL Set to 3

Table 38.19 Alpha Blending Setting

Register Name	Bit Name	Initial Value	Description
GR_AB1	GR_DISP_SEL [1:0]	0	Selects the graphics display mode. 0:Background color display 1:Lower-layer graphics display 2:Current graphics display 3:Blended display of lower-layer graphics and current graphics*1
GR_AB1	GR_ARC_ON	0	Turns on/off alpha blending in a rectangular area.*2 0: Off 1: On
GR_AB1	GR_ARC_MUL	0	Turns on/off multiplication processing with current alpha at alpha blending in a rectangular area *3 0: Off 1: On
GR_AB1	GR_ACALC_MD	0	Turns on/off premultiplication processing at alpha blending in one-pixel units *3 0: Off 1: On
GR_AB7	GR_CK_ON	0	Turns on/off CLUT-index/RGB-index chroma-key processing. 0: Off 1: On

Note 1. The graphics 0 process supports the chroma-key processing only. When performing chroma-key processing, set the α value for converting the pixels to be subjected to chroma-key processing, and the α value of the pixels not to be subjected to the chroma-key processing to 255 to display the current graphics only.
Note 2. This function is supported only for the graphics 2 and 3 processes. This bit is not provided for the graphics 0 process.
Note 3. This function is supported only for the graphics 2 and 3 processes. This bit is not provided for the graphics 0 process.

38.1.7 Background Color Display Processing

The color set with the GR_BASE_G[7:0], GR_BASE_B[7:0], and GR_BASE_R[7:0] bits is displayed.

G output = GR_BASE_G

B output = GR_BASE_B

R output = GR_BASE_R

Table 38.20 Background Color Setting

Register Name	Bit Name	Initial Value	Description
GR_BASE	GR_BASE_G[7:0]	0	Background color G signal G: 8 bits; unsigned (0 to 255 [LSB])
GR_BASE	GR_BASE_B[7:0]	0	Background color B signal B: 8 bits; unsigned (0 to 255 [LSB])
GR_BASE	GR_BASE_R[7:0]	0	Background color R signal R: 8 bits; unsigned (0 to 255 [LSB])

38.1.8 Lower-Layer Graphics Display Processing

The lower-layer graphics are displayed as follows:

G output = G input of lower-layer graphics

B output = B input of lower-layer graphics

R output = R input of lower-layer graphics

38.1.9 Current Graphics Display Processing

The current graphics are displayed as follows:

G output = G input of current graphics

B output = B input of current graphics

R output = R input of current graphics

38.1.10 Display with Alpha Blending in a Rectangular Area

The rectangular area subjected to alpha blending is set with the GR_ARC_HS[10:0], GR_ARC_HW[10:0], GR_ARC_VS[10:0], and GR_ARC_VW[10:0] bits based on the rising edges of the Hsync and Vsync signals. This function is not supported for the graphics 0 process.

Figure 38.10 shows the rectangular area setting for alpha blending.

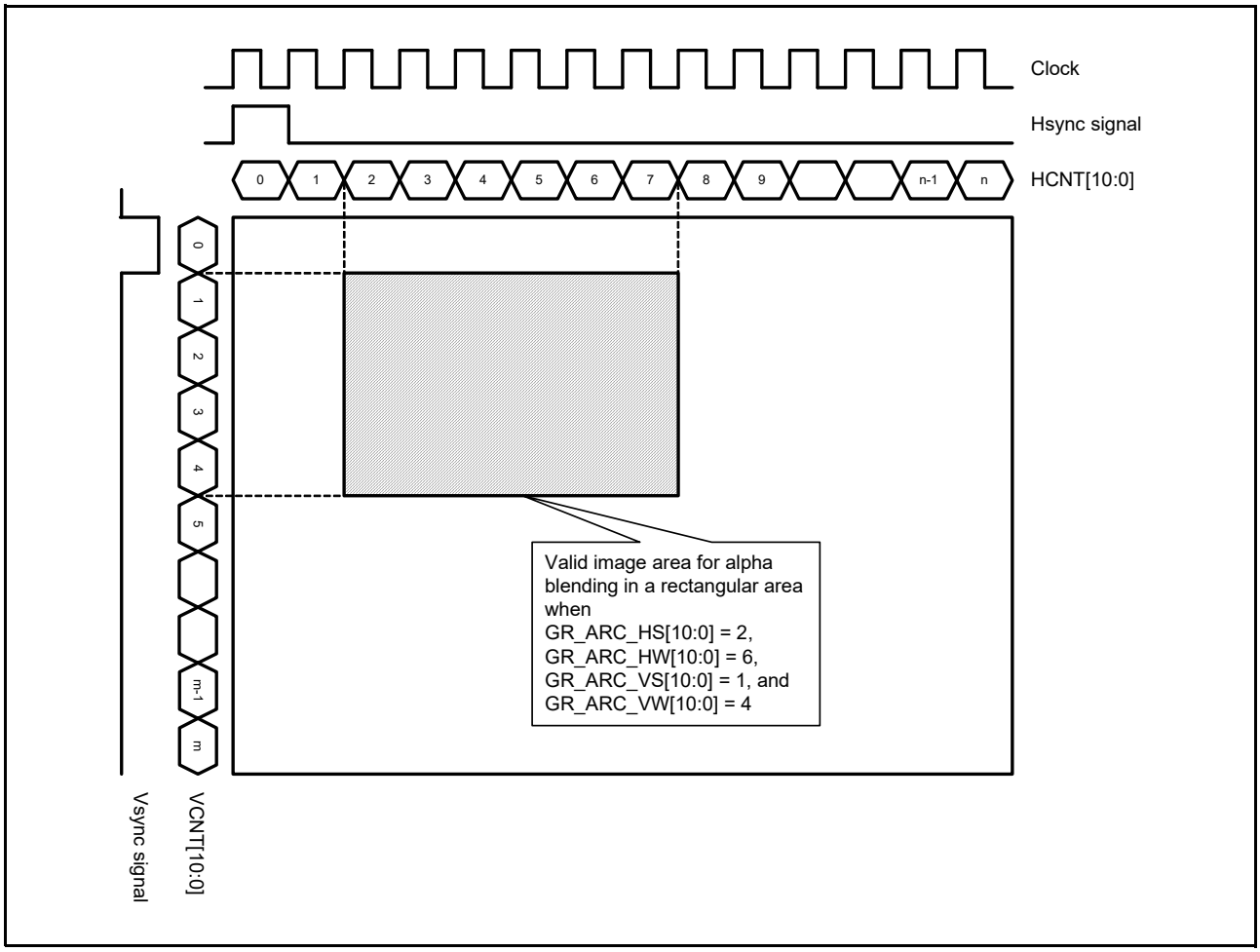


Figure 38.10 Rectangular Area Setting for Alpha Blending

The frame line of graphics area can be displayed by setting the GR_ARC_DISP_ON bit to 1.

Table 38.21 Setting of Rectangular Area for Alpha Blending

Register Name	Bit Name	Initial Value	Description
GR_AB5	GR_ARC_HS[10:0]	0	Sets the horizontal start position of the valid image area for alpha blending in a rectangular area.
GR_AB5	GR_ARC_HW[10:0]	0	Sets the horizontal width of the valid image area for alpha blending in a rectangular area.
GR_AB4	GR_ARC_VS[10:0]	0	Sets the vertical start position of the valid image area for alpha blending in a rectangular area.
GR_AB4	GR_ARC_VW[10:0]	0	Sets the vertical width of the valid image area for alpha blending in a rectangular area.
GR_AB1	GR_ARC_DISP_ON	0	Turns on/off frame-line display of the valid image area for alpha blending in a rectangular area. 0: Frame-line display off 1: Frame-line display on

In alpha blending in a rectangular area, the current graphics are faded in or out by setting the fade-in or fade-out coefficients with the GR_ARC_DEF[7:0], GR_ARC_MODE, GR_ARC_COEF[7:0], and GR_ARC_RATE[7:0] bits.

First, the value of the GR_ARC_DEF[7:0] bits is assigned to the α value.

Then, each time the Vsync signal rises for the number of times set with the GR_ARC_RATE[7:0] bits + 1, the value of the GR_ARC_COEF[7:0] bit is added to or subtracted from the α value.

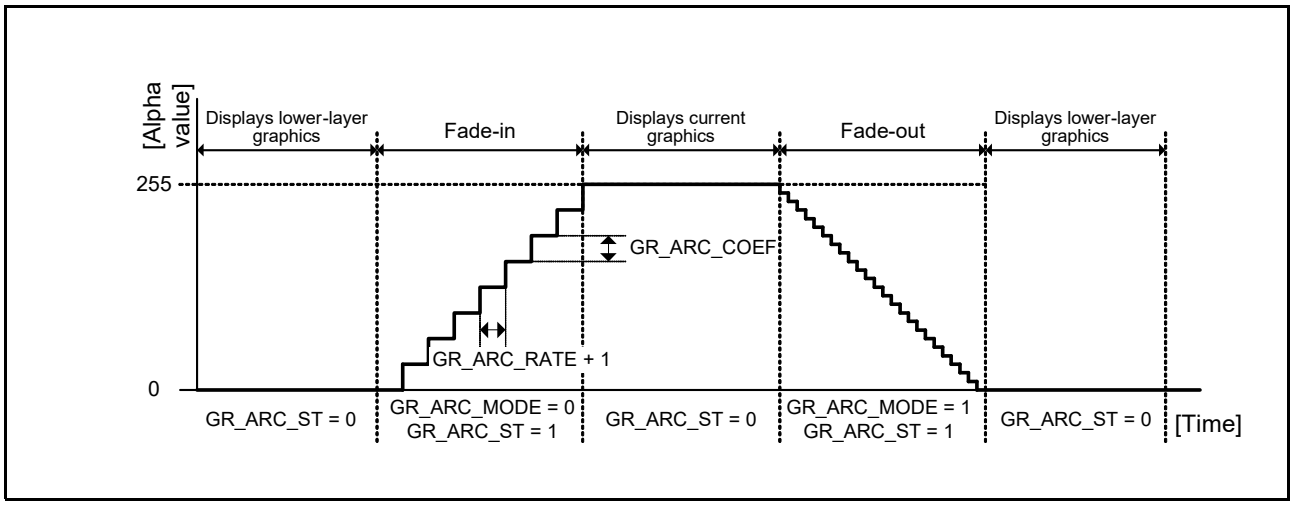


Figure 38.11 Fade In and Fade Out

Table 38.22 Setting for Alpha Blending in a Rectangular Area

Register Name	Bit Name	Initial Value	Description
GR_AB7	GR_ARC_DEF[7:0]	0	Sets the initial alpha value for alpha blending in a rectangular area.
GR_AB6	GR_ARC_MODE	0	Alpha Blending Mode in Rectangular Area 0: Addition 1: Subtraction
GR_AB6	GR_ARC_COEF[7:0]	0	Sets the alpha coefficient for alpha blending in a rectangular area. (0 to 255) [7:0]: Variation (absolute value)
GR_AB6	GR_ARC_RATE[7:0]	0	Sets the value obtained by subtracting 1 from the frame rate for alpha blending in a rectangular area.
GR_MON	GR_ARC_ST	—	Status Flag for Alpha Blending in Rectangular Area 0: Addition or subtraction has been completed. (α value is 0 or 255) 1: Addition or subtraction is in progress.

The values specified with the following expressions are used in the alpha blending calculation described in section 38.1.14, Alpha Blending Calculation.

α value = Fade-in/out coefficient

G value = G input of the current graphics

B value = B input of the current graphics

R value = R input of the current graphics

38.1.11 RGB-Index Chroma-Key Processing

The pixels that satisfy all the expressions below are subjected to RGB-index chroma-key processing.

G input of the current graphics = GR_CK_KG

B input of the current graphics = GR_CK_KB

R input of the current graphics = GR_CK_KR

In RGB-index chroma-key processing, the values specified with the following expressions are used in the alpha blending calculation described in section 38.1.14, Alpha Blending Calculation. This function is not supported in the VIN synthesizer.

α value = GR_CK_A

G value = GR_CK_G

B value = GR_CK_B

R value = GR_CK_R

For the pixels that are not subjected to RGB-index chroma-key processing, the values specified with the following expressions are used in the alpha blending calculation described in section 38.1.14, Alpha Blending Calculation.

α value = α input of the current graphics

G value = G input of the current graphics

B value = B input of the current graphics

R value = R input of the current graphics

Table 38.23 Setting for RGB-Index Chroma-Key Processing

Register Name	Bit Name	Initial Value	Description
GR_AB8	GR_CK_KG[7:0]	0	G Signal for RGB-Index Chroma-Key Processing G: 8 bits; unsigned (0 to 255 [LSB])
GR_AB8	GR_CK_KB[7:0]	0	B Signal for RGB-Index Chroma-Key Processing B: 8 bits; unsigned (0 to 255 [LSB])
GR_AB8	GR_CK_KR[7:0]	0	R Signal for RGB-Index Chroma-Key Processing R: 8 bits; unsigned (0 to 255 [LSB])
GR_AB9	GR_CK_A[7:0]	0	Replaced Alpha Signal after RGB-Index Chroma-Key Processing * α: 8 bits; unsigned (0 to 255 [LSB])
GR_AB9	GR_CK_G[7:0]	0	Replaced G Signal after RGB-Index Chroma-Key Processing G: 8 bits; unsigned (0 to 255 [LSB])
GR_AB9	GR_CK_B[7:0]	0	Replaced B Signal after RGB-Index Chroma-Key Processing B: 8 bits; unsigned (0 to 255 [LSB])
GR_AB9	GR_CK_R[7:0]	0	Replaced R Signal after RGB-Index Chroma-Key Processing R: 8 bits; unsigned (0 to 255 [LSB])

Note: * To use this function for the graphics 0 process, the alpha value should be set to 255.

38.1.12 CLUT-Index Chroma-Key Processing

The pixels that satisfy the expression below are subjected to CLUT-index chroma-key processing.

CLUT input of the current graphics = GR_CK_KCLUT

In CLUT-index chroma-key processing, the values specified with the following expressions are used in the alpha blending calculation described in section 38.1.14, Alpha Blending Calculation. This function is not supported in the VIN synthesizer.

α value = GR_CK_A

G value = GR_CK_G

B value = GR_CK_B

R value = GR_CK_R

For the pixels that are not subjected to CLUT-index chroma-key processing, the values specified with the following expressions are used in the alpha blending calculation described in section 38.1.14, Alpha Blending Calculation.

α value = α input of the current graphics

G value = G input of the current graphics

B value = B input of the current graphics

R value = R input of the current graphics

Table 38.24 Setting for CLUT-Index Chroma-Key Processing

Register Name	Bit Name	Initial Value	Description
GR_AB8	GR_CK_ KCLUT[7:0]	0	CLUT Signal for CLUT-Index Chroma-Key Processing CLUT: 8 bits; unsigned (0 to 255 [LSB])
GR_AB10	GR_A0[7:0]	0	CLUT1 α0 Signal * Replaced with α signal when in the CLUT1 format and CLUT1= 0. Replaced with α signal when in the αRGB1555 format and α = 0.
GR_AB10	GR_G0[7:0]	0	CLUT1 G0 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 0.
GR_AB10	GR_B0[7:0]	0	CLUT1 B0 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 0.
GR_AB10	GR_R0[7:0]	0	CLUT1 R0 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 0.
GR_AB11	GR_A1[7:0]	0	CLUT1 α1 Signal * Replaced with α signal when in the CLUT1 format and CLUT1 = 1. Replaced with α signal when in the αRGB1555 format and α = 1.
GR_AB11	GR_G1[7:0]	0	CLUT1 G1 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 1.
GR_AB11	GR_B1[7:0]	0	CLUT1 B1 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 1.
GR_AB11	GR_R1[7:0]	0	CLUT1 R1 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 1.

Note: * To use this function for the graphics 0 process, the alpha value should be set to 255.

38.1.13 Display with Alpha Blending in One-Pixel Units

In the alpha blending in one-pixel units, the values specified with the following expressions are used in the alpha blending calculation described in section 38.1.14, Alpha Blending Calculation. This function is not supported in the graphics 0 process and the VIN synthesizer.

α value = α input of the current graphics

G value = G input of the current graphics

B value = B input of the current graphics

R value = R input of the current graphics

38.1.14 Alpha Blending Calculation

Alpha blending of two input signals is performed using the α value as described below (rounded up if the result includes a decimal fraction).

[GR_ACALC_MD = 0]

G output = (G value × α value + G input of the lower-layer graphics × (255 − α value)) ÷ 256

B output = (B value × α value + B input of the lower-layer graphics × (255 − α value)) ÷ 256

R output = (R value × α value + R input of the lower-layer graphics × (255 − α value)) ÷ 256

[GR_ACALC_MD = 1 (premultiplication)]

G output = (G value + G input of the lower-layer graphics × (255 − α value)) ÷ 256

B output = (B value + B input of the lower-layer graphics × (255 − α value)) ÷ 256

R output = (R value + R input of the lower-layer graphics × (255 − α value)) ÷ 256

38.1.15 CLUT Table

When the signal format is CLUT8 or CLUT4, the format is converted to αRGB8888 based on the CLUT table. When the format is CLUT1, it is converted to αRGB8888 based on the register value.

Figure 38.12 shows data arrangement in the CLUT table.

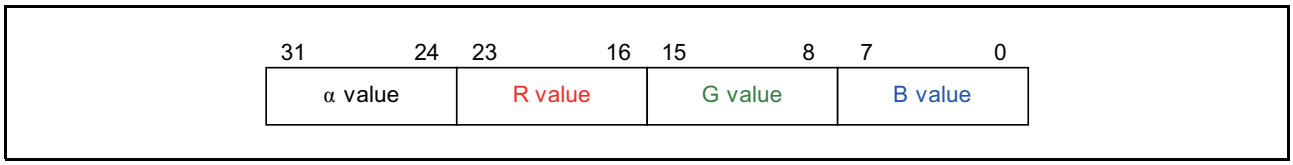


Figure 38.12 Data Arrangement in CLUT Table

The CLUT tables are arranged in the following addresses:

Graphics 0 CLUT table: H'FCFF6000 to H'FCFF63FF
(For CLUT4, addresses H'FCFF6000 to H'FCFF603F are valid.)

Graphics 2 CLUT table: H'FCFF6800 to H'FCFF6BFF
(For CLUT4, addresses H'FCFF6800 to H'FCFF683F are valid.)

Graphics 3 CLUT table: H'FCFF6C00 to H'FCFF6FFF
(For CLUT4, addresses H'FCFF6C00 to H'FCFF6C3F are valid.)

Two CLUT tables (CLUT table 0, CLUT table 1) on the different planes are allocated to the same address and one of the tables is selected with the GR_CLT_SEL bit. This allows rewriting one CLUT table when this module refers to the other CLUT table. To switch the CLUT tables after having written to one, execute a dummy read from any location in the address space of the given CLUT table before switching them.

Table 38.25 CLUT Table Selection

Register Name	Bit Name	Initial Value	Description
GR_CLUT	GR_CLT_SEL	0	CLUT Table Select Signal 0: Selects CLUT table 0. The format is converted to αRGB8888 based on the CLUT table 0. CLUT table 1 can be read from or written to by the CPU. 1: Selects CLUT table 1. The format is converted to αRGB8888 based on the CLUT table 1. CLUT table 0 can be read from or written to by the CPU.

38.1.16 Multiplication Processing with Current Alpha at Alpha Blending in Rectangular Area

In multiplication processing with current alpha at alpha blending in a rectangular area, the values specified with the following expressions are used in the alpha blending calculation described in section 38.1.14, Alpha Blending Calculation.

[GR_ARC_MUL=0]

α value = Fade-in/out coefficient

G value = G input of the current graphics

B value = B input of the current graphics

R value = R input of the current graphics

[GR_ARC_MUL=1 (premultiplication)]

α value = Fade-in/out coefficient x α input of current graphics

G value = G input of the current graphics

B value = B input of the current graphics

R value = R input of the current graphics

38.1.17 Selection of Lower-Layer Graphics in VIN Synthesizer

Graphics 0 is allocated to the lower-layer graphics in the VIN synthesizer.

Table 38.26 Selection of Lower-Layer Plane in Scaler

Register Name	Bit Name	Initial Value	Description
GR_VIN_AB1	GR_VIN_SCL_UND_SEL	0	Specifies lower-layer plane in the scaler. 0: Selects graphics 0 as lower-layer graphics. 1: Setting prohibited

38.2 Register Descriptions

Table 38.27 to Table 38.30 show the register configuration.

- Symbols used in Register Description:

Initial value: Register value after a reset
—: Undefined value
R/W: Readable/writable. The written value can be read.
R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.
R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.
R: Read-only. The write value should always be 0.
—/W: Write-only. The read value is undefined.

Table 38.27 shows the register configuration for the graphics 2 process.

Table 38.28 shows the register configuration for the graphics 3 process.

Table 38.29 shows the CLUT table configuration.

Table 38.30 shows the register configuration for the VIN synthesizer.

The register configuration for the graphics 0 process is described in section 36, Video Display Controller 6 (3): Scaler.

Table 38.27 Register Configuration of the Image Synthesizer (Graphics 2 Process)

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Graphics 2 register update control register	GR2_UPDATE	R/WC1	H'0000 0000	H'FCFF 7700	32
Frame buffer read control register (Graphics 2)	GR2_FLM_RD	R/W	H'0000 0000	H'FCFF 7704	32
Frame buffer control register 1 (Graphics 2)	GR2_FLM1	R/W	H'0000 0000	H'FCFF 7708	32
Frame buffer control register 2 (Graphics 2)	GR2_FLM2	R/W	H'0000 0000	H'FCFF 770C	32
Frame buffer control register 3 (Graphics 2)	GR2_FLM3	R/W	H'0000 0000	H'FCFF 7710	32
Frame buffer control register 4 (Graphics 2)	GR2_FLM4	R/W	H'0000 0000	H'FCFF 7714	32
Frame buffer control register 5 (Graphics 2)	GR2_FLM5	R/W	H'0000 03FF	H'FCFF 7718	32
Frame buffer control register 6 (Graphics 2)	GR2_FLM6	R/W	H'0000 0000	H'FCFF 771C	32
Alpha blending control register 1 (Graphics 2)	GR2_AB1	R/W	H'0000 0000	H'FCFF 7720	32
Alpha blending control register 2 (Graphics 2)	GR2_AB2	R/W	H'0000 0000	H'FCFF 7724	32
Alpha blending control register 3 (Graphics 2)	GR2_AB3	R/W	H'0000 0000	H'FCFF 7728	32
Alpha blending control register 4 (Graphics 2)	GR2_AB4	R/W	H'0000 0000	H'FCFF 772C	32
Alpha blending control register 5 (Graphics 2)	GR2_AB5	R/W	H'0000 0000	H'FCFF 7730	32
Alpha blending control register 6 (Graphics 2)	GR2_AB6	R/W	H'0000 0000	H'FCFF 7734	32

Table 38.27 Register Configuration of the Image Synthesizer (Graphics 2 Process)

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Alpha blending control register 7 (Graphics 2)	GR2_AB7	R/W	H'00FF 0000	H'FCFF 7738	32
Alpha blending control register 8 (Graphics 2)	GR2_AB8	R/W	H'0000 0000	H'FCFF 773C	32
Alpha blending control register 9 (Graphics 2)	GR2_AB9	R/W	H'0000 0000	H'FCFF 7740	32
Alpha blending control register 10 (Graphics 2)	GR2_AB10	R/W	H'0000 0000	H'FCFF 7744	32
Alpha blending control register 11 (Graphics 2)	GR2_AB11	R/W	H'0000 0000	H'FCFF 7748	32
Background color control register (Graphics 2)	GR2_BASE	R/W	H'0000 0000	H'FCFF 774C	32
CLUT table control register (Graphics 2)	GR2_CLUT	R/W	H'0000 0000	H'FCFF 7750	32
Status monitor register (Graphics 2)	GR2_MON	R	H'0000 0000	H'FCFF 7754	32

Table 38.28 Register Configuration of the Image Synthesizer (Graphics 3 Process)

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Graphics 3 register update control register	GR3_UPDATE	R/WC1	H'0000 0000	H'FCFF 7780	32
Frame buffer read control register (Graphics 3)	GR3_FLM_RD	R/W	H'0000 0000	H'FCFF 7784	32
Frame buffer control register 1 (Graphics 3)	GR3_FLM1	R/W	H'0000 0000	H'FCFF 7788	32
Frame buffer control register 2 (Graphics 3)	GR3_FLM2	R/W	H'0000 0000	H'FCFF 778C	32
Frame buffer control register 3 (Graphics 3)	GR3_FLM3	R/W	H'0000 0000	H'FCFF 7790	32
Frame buffer control register 4 (Graphics 3)	GR3_FLM4	R/W	H'0000 0000	H'FCFF 7794	32
Frame buffer control register 5 (Graphics 3)	GR3_FLM5	R/W	H'0000 03FF	H'FCFF 7798	32
Frame buffer control register 6 (Graphics 3)	GR3_FLM6	R/W	H'0000 0000	H'FCFF 779C	32
Alpha blending control register 1 (Graphics 3)	GR3_AB1	R/W	H'0000 0000	H'FCFF 77A0	32
Alpha blending control register 2 (Graphics 3)	GR3_AB2	R/W	H'0000 0000	H'FCFF 77A4	32
Alpha blending control register 3 (Graphics 3)	GR3_AB3	R/W	H'0000 0000	H'FCFF 77A8	32
Alpha blending control register 4 (Graphics 3)	GR3_AB4	R/W	H'0000 0000	H'FCFF 77AC	32
Alpha blending control register 5 (Graphics 3)	GR3_AB5	R/W	H'0000 0000	H'FCFF 77B0	32
Alpha blending control register 6 (Graphics 3)	GR3_AB6	R/W	H'0000 0000	H'FCFF 77B4	32
Alpha blending control register 7 (Graphics 3)	GR3_AB7	R/W	H'00FF 0000	H'FCFF 77B8	32
Alpha blending control register 8 (Graphics 3)	GR3_AB8	R/W	H'0000 0000	H'FCFF 77BC	32
Alpha blending control register 9 (Graphics 3)	GR3_AB9	R/W	H'0000 0000	H'FCFF 77C0	32

Table 38.28 Register Configuration of the Image Synthesizer (Graphics 3 Process)

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Alpha blending control register 10 (Graphics 3)	GR3_AB10	R/W	H'0000 0000	H'FCFF 77C4	32
Alpha blending control register 11 (Graphics 3)	GR3_AB11	R/W	H'0000 0000	H'FCFF 77C8	32
Background color control register (Graphics 3)	GR3_BASE	R/W	H'0000 0000	H'FCFF 77CC	32
CLUT table and interrupt control register (Graphics 3)	GR3_CLUT_INT	R/W	H'0000 0000	H'FCFF 77D0	32
Status monitor register (Graphics 3)	GR3_MON	R	H'0000 0000	H'FCFF 77D4	32

Table 38.29 CLUT Table Configuration

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Graphics 0 CLUT table	GR0_CLUTT	R/W	—	H'FCFF 6000 to H'FCFF 63FF	32
Graphics 2 CLUT table	GR2_CLUTT	R/W	—	H'FCFF 6800 to H'FCFF 6BFF	32
Graphics 3 CLUT table	GR3_CLUTT	R/W	—	H'FCFF 6C00 to H'FCFF 6FFF	32

Table 38.30 Register Configuration of the VIN Synthesizer

Name	Abbreviation	R/W	Initial Value	Address	Access Size
VIN synthesizer register update control register	GR_VIN_UPDATE	R/WC1	H'0000 0000	H'FCFF 7E00	32
Alpha blending control register 1 (VIN synthesizer)	GR_VIN _AB1	R/W	H'0000 0000	H'FCFF 7E20	32

38.2.1 Graphics 2 Register Update Control Register (GR2_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GR2_UPDATE	—	—	—	GR2_P_VEN	—	—	—	GR2_IBUS_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR2_UPDATE	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR2_P_VEN	0	R/WC1	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR2_IBUS_VEN	0	R/WC1	Frame Buffer Read Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

38.2.2 Frame Buffer Read Control Register (Graphics 2) (GR2_FLM_RD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_R_ENB
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR2_R_ENB	0	R/W	Frame Buffer Read Enable 0: Frame buffer reading is disabled. 1: Frame buffer reading is enabled.

Note: This register is updated when GR2_IBUS_VEN in GR2_UPDATE is 1.

38.2.3 Frame Buffer Control Register 1 (Graphics 2) (GR2_FLM1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_LN_OFF_DIR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR2_FLM_SEL[1:0]		—	—	—	—	—	—	—	GR2_BST_MD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR2_LN_OFF_DIR	0	R/W	Selects the line offset address direction of the frame buffer. 0: Increments the address by the line offset address. 1: Decrements the address by the line offset address.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	GR2_FLM_SEL[1:0]	0	R/W	Selects a frame buffer address setting signal. 0: Selects frame 0. 1: Selects register GR2_FLM_NUM. 2: Selects frame 0. 3: Setting prohibited
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR2_BST_MD	0	R/W	Frame Buffer Burst Transfer Mode 0: 32-byte transfer 1: 128- byte transfer

Note: GR2_LN_OFF_DIR and GR2_FLM_SEL are updated when GR2_IBUS_VEN in GR2_UPDATE is 1.
GR2_BST_MD is updated when GR2_IBUS_VEN and GR2_P_VEN in GR2_UPDATE are 1.

38.2.4 Frame Buffer Control Register 2 (Graphics 2) (GR2_FLM2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_BASE[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_BASE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GR2_BASE [31:0]	0	R/W	Frame Buffer Base Address (upper) Sets the start address of the frame buffer where frame data is to be stored. GR_BASE[4:3] and GR_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower 3 bits should be fixed to 000.

Note: This register is updated when GR2_IBUS_VEN and GR2_P_VEN in GR2_UPDATE are 1.

38.2.5 Frame Buffer Control Register 3 (Graphics 2) (GR2_FLM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	GR2_LN_OFF[14:0]														
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR2_FLM_NUM[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	GR2_LN_ OFF[14:0]	0	R/W	Frame Buffer Line Offset Address Sets the line offset address for calculating the start address of each line. Line 0: GR2_BASE Line 1: GR2_BASE + GR2_LN_OFF × 1 : Line n: GR2_BASE + GR2_LN_OFF × n For 32 byte transfer, the lower 5 bits should be fixed to 0_0000. For 128 byte transfer, the lower 7 bits should be fixed to 000_0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	GR2_FLM_ NUM[9:0]	0	R/W	Frame Number of Frame Buffer Manually set the frame number when GR2_FLM_SEL = 1.

Note: This register is updated when GR2_IBUS_VEN in GR2_UPDATE is 1.

38.2.6 Frame Buffer Control Register 4 (Graphics 2) (GR2_FLM4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	GR2_FLM_OFF[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_FLM_OFF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	GR2_FLM_OFF[22:0]	0	R/W	Frame Buffer Frame Offset Address (upper) Specifies the frame offset address used for calculating the start address of each frame buffer when more than one buffer is used. Buffer 0: GR2_BASE Buffer 1: GR2_BASE + GR2_FLM_OFF × 1 : Buffer n: GR2_BASE + GR2_FLM_OFF × n For 32 byte transfer, the lower 5 bits should be fixed to 0_0000. For 128 byte transfer, the lower 7 bits should be fixed to 000_0000.

Note: This register is updated when GR2_IBUS_VEN in GR2_UPDATE is 1.

38.2.7 Frame Buffer Control Register 5 (Graphics 2) (GR2_FLM5)

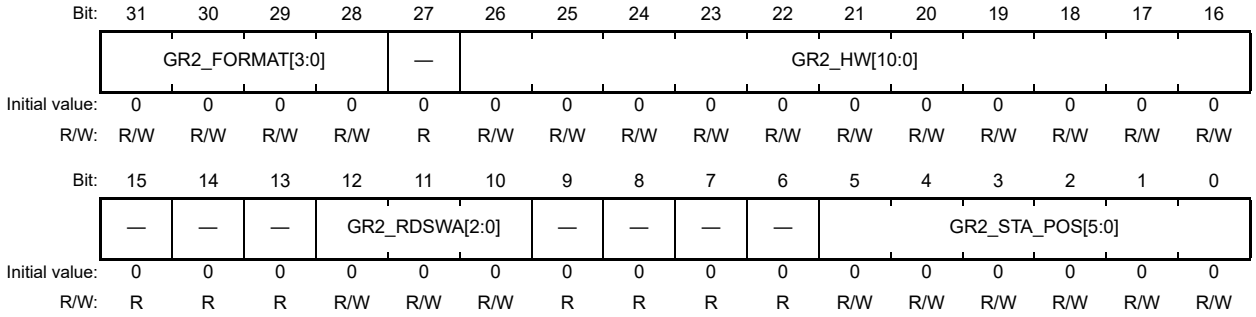
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_FLM_LNUM[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_FLM_LOOP[10:0]										
Initial value:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR2_FLM_LNUM[10:0]	0	R/W	Sets number of lines in a frame The number of lines is (GR2_FLM_LNUM + 1).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR2_FLM_LOOP[10:0]	1023	R/W	Number of lines when reading the addresses repeatedly by returning to the start address after reaching the end address. The number of lines is (GR2_FLM_LOOP + 1).

Note: This register is updated when GR2_IBUS_VEN in GR2_UPDATE is 1.

38.2.8 Frame Buffer Control Register 6 (Graphics 2) (GR2_FLM6)



Bit	Bit Name	Initial Value	R/W	Description
31 to 28	GR2_ FORMAT[3:0]	0	R/W	Sets the format of the frame buffer read signal. 0: RGB565 1: RGB888 2: αRGB1555 3: αRGB4444 4: αRGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8: Setting prohibited 9: Setting prohibited 10: RGBα5551 11: RGBα8888 12 to 15: Setting prohibited
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26 to 16	GR2_HW [10:0]	0	R/W	Sets the width of the horizontal valid period. The width is (GR2_HW + 1) pixels. Note: Set to 2 or greater.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 10	GR2_ RDSWA[2:0]	0	R/W	Sets 8-, 16-, and 32-bit swap. These three bits specify the method for swapping the bits of frame buffer read data as follows. Bit 0 0: 8 bits are not swapped. 1: 8 bits are swapped. Bit 1 0: 16 bits are not swapped. 1: 16 bits are swapped. Bit 2 0: 32 bits are not swapped. 1: 32 bits are swapped. When eight bits are put together, they are swapped as follows. Each of (1) to (8) indicates eight-bit data. 000: (1) (2) (3) (4) (5) (6) (7) (8) [No swap] 001: (2) (1) (4) (3) (6) (5) (8) (7) [8-bit swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [16-bit swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [16-bit swap + 8-bit swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [32-bit swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [32-bit swap + 8-bit swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [32-bit swap + 16-bit swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [32-bit swap + 16-bit swap + 8-bit swap]
9 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	GR2_STA_ POS[5:0]	0	R/W	Sets the amount of data to be skipped through. Specifically data amount equal to the amount indicated by GR2_STA_POS is skipped from the start of the line.

Note: GR2_STA_POS is updated when GR2_P_VEN in GR2_UPDATE is 1. GR2_RDSWA is updated when GR2_UPDATE in GR2_UPDATE is 1.
GR2_FORMAT and GR2_HW are updated when GR2_IBUS_VEN and GR2_P_VEN in GR2_UPDATE are 1.

38.2.9 Alpha Blending Control Register 1 (Graphics 2) (GR2_AB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_ ARC_MUL	GR2_ ACALC_MD	—	GR2_ARC_ ON	—	—	—	GR2_ARC_ DISP_ON	—	—	—	GR2_GRC_ DISP_ON	—	—	GR2_DISP_SEL [1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	GR2_ARC_MUL	0	R/W	Turns on/off multiplication processing with current alpha at alpha blending in a rectangular area. 0: Off 1: On
14	GR2_ACALC_MD	0	R/W	Turns on/off premultiplication processing at alpha blending in one-pixel units. 0: Off 1: On
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	GR2_ARC_ON	0	R/W	Turns on/off alpha blending in a rectangular area. 0: Off 1: On
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR2_ARC_ DISP_ON	0	R/W	Turns on/off frame-line display of the image area for alpha blending in a rectangular area. 0: Frame-line display off 1: Frame-line display on
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR2_GRC_ DISP_ON	0	R/W	Turns on/off frame-line display of the graphics image area. 0: Frame-line display off 1: Frame-line display on
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	GR2_DISP_ SEL[1:0]	0	R/W	Selects the graphics display mode. 0: Background color display 1: Lower-layer graphics display 2: Current graphics display 3: Blended display of lower-layer graphics and current graphics

Note: This register is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.10 Alpha Blending Control Register 2 (Graphics 2) (GR2_AB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_GRC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_GRC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR2_GRC_VS[10:0]	0	R/W	Sets the vertical start position of the graphics image area. Note: Set to 4 or greater lines and the result of GR2_GRC_VS + GR2_GRC_VW should be smaller than or equal to 2039 lines.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR2_GRC_VW[10:0]	0	R/W	Sets the vertical width of the graphics image area.

Note: This register is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.11 Alpha Blending Control Register 3 (Graphics 2) (GR2_AB3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_GRC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_GRC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR2_GRC_HS[10:0]	0	R/W	Sets the horizontal start position of the graphics image area. Note: Set to 16 or greater clocks and the result of GR2_GRC_HS + GR2_GRC_HW should be smaller than or equal to 2015 clocks.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR2_GRC_HW[10:0]	0	R/W	Sets the horizontal width of the graphics image area. Note: For displaying an image with 1- or 2-pixel horizontal width, set GR2_HW to 2 and GR2_GRC_HW to 1 (1-pixel) or 2 (2-pixel).

Note: This register is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.12 Alpha Blending Control Register 4 (Graphics 2) (GR2_AB4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_ARC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_ARC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR2_ARC_VS[10:0]	0	R/W	Sets the vertical start position of the valid image area for alpha blending in a rectangular area.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR2_ARC_VW[10:0]	0	R/W	Sets the vertical width of the valid image area for alpha blending in a rectangular area.

Note: This register is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.13 Alpha Blending Control Register 5 (Graphics 2) (GR2_AB5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_ARC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_ARC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR2_ARC_HS[10:0]	0	R/W	Sets the horizontal start position of the valid image area for alpha blending in a rectangular area.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR2_ARC_HW[10:0]	0	R/W	Sets the horizontal width of the valid image area for alpha blending in a rectangular area.

Note: This register is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.14 Alpha Blending Control Register 6 (Graphics 2) (GR2_AB6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	GR2_ARC_MODE	GR2_ARC_COEF[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	GR2_ARC_RATE[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	GR2_ARC_MODE	0	R/W	Alpha Blending Mode in Rectangular Area 0: Addition 1: Subtraction
23 to 16	GR2_ARC_COEF[7:0]	0	R/W	Sets the alpha coefficient for alpha blending in a rectangular area. (0 to 255) [7:0]: Variation (absolute value)
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	GR2_ARC_RATE[7:0]	0	R/W	Sets the value obtained by subtracting 1 from the frame rate for alpha blending in a rectangular area.

Note: This bit is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.15 Alpha Blending Control Register 7 (Graphics 2) (GR2_AB7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR2_ARC_DEF[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_CK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR2_ARC_DEF[7:0]	255	R/W	Sets the initial alpha value for alpha blending in a rectangular area.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR2_CK_ON	0	R/W	Turns on/off CLUT-index/RGB-index chroma-key processing. 0: Off 1: On

Note: This register is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.16 Alpha Blending Control Register 8 (Graphics 2) (GR2_AB8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_CK_KCLUT[7:0]								GR2_CK_KG[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_CK_KB[7:0]								GR2_CK_KR[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR2_CK_ KCLUT[7:0]	0	R/W	CLUT Signal for CLUT-Index Chroma-Key Processing CLUT: Unsigned 8 bits (0 to 255 [LSB])
23 to 16	GR2_CK_ KG[7:0]	0	R/W	G Signal for RGB-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR2_CK_ KB[7:0]	0	R/W	B Signal for RGB-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR2_CK_ KR[7:0]	0	R/W	R Signal for RGB-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.17 Alpha Blending Control Register 9 (Graphics 2) (GR2_AB9)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_CK_A[7:0]								GR2_CK_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_CK_B[7:0]								GR2_CK_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR2_CK_A [7:0]	0	R/W	Replaced Alpha Signal after RGB/CLUT-Index Chroma-Key Processing α: Unsigned 8 bits (0 to 255 [LSB])
23 to 16	GR2_CK_G [7:0]	0	R/W	Replaced G Signal after RGB/CLUT-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR2_CK_B [7:0]	0	R/W	Replaced B Signal after RGB/CLUT-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR2_CK_R [7:0]	0	R/W	Replaced R Signal after RGB/CLUT-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.18 Alpha Blending Control Register 10 (Graphics 2) (GR2_AB10)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_A0[7:0]								GR2_G0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_B0[7:0]								GR2_R0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR2_A0 [7:0]	0	R/W	CLUT1 α0 Signal Replaced with α signal when in the CLUT1 format and CLUT1= 0. Replaced with α signal when in the αRGB1555 or RGBα5551 format and α = 0.
23 to 16	GR2_G0 [7:0]	0	R/W	CLUT1 G0 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 0.
15 to 8	GR2_B0 [7:0]	0	R/W	CLUT1 B0 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 0.
7 to 0	GR2_R0 [7:0]	0	R/W	CLUT1 R0 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 0.

Note: This register is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.19 Alpha Blending Control Register 11 (Graphics 2) (GR2_AB11)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_A1[7:0]								GR2_G1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_B1[7:0]								GR2_R1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR2_A1 [7:0]	0	R/W	CLUT1 α1 Signal Replaced with α signal when in the CLUT1 format and CLUT1 = 1. Replaced with α signal when in the αRGB1555 or RGBα5551 format and α = 1.
23 to 16	GR2_G1 [7:0]	0	R/W	CLUT1 G1 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 1.
15 to 8	GR2_B1 [7:0]	0	R/W	CLUT1 B1 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 1.
7 to 0	GR2_R1 [7:0]	0	R/W	CLUT1 R1 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 1.

Note: This register is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.20 Background Color Control Register (Graphics 2) (GR2_BASE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR2_BASE_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_BASE_B[7:0]								GR2_BASE_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR2_BASE_G [7:0]	0	R/W	Background Color G Signal G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR2_BASE_B [7:0]	0	R/W	Background Color B Signal B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR2_BASE_R [7:0]	0	R/W	Background Color R Signal R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.21 CLUT Table Control Register (Graphics 2) (GR2_CLUT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_CLT_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR2_CLT_ SEL	0	R/W	CLUT Table Select Signal 0: Selects CLUT table 0. The format is converted to αRGB8888 based on the CLUT table 0. CLUT table 1 can be read from or written to by the CPU. 1: Selects CLUT table 1. The format is converted to αRGB8888 based on the CLUT table 1. CLUT table 0 can be read from or written to by the CPU.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: This register is updated when GR2_P_VEN in GR2_UPDATE is 1.

38.2.22 Status Monitor Register (Graphics 2) (GR2_MON)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_ARC_ST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR2_ARC_ST	0	R	Status Flag for Alpha Blending in Rectangular Area 0: Addition or subtraction has been completed. (α value is 0 or 255) 1: Addition or subtraction is in progress.

38.2.23 Graphics 3 Register Update Control Register (GR3_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GR3UPDATE	—	—	—	GR3_P_VEN	—	—	—	GR3_IBUS_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR3_UPDATE	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR3_P_VEN	0	R/WC1	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR3_IBUS_VEN	0	R/WC1	Frame Buffer Read Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

38.2.24 Frame Buffer Read Control Register (Graphics 3) (GR3_FLM_RD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_R_ENB
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR3_R_ENB	0	R/W	Frame Buffer Read Enable 0: Frame buffer reading is disabled. 1: Frame buffer reading is enabled.

Note: This register is updated when GR3_IBUS_VEN in GR3_UPDATE is 1.

38.2.25 Frame Buffer Control Register 1 (Graphics 3) (GR3_FLM1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_LN_OFF_DIR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR3_FLM_SEL[1:0]		—	—	—	—	—	—	—	GR3_BST_MD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR3_LN_OFF_DIR	0	R/W	Selects the line offset address direction of the frame buffer. 0: Increments the address by the line offset address. 1: Decrements the address by the line offset address.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	GR3_FLM_SEL[1:0]	0	R/W	Selects a frame buffer address setting signal. 0: Selects frame 0. 1: Selects register GR3_FLM_NUM. 2: Selects frame 0. 3: Setting prohibited
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR3_BST_MD	0	R/W	Frame Buffer Burst Transfer Mode 0: 32-byte transfer 1: 128- byte transfer

Note: GR3_LN_OFF_DIR and GR3_FLM_SEL are updated when GR3_IBUS_VEN in GR3_UPDATE is 1.
GR3_BST_MD is updated when GR3_IBUS_VEN and GR3_P_VEN in GR3_UPDATE are 1.

38.2.26 Frame Buffer Control Register 2 (Graphics 3) (GR3_FLM2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_BASE[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_BASE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GR3_BASE [31:0]	0	R/W	Frame Buffer Base Address (upper) Sets the start address of the frame buffer where frame data is to be stored. GR_BASE[4:3] and GR_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower 3 bits should be fixed to 000.

Note: This register is updated when GR3_IBUS_VEN and GR3_P_VEN in GR3_UPDATE are 1.

38.2.27 Frame Buffer Control Register 3 (Graphics 3) (GR3_FLM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	GR3_LN_OFF[14:0]														
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR3_FLM_NUM[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	GR3_LN_ OFF[14:0]	0	R/W	Frame Buffer Line Offset Address Sets the line offset address for calculating the start address of each line. Line 0: GR3_BASE Line 1: GR3_BASE + GR3_LN_OFF × 1 : Line n: GR3_BASE + GR3_LN_OFF × n For 32 byte transfer, the lower 5 bits should be fixed to 0_0000. For 128 byte transfer, the lower 7 bits should be fixed to 000_0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	GR3_FLM_ NUM[9:0]	0	R/W	Frame Number of Frame Buffer Manually set the frame number when GR3_FLM_SEL = 1.

Note: This register is updated when GR3_IBUS_VEN in GR3_UPDATE is 1.

38.2.28 Frame Buffer Control Register 4 (Graphics 3) (GR3_FLM4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	GR3_FLM_OFF[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_FLM_OFF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	GR3_FLM_OFF[22:0]	0	R/W	Frame Buffer Frame Offset Address (upper) Specifies the frame offset address used for calculating the start address of each frame buffer when more than one buffer is used. Buffer 0: GR3_BASE Buffer 1: GR3_BASE + GR3_FLM_OFF × 1 : Buffer n: GR3_BASE + GR3_FLM_OFF × n For 32 byte transfer, the lower 5 bits should be fixed to 0_0000. For 128 byte transfer, the lower 7 bits should be fixed to 000_0000.

Note: This register is updated when GR3_IBUS_VEN in GR3_UPDATE is 1.

38.2.29 Frame Buffer Control Register 5 (Graphics 3) (GR3_FLM5)

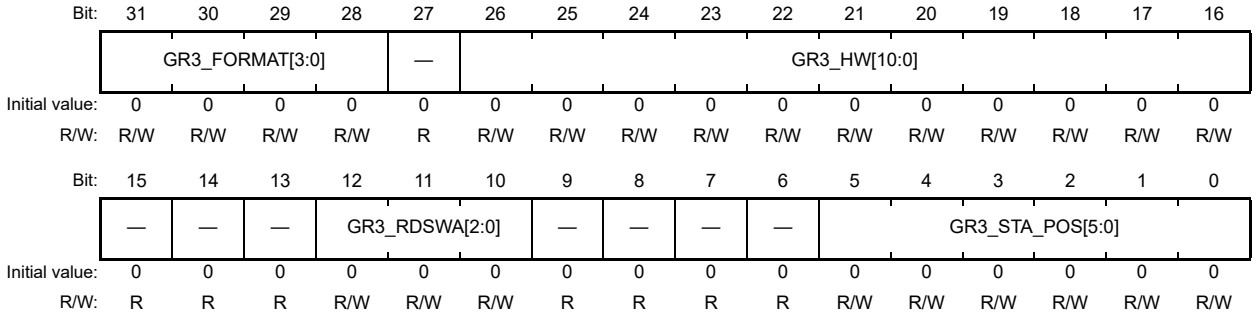
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_FLM_LNUM[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_FLM_LOOP[10:0]										
Initial value:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR3_FLM_LNUM[10:0]	0	R/W	Sets number of lines in a frame The number of lines is (GR3_FLM_LNUM + 1).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR3_FLM_LOOP[10:0]	1023	R/W	Number of lines when reading the addresses repeatedly by returning to the start address after reaching the end address. The number of lines is (GR3_FLM_LOOP + 1).

Note: This register is updated when GR3_IBUS_VEN in GR3_UPDATE is 1.

38.2.30 Frame Buffer Control Register 6 (Graphics 3) (GR3_FLM6)



Bit	Bit Name	Initial Value	R/W	Description
31 to 28	GR3_FORMAT [3:0]	0	R/W	Sets the format of the frame buffer read signal. 0: RGB565 1: RGB888 2: αRGB1555 3: αRGB4444 4: αRGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8: Setting prohibited 9: Setting prohibited 10: RGBα5551 11: RGBα8888 12 to 15: Setting prohibited
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26 to 16	GR3_HW[10:0]	0	R/W	Sets the width of the horizontal valid period. The width is (GR3_HW + 1) pixels. Note: Set to 2 or greater.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 10	GR3_RDSWA [2:0]	0	R/W	Sets 8-, 16-, and 32-bit swap. These three bits specify the method for swapping the bits of frame buffer read data as follows. Bit 0 0: 8 bits are not swapped. 1: 8 bits are swapped. Bit 1 0: 16 bits are not swapped. 1: 16 bits are swapped. Bit 2 0: 32 bits are not swapped. 1: 32 bits are swapped. When eight bits are put together, they are swapped as follows. Each of (1) to (8) indicates eight-bit data. 000: (1) (2) (3) (4) (5) (6) (7) (8) [No swap] 001: (2) (1) (4) (3) (6) (5) (8) (7) [8-bit swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [16-bit swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [16-bit swap + 8-bit swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [32-bit swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [32-bit swap + 8-bit swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [32-bit swap + 16-bit swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [32-bit swap + 16-bit swap + 8-bit swap]
9 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	GR3_STA_ POS[5:0]	0	R/W	Sets the amount of data to be skipped through. Specifically data amount equal to the amount indicated by GR3_STA_POS is skipped from the start of the line.

Note: GR3_STA_POS is updated when GR3_P_VEN in GR3_UPDATE is 1. GR3_RDSWA is updated when GR3_UPDATE in GR3_UPDATE is 1.
GR3_FORMAT and GR3_HW are updated when GR3_IBUS_VEN and GR3_P_VEN in GR3_UPDATE are 1.

38.2.31 Alpha Blending Control Register 1 (Graphics 3) (GR3_AB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_ARC_MUL	GR3_ACALC_MD	—	GR3_ARC_ON	—	—	—	GR3_ARC_DISP_ON	—	—	—	GR3_GRC_DISP_ON	—	—	GR3_DISP_SEL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	GR3_ARC_MUL	0	R/W	Turns on/off multiplication processing with current alpha at alpha blending in a rectangular area. 0: Off 1: On
14	GR3_ACALC_MD	0	R/W	Turns on/off premultiplication processing at alpha blending in one-pixel units. 0: Off 1: On
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	GR3_ARC_ON	0	R/W	Turns on/off alpha blending in a rectangular area. 0: Off 1: On
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR3_ARC_DISP_ON	0	R/W	Turns on/off frame-line display of the image area for alpha blending in a rectangular area. 0: Frame-line display off 1: Frame-line display on
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR3_GRC_DISP_ON	0	R/W	Turns on/off frame-line display of the graphics image area. 0: Frame-line display off 1: Frame-line display on
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	GR3_DISP_SEL[1:0]	0	R/W	Selects the graphics display mode. 0: Background color display 1: Lower-layer graphics display 2: Current graphics display 3: Blended display of lower-layer graphics and current graphics

Note: This register is updated when GR3_P_VEN in GR3_UPDATE is 1.

38.2.32 Alpha Blending Control Register 2 (Graphics 3) (GR3_AB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_GRC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_GRC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR3_GRC_VS[10:0]	0	R/W	Sets the vertical start position of the graphics image area. Note: Set to 4 or greater lines and the result of GR3_GRC_VS + GR3_GRC_VW should be smaller than or equal to 2039 lines.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR3_GRC_VW[10:0]	0	R/W	Sets the vertical width of the graphics image area.

Note: This register is updated when GR3_P_VEN in GR3_UPDATE is 1.

38.2.33 Alpha Blending Control Register 3 (Graphics 3) (GR3_AB3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_GRC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_GRC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR3_GRC_HS[10:0]	0	R/W	Sets the horizontal start position of the graphics image area. Note: Set to 16 or greater clocks and the result of GR3_GRC_HS + GR3_GRC_HW should be smaller than or equal to 2015 clocks.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR3_GRC_HW[10:0]	0	R/W	Sets the horizontal width of the graphics image area. Note: For displaying an image with 1- or 2-pixel horizontal width, set GR3_HW to 2 and GR3_GRC_HW to 1 (1-pixel) or 2 (2-pixel).

Note: All the bits assigned to this address are updated when GR3_P_VEN in GR3_UPDATE is 1.

38.2.34 Alpha Blending Control Register 4 (Graphics 3) (GR3_AB4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_ARC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_ARC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR3_ARC_VS[10:0]	0	R/W	Sets the vertical start position of the valid image area for alpha blending in a rectangular area.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR3_ARC_VW[10:0]	0	R/W	Sets the vertical width of the valid image area for alpha blending in a rectangular area.

Note: This register is updated when GR3_P_VEN in GR3_UPDATE is 1.

38.2.35 Alpha Blending Control Register 5 (Graphics 3) (GR3_AB5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_ARC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_ARC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR3_ARC_HS[10:0]	0	R/W	Sets the horizontal start position of the valid image area for alpha blending in a rectangular area.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR3_ARC_HW[10:0]	0	R/W	Sets the horizontal width of the valid image area for alpha blending in a rectangular area.

Note: This register is updated when GR3_P_VEN in GR3_UPDATE is 1.

38.2.36 Alpha Blending Control Register 6 (Graphics 3) (GR3_AB6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	GR3_ARC_MODE	GR3_ARC_COEF[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	GR3_ARC_RATE[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	GR3_ARC_MODE	0	R/W	Alpha Blending Mode in Rectangular Area 0: Addition 1: Subtraction
23 to 16	GR3_ARC_COEF[7:0]	0	R/W	Sets the alpha coefficient for alpha blending in a rectangular area. (0 to 255) [7:0]: Variation (absolute value)
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	GR3_ARC_RATE[7:0]	0	R/W	Sets the value obtained by subtracting 1 from the frame rate for alpha blending in a rectangular area.

Note: This register is updated when GR3_P_VEN in GR3_UPDATE is 1.

38.2.37 Alpha Blending Control Register 7 (Graphics 3) (GR3_AB7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR3_ARC_DEF[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_CK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR3_ARC_DEF[7:0]	255	R/W	Sets the initial alpha value for alpha blending in a rectangular area.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR3_CK_ON	0	R/W	Turns on/off CLUT-index/RGB-index chroma-key processing. 0: Off 1: On

Note: This register is updated when GR3_P_VEN in GR3_UPDATE is 1.

38.2.38 Alpha Blending Control Register 8 (Graphics 3) (GR3_AB8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_CK_KCLUT[7:0]								GR3_CK_KG[7:0]]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_CK_KB[7:0]								GR3_CK_KR[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR3_CK_ KCLUT[7:0]	0	R/W	CLUT Signal for CLUT-Index Chroma-Key Processing CLUT: Unsigned 8 bits (0 to 255 [LSB])
23 to 16	GR3_CK_ KG[7:0]	0	R/W	G Signal for RGB-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR3_CK_ KB[7:0]	0	R/W	B Signal for RGB-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR3_CK_ KR[7:0]	0	R/W	R Signal for RGB-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR3_P_VEN in GR3_UPDATE is 1.

38.2.39 Alpha Blending Control Register 9 (Graphics 3) (GR3_AB9)

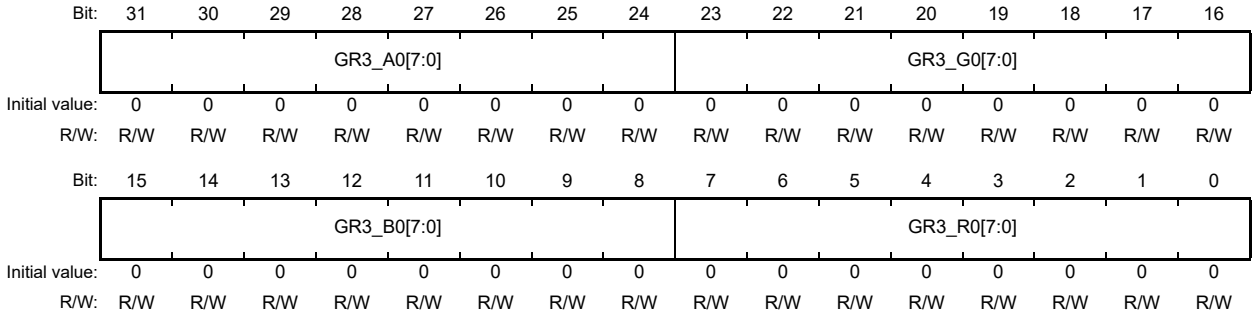
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_CK_A[7:0]								GR3_CK_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_CK_B[7:0]								GR3_CK_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR3_CK_A [7:0]	0	R/W	Replaced Alpha Signal after RGB/CLUT-Index Chroma-Key Processing α: Unsigned 8 bits (0 to 255 [LSB])
23 to 16	GR3_CK_G [7:0]	0	R/W	Replaced G Signal after RGB/CLUT-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR3_CK_B [7:0]	0	R/W	Replaced B Signal after RGB/CLUT-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR3_CK_R [7:0]	0	R/W	Replaced R Signal after RGB/CLUT-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR3_P_VEN in GR3_UPDATE is 1.

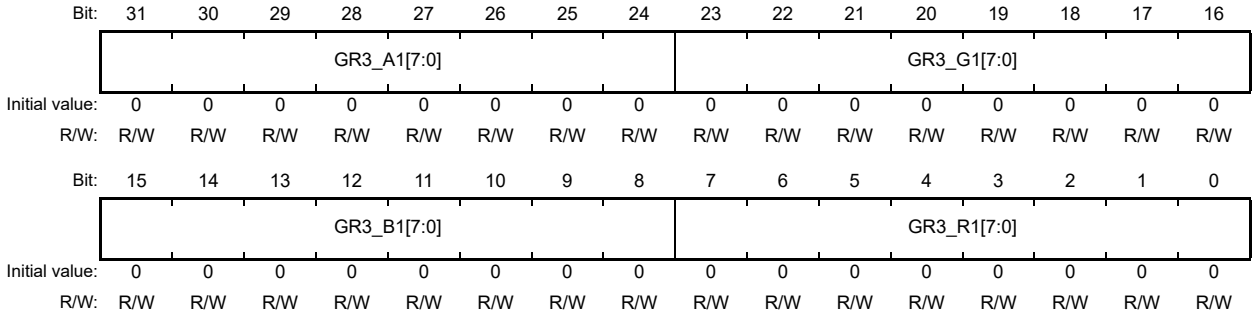
38.2.40 Alpha Blending Control Register 10 (Graphics 3) (GR3_AB10)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR3_A0 [7:0]	0	R/W	CLUT1 α0 Signal Replaced with α signal when in the CLUT1 format and CLUT1 = 0. Replaced with α signal when in the αRGB1555 or RGBα5551 format and α = 0.
23 to 16	GR3_G0 [7:0]	0	R/W	CLUT1 G0 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 0.
15 to 8	GR3_B0 [7:0]	0	R/W	CLUT1 B0 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 0.
7 to 0	GR3_R0 [7:0]	0	R/W	CLUT1 R0 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 0.

Note: This register is updated when GR3_P_VEN in GR3_UPDATE is 1.

38.2.41 Alpha Blending Control Register 11 (Graphics 3) (GR3_AB11)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR3_A1 [7:0]	0	R/W	CLUT1 α1 Signal Replaced with α signal when in the CLUT1 format and CLUT1 = 1. Replaced with α signal when in the αRGB1555 or RGBα5551 format and α = 1.
23 to 16	GR3_G1 [7:0]	0	R/W	CLUT1 G1 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 1.
15 to 8	GR3_B1 [7:0]	0	R/W	CLUT1 B1 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 1.
7 to 0	GR3_R1 [7:0]	0	R/W	CLUT1 R1 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 1.

Note: This register is updated when GR3_P_VEN in GR3_UPDATE is 1.

38.2.42 Background Color Control Register (Graphics 3) (GR3_BASE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR3_BASE_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_BASE_B[7:0]								GR3_BASE_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR3_BASE_G [7:0]	0	R/W	Background Color G Signal G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR3_BASE_B [7:0]	0	R/W	Background Color B Signal B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR3_BASE_R [7:0]	0	R/W	Background Color R Signal R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR3_P_VEN in GR3_UPDATE is 1.

38.2.43 CLUT Table and Interrupt Control Register (Graphics 3) (GR3_CLUT_INT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_CLT_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_LINE[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR3_CLT_ SEL	0	R/W	CLUT Table Select Signal 0: Selects CLUT table 0. The format is converted to αRGB8888 based on the CLUT table 0. CLUT table 1 can be read from or written to by the CPU. 1: Selects CLUT table 1. The format is converted to αRGB8888 based on the CLUT table 1. CLUT table 0 can be read from or written to by the CPU.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR3_LINE [10:0]	0	R/W	Line Interrupt Set When number of lines matches the value of the GR3_LINE bits, an interrupt signal is output. This function is enabled even when the graphics 3 process is not used.

Note: This register is updated when GR3_P_VEN in GR3_UPDATE is 1.

38.2.44 Status Monitor Register (Graphics 3) (GR3_MON)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_LIN_STAT[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_ARC_ST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR3_LIN_STAT[10:0]	0	R	Line Position of Image being Currently Read
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR3_ARC_ST	0	R	Status Flag for Alpha Blending in Rectangular Area 0: Addition or subtraction has been completed. (α value is 0 or 255) 1: Addition or subtraction is in progress.

38.2.45 VIN Synthesizer Register Update Control Register (GR_VIN_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GR_VIN_UPDATE	—	—	—	GR_VIN_P_VEN	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR_VIN_UPDATE	0	R/WC1	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR_VIN_P_VEN	0	R/WC1	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

38.2.46 Alpha Blending Control Register 1 (VIN Synthesizer) (GR_VIN_AB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	GR_VIN_SCL_UND_SEL	GR_VIN_DISP_SEL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	GR_VIN_SCL_UND_SEL	0	R/W	Selection of Lower-Layer Plane in Scaler 0: Selects graphics 0 as lower-layer graphics. 1: Setting prohibited
1, 0	GR_VIN_DISP_SEL[1:0]	0	R/W	Selects the graphics display mode. 0: Background color display (fixed to black) 1: Lower-layer graphics display 2: Setting prohibited 3: Setting prohibited

Note: GR_VIN_SCL_UND_SEL is updated when GR_VIN_UPDATE in GR_VIN_UPDATE is 1. The other bits of this register are updated when GR_VIN_P_VEN in GR_VIN_UPDATE is 1.

38.3 Usage Method

38.3.1 Mute Image

The initial values of the GR0_DISP_SEL[1:0], GR2_DISP_SEL[1:0], GR3_DISP_SEL[1:0], and GR_VIN_DISP_SEL[1:0] bits are all 0. Accordingly, in the initial setting, a background color is displayed both inside and outside the graphics area for the graphics 0, 2, and 3 processes and the VIN synthesizer. Since the default background color is black, the black mute image is displayed in the initial state. Note that the background color for the VIN synthesizer is fixed to black.

38.3.2 Alpha Blending in Rectangular Area

The alpha coefficient and the frame rate can be changed during fade in and fade out by modifying the GR_ARC_MODE, GR_ARC_COEF[7:0] and GR_ARC_RATE[7:0] bits, respectively.

39. Video Display Controller 6 (7): Output Controller

39.1 Output Controller

39.1.1 Overview of Functions

The output controller subjects RGB signals output from the image synthesizer to brightness adjustment, contrast adjustment, gamma correction of individual RGB, dither process, and output format conversion.
RBG signals can be output from the LVDS. For output from the LVDS, see section 41, LVDS Output Interface.
Figure 39.1 shows the function block diagram of the output controller.

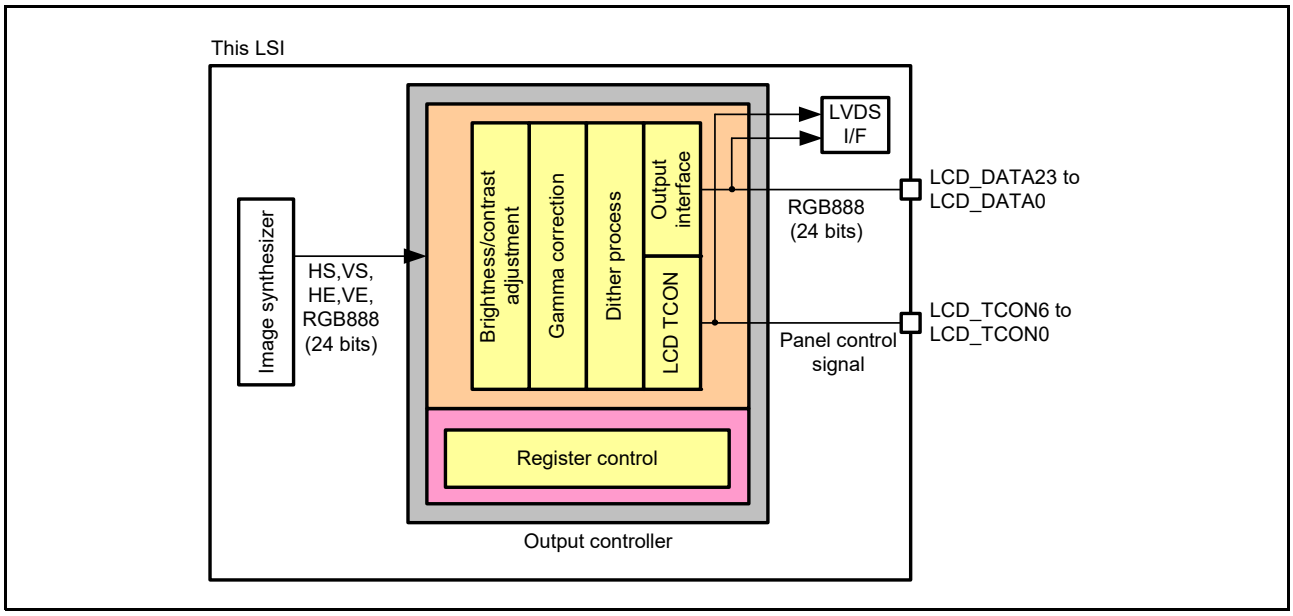


Figure 39.1 Functional Block Diagram of Output Controller

39.1.2 Register Update Control

The Vsync signal is used to control the update timing of all the registers of the output controller.

After 1 is set to the bits in the update control register, the contents of the relevant registers are actually modified at the rising edge of the Vsync signal, when the update control register is automatically cleared to 0.

Table 39.1 Register Update Control

Register Name	Bit Name	Initial Value	Description
OUT_UPDATE	OUTCNT_VEN	0	Brightness/Contrast Control, Dither Process, Output Interface Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GAM_G_UPDATE	GAM_G_VEN	0	Gamma Correction (G) Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GAM_B_UPDATE	GAM_B_VEN	0	Gamma Correction (B) Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GAM_R_UPDATE	GAM_R_VEN	0	Gamma Correction (R) Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
TCON_UPDATE	TCON_VEN	0	LCD TCON Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

39.1.3 Route Selection

The processing sequence of the brightness/contrast control and gamma correction control can be swapped according to the settings of the register.

Table 39.2 Route Selection

Register Name	Bit Name	Initial Value	Description
OUT_CLK_PHASE	OUTCNT_ FRONT_GAM	0	Correction Circuit Sequence Control 0: Brightness → contrast → gamma correction 1: Gamma correction → brightness → contrast

39.1.4 Panel Brightness Adjustment

Brightness (DC) adjustment is individually performed for RGB signals from the image synthesizer.

(BRT_R/G/BOUT after brightness adjustment has many bits to prevent overflow or underflow. The overflow or underflow process is performed at contrast calculation.)

(1) Calculation formulas for brightness (DC) adjustment

BRT_GOUT = GIN + PBRT_G − 512
BRT_BOUT = BIN + PBRT_B − 512
BRT_ROUT = RIN + PBRT_R − 512

Table 39.3 Brightness (DC) Adjustment

Register Name	Bit Name	Initial Value	Description
OUT_BRIGHT1	PBRT_G[9:0]	512	Brightness (DC) Adjustment of G Signal Unsigned (0 (-512) to 512 (0) to 1023 (+511) [LSB], 512 [LSB] with offset)
OUT_BRIGHT2	PBRT_B[9:0]	512	Brightness (DC) Adjustment of B Signal Unsigned (0 (-512) to 512 (0) to 1023 (+511) [LSB], 512 [LSB] with offset)
OUT_BRIGHT2	PBRT_R[9:0]	512	Brightness (DC) Adjustment of R Signal Unsigned (0 (-512) to 512 (0) to 1023 (+511) [LSB], 512 [LSB] with offset)

39.1.5 Contrast Adjustment

Contrast is calculated for RGB signals obtained after brightness calculation.

(If an overflow or underflow occurs, contrast is clipped to the maximum or minimum value.)

(1) Calculation formulas for contrast (gain) adjustment

GOUT = BRT_GOUT × CONT_G/128
BOUT = BRT_BOUT × CONT_B/128
ROUT = BRT_ROUT × CONT_R/128

Table 39.4 Contrast (Gain) Adjustment

Register Name	Bit Name	Initial Value	Description
OUT_CONTRAST	CONT_G [7:0]	128	Contrast (Gain) Adjustment of G Signal 0/128 to 255/128 (approx.2 times)
OUT_CONTRAST	CONT_B [7:0]	128	Contrast (Gain) Adjustment of B Signal 0/128 to 255/128 (approx.2 times)
OUT_CONTRAST	CONT_R [7:0]	128	Contrast (Gain) Adjustment of R Signal 0/128 to 255/128 (approx.2 times)

39.1.6 Gamma Correction

Gamma correction is carried out by dividing an input signal having 256 gradation levels into 32 and controlling the gain of each area. Gain coefficient of each area can be set as 0 to approx. 2.0 [times]

(1) Gamma correction formula for each area

$DOUT = ((DIN - TH_{(n)}) \times GAIN_{(n)} + OFFSET_{(n)})/256$

DIN: Input signal (8-bit)
DOUT: Output signal (10-bit)
$TH_{(n)}$: Threshold (8-bit)
$OFFSET_{(n)}$: Offset value (19-bit)
$GAIN_{(n)}$: Gain coefficient (11-bit)

(2) Offset calculation formulas for each area

$OFFSET_{(n)} = OFFSET_{(n-1)} + DEF\_O_{(n)}$ (When n = 0, $OFFSET_{(0)} = 0$.)
$DEF\_O_{(n)} = (TH_{(n)} - TH_{(n-1)}) \times GAIN_{(n-1)}$ (When n = 0, $OFFSET_{(0)} = 0$.)

$OFFSET_{(n)}$: Offset value of current area (19-bit)
$OFFSET_{(n-1)}$: Offset value of previous area (19-bit)
$DEF\_O_{(n)}$: Difference in offset value of Current and previous area (19-bit)
$TH_{(n)}$: Threshold of current area (8-bit)
$TH_{(n-1)}$: Threshold of previous area (8-bit)
$GAIN_{(n-1)}$: Gain coefficient of previous area (11-bit)

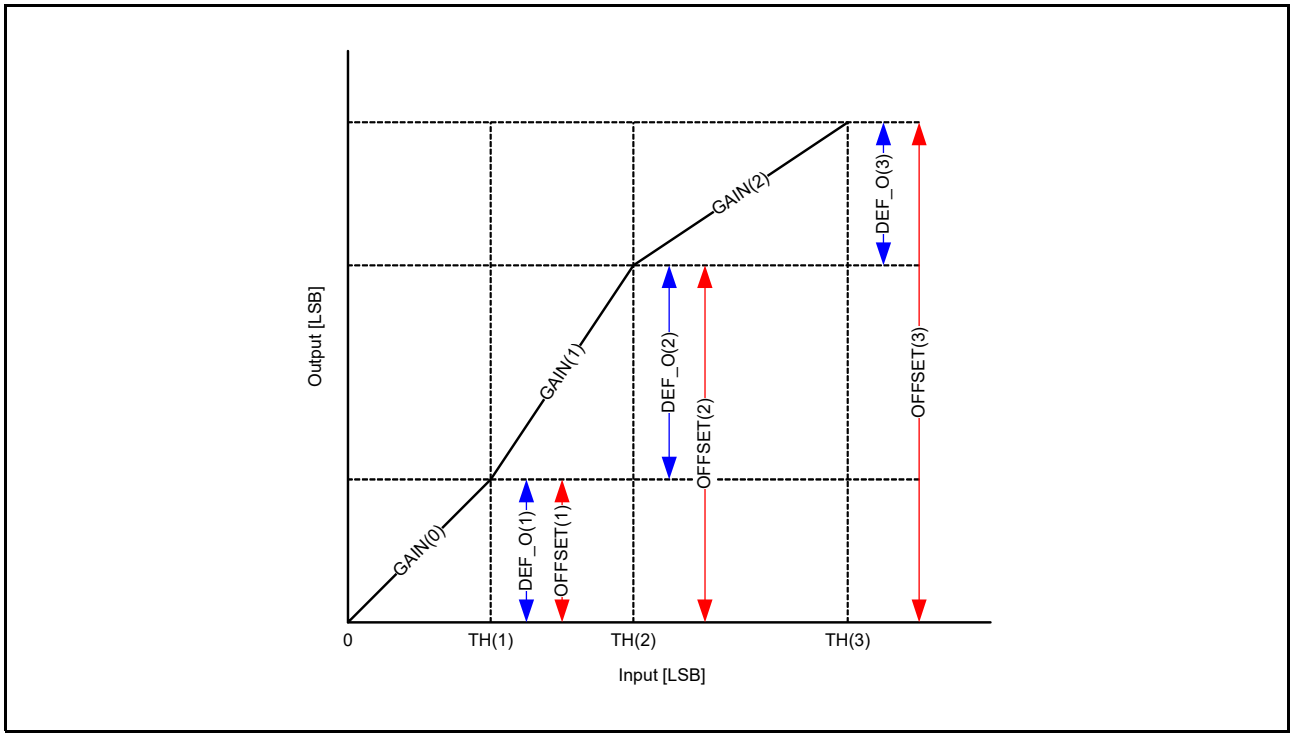


Figure 39.2 Corresponding Chart of Offset Calculation Formulas

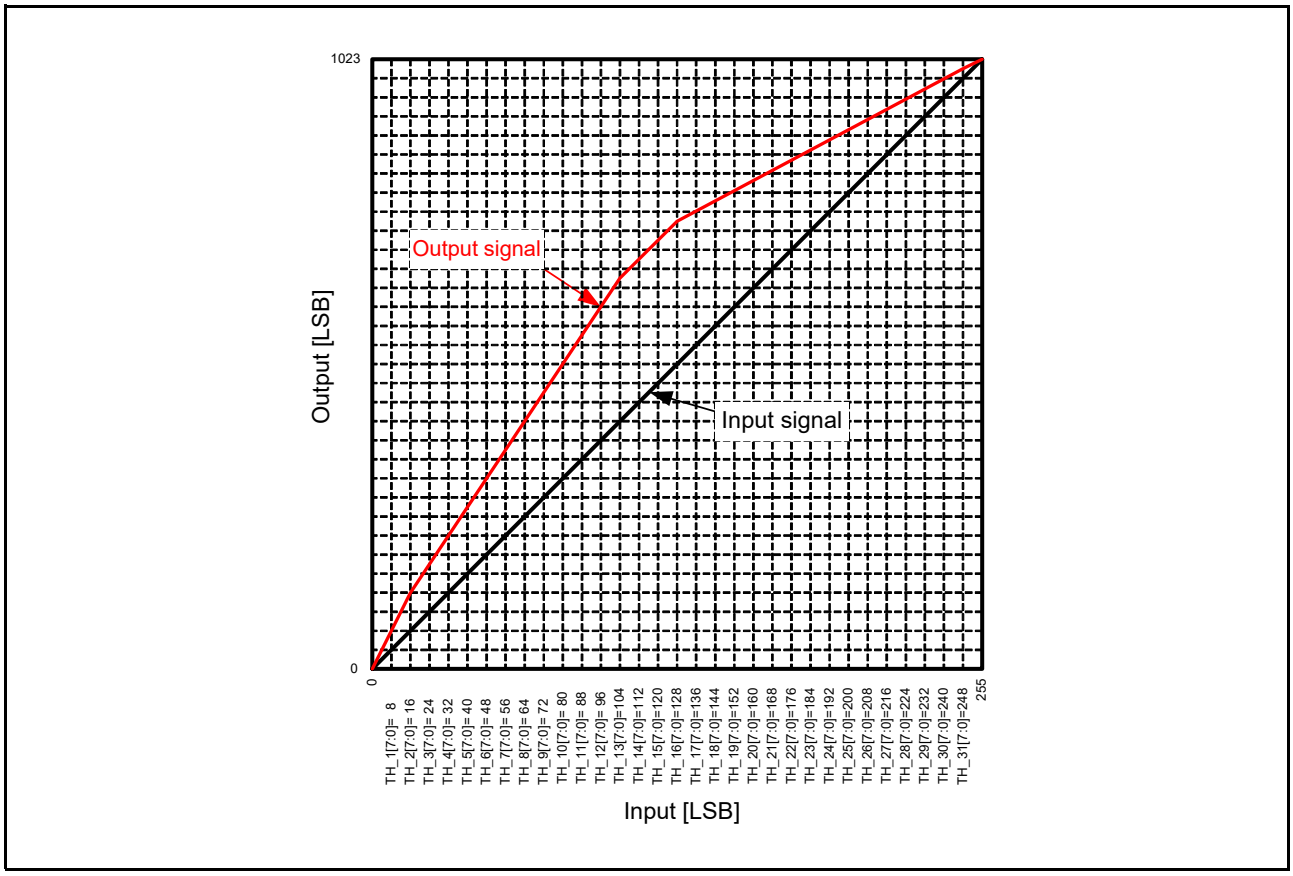


Figure 39.3 Example of Input-Output Characteristics of Gamma Correction

Table 39.5 Gamma Correction

Register Name	Bit Name	Initial Value	Description
GAM_SW	GAM_ON	0	Gamma Correction On/Off Control 0: Off 1: On
GAM_G_AREA1 to GAM_G_AREA8	GAM_G_TH_01 to GAM_G_TH_31 [7:0]	*	Start Threshold of Area 1 to 31 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area*1 < Threshold of current area < Threshold of next area*2 *1: GAM_G_TH_01 is 0 *2: GAM_G_TH_31 is ≤ 255 *Initial Value GAM_G_TH_01:8, GAM_G_TH_02:16, GAM_G_TH_03:24, GAM_G_TH_04:32, GAM_G_TH_05:40, GAM_G_TH_06:48, GAM_G_TH_07:56, GAM_G_TH_08:64, GAM_G_TH_09:72, GAM_G_TH_10:80 GAM_G_TH_11:88, GAM_G_TH_12:96, GAM_G_TH_13:104, GAM_G_TH_14:112, GAM_G_TH_15:120, GAM_G_TH_16:128, GAM_G_TH_17:136, GAM_G_TH_18:144, GAM_G_TH_19:152, GAM_G_TH_20:160, GAM_G_TH_21:168, GAM_G_TH_22:176, GAM_G_TH_23:184, GAM_G_TH_24:192, GAM_G_TH_25:200, GAM_G_TH_26:208, GAM_G_TH_27:216, GAM_G_TH_28:224, GAM_G_TH_29:232, GAM_G_TH_30:240, GAM_G_TH_31:248

Table 39.5 Gamma Correction

Register Name	Bit Name	Initial Value	Description
GAM_G_LUT1 to GAM_G_LUT16	GAM_G_GAIN_00 to GAM_G_GAIN_31 [10:0]	1024	Gain Adjustment of Area 0 to 31 of G Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times])
GAM_B_AREA1 to GAM_B_AREA8	GAM_B_TH_01 to GAM_B_TH_31 [7:0]	*	Start Threshold of Area 1 to 31 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area*1 < Threshold of current area < Threshold of next area*2 *1: GAM_B_TH_01 is 0 *2: GAM_B_TH_31 is ≤ 255 *Initial Value GAM_B_TH_01:8, GAM_B_TH_02:16, GAM_B_TH_03:24, GAM_B_TH_04:32, GAM_B_TH_05:40, GAM_B_TH_06:48, GAM_B_TH_07:56, GAM_B_TH_08:64, GAM_B_TH_09:72, GAM_B_TH_10:80 GAM_B_TH_11:88, GAM_B_TH_12:96, GAM_B_TH_13:104, GAM_B_TH_14:112, GAM_B_TH_15:120, GAM_B_TH_16:128, GAM_B_TH_17:136, GAM_B_TH_18:144, GAM_B_TH_19:152, GAM_B_TH_20:160, GAM_B_TH_21:168, GAM_B_TH_22:176, GAM_B_TH_23:184, GAM_B_TH_24:192, GAM_B_TH_25:200, GAM_B_TH_26:208, GAM_B_TH_27:216, GAM_B_TH_28:224, GAM_B_TH_29:232, GAM_B_TH_30:240, GAM_B_TH_31:248
GAM_B_LUT1 to GAM_B_LUT16	GAM_B_GAIN_00 to GAM_B_GAIN_31 [10:0]	1024	Gain Adjustment of Area 0 to 31 of B Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times])
GAM_R_AREA1 to GAM_R_AREA8	GAM_R_TH_01 to GAM_R_TH_31 [7:0]	*	Start Threshold of Area 1 to 31 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area*1 < Threshold of current area < Threshold of next area*2 *1: GAM_R_TH_01 is 0 *2: GAM_R_TH_31 is ≤ 255 *Initial Value GAM_R_TH_01:8, GAM_R_TH_02:16, GAM_R_TH_03:24, GAM_R_TH_04:32, GAM_R_TH_05:40, GAM_R_TH_06:48, GAM_R_TH_07:56, GAM_R_TH_08:64, GAM_R_TH_09:72, GAM_R_TH_10:80 GAM_R_TH_11:88, GAM_R_TH_12:96, GAM_R_TH_13:104, GAM_R_TH_14:112, GAM_R_TH_15:120, GAM_R_TH_16:128, GAM_R_TH_17:136, GAM_R_TH_18:144, GAM_R_TH_19:152, GAM_R_TH_20:160, GAM_R_TH_21:168, GAM_R_TH_22:176, GAM_R_TH_23:184, GAM_R_TH_24:192, GAM_R_TH_25:200, GAM_R_TH_26:208, GAM_R_TH_27:216, GAM_R_TH_28:224, GAM_R_TH_29:232, GAM_R_TH_30:240, GAM_R_TH_31:248
GAM_R_LUT1 to GAM_R_LUT16	GAM_R_GAIN_00 to GAM_R_GAIN_31[10:0]	1024	Gain Adjustment of Area 0 to 31 of R Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times])

39.1.7 Dither Process

Dither process is carried out by adjusting brightness/contrast or reducing 10-bit RGB signals output from the gamma correction block to 8-bit, 6-bit, or 5-bit RGB signals. The operation mode of dither process can be selected from truncate mode, round-off mode, 2 × 2 pattern dither mode and random pattern dither mode.

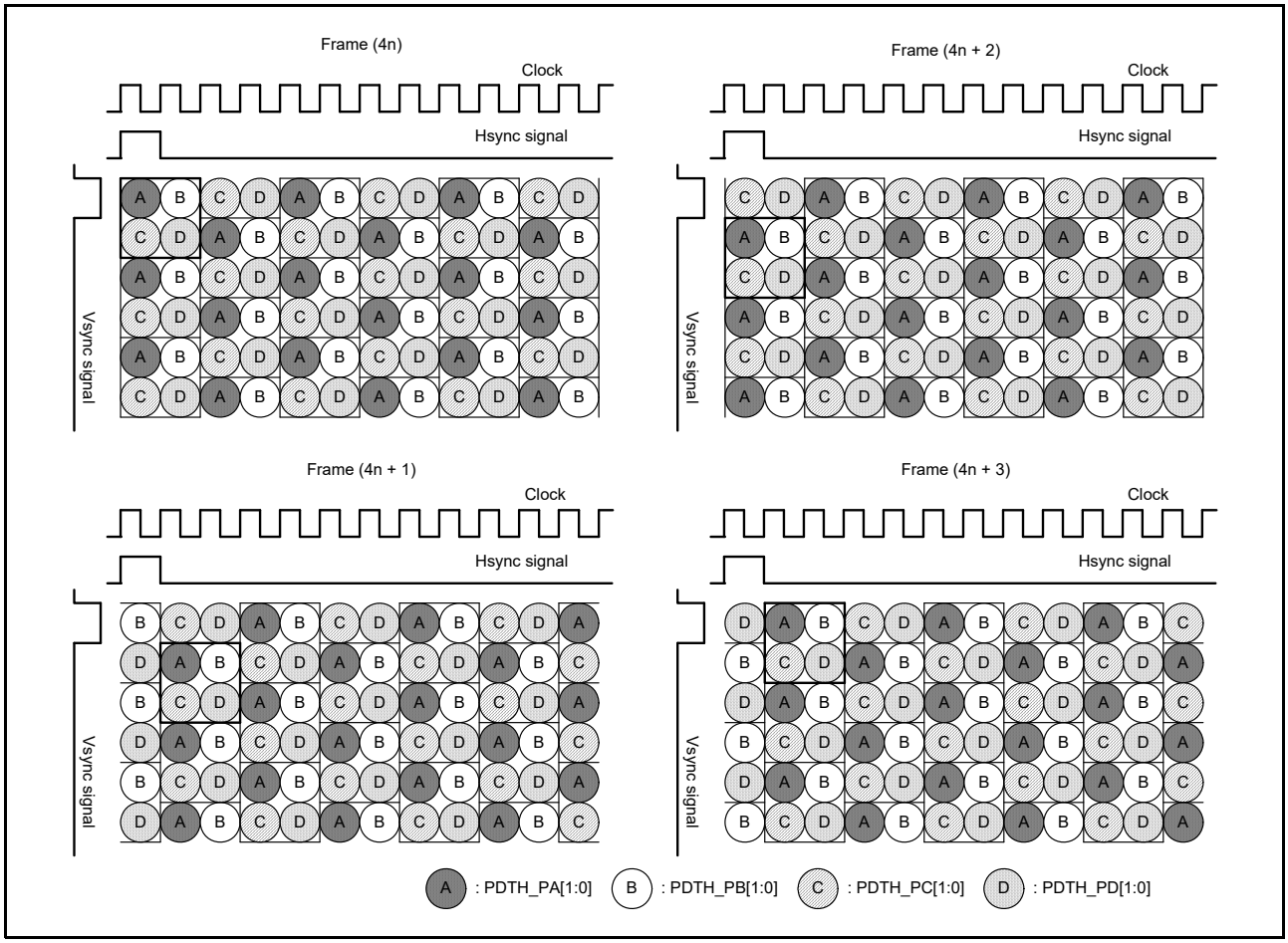


Figure 39.4 Operation Specification of 2 × 2 Pattern Dither

The conversion equations are as follows.

[Truncate mode]

(a) 10 bits to 8 bits

Output RGB data[7:0] = Input RGB data[9:0] ÷ 4 (truncate the number below the decimal point)

(b) 10 bits to 6 bits

Output RGB data[7:2] = Input RGB data[9:0] ÷ 16 (truncate the number below the decimal point)

(c) 10 bits to 5 bits

Output RGB data[7:3] = Input RGB data[9:0] ÷ 32 (truncate the number below the decimal point)

[Round-off mode]

(a) 10 bits to 8 bits

Output RGB data[7:0] = Input RGB data[9:0] ÷ 4 (round off to an integer)

(b) 10 bits to 6 bits

Output RGB data[7:2] = Input RGB data[9:0] ÷ 16 (round off to an integer)

(c) 10 bits to 5 bits

Output RGB data[7:3] = Input RGB data[9:0] ÷ 32 (round off to an integer)

[2 × 2 pattern dither mode, random pattern dither mode]

(a) 10 bits to 8 bits

Output RGB data[7:0] = Input RGB data[9:0] ÷ 4 + pattern value at the first decimal place (truncate the number below the decimal point after addition)

(b) 10 bits to 6 bits

Output RGB data[7:2] = Input RGB data[9:0] ÷ 16 + pattern value at the first decimal place (truncate the number below the decimal point after addition)

(c) 10 bits to 5 bits

Output RGB data[7:3] = Input RGB data[9:0] ÷ 32 + pattern value at the first decimal place (truncate the number below the decimal point after addition)

Table 39.6 Panel Dither Correction

Register Name	Bit Name	Initial Value	Description
OUT_PDTHA	PDTH_SEL[1:0]	0	Panel Dither Operation Mode 0: Truncate 1: Round-off 2: 2 × 2 pattern dither 3: Random pattern dither
OUT_PDTHA	PDTH_FORMAT[1:0]	0	Panel Dither Output Format Select 0: RGB888 1: RGB666 2: RGB565 3: Setting prohibited
OUT_PDTHA	PDTH_PA[1:0]	3	Pattern Value (A) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])
OUT_PDTHA	PDTH_PB[1:0]	0	Pattern Value (B) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])
OUT_PDTHA	PDTH_PC[1:0]	2	Pattern Value (C) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])
OUT_PDTHA	PDTH_PD[1:0]	1	Pattern Value (D) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])

39.1.8 Output Format Conversion

In output format conversion, the RGB signal after dither process is converted to LCD output signal having any of the following formats, namely, parallel RGB888, parallel RGB666, parallel RGB565, and serial RGB.

Further, converted data can be allocated to LCD output pins as selected.

(1) Bit Allocation of LCD Signals for RGB888 Output

Table 39.7 shows the RGB signal input allocated to the LCD signal output for RGB888 output.

R/G/BIN[7:0] are the RGB internal signals after dither process.

Table 39.7 Bit Allocation of RGB Signal Input for RGB888 Output

OUT_FORMAT	0	0	0	0
OUT_ENDIAN_ON	0	0	1	1
OUT_SWAP_ON	0	1	0	1
LCD_DATA23	RIN[7]	BIN[7]	RIN[0]	BIN[0]
LCD_DATA22	RIN[6]	BIN[6]	RIN[1]	BIN[1]
LCD_DATA21	RIN[5]	BIN[5]	RIN[2]	BIN[2]
LCD_DATA20	RIN[4]	BIN[4]	RIN[3]	BIN[3]
LCD_DATA19	RIN[3]	BIN[3]	RIN[4]	BIN[4]
LCD_DATA18	RIN[2]	BIN[2]	RIN[5]	BIN[5]
LCD_DATA17	RIN[1]	BIN[1]	RIN[6]	BIN[6]
LCD_DATA16	RIN[0]	BIN[0]	RIN[7]	BIN[7]
LCD_DATA15	GIN[7]	GIN[7]	GIN[0]	GIN[0]
LCD_DATA14	GIN[6]	GIN[6]	GIN[1]	GIN[1]
LCD_DATA13	GIN[5]	GIN[5]	GIN[2]	GIN[2]
LCD_DATA12	GIN[4]	GIN[4]	GIN[3]	GIN[3]
LCD_DATA11	GIN[3]	GIN[3]	GIN[4]	GIN[4]
LCD_DATA10	GIN[2]	GIN[2]	GIN[5]	GIN[5]
LCD_DATA9	GIN[1]	GIN[1]	GIN[6]	GIN[6]
LCD_DATA8	GIN[0]	GIN[0]	GIN[7]	GIN[7]
LCD_DATA7	BIN[7]	RIN[7]	BIN[0]	RIN[0]
LCD_DATA6	BIN[6]	RIN[6]	BIN[1]	RIN[1]
LCD_DATA5	BIN[5]	RIN[5]	BIN[2]	RIN[2]
LCD_DATA4	BIN[4]	RIN[4]	BIN[3]	RIN[3]
LCD_DATA3	BIN[3]	RIN[3]	BIN[4]	RIN[4]
LCD_DATA2	BIN[2]	RIN[2]	BIN[5]	RIN[5]
LCD_DATA1	BIN[1]	RIN[1]	BIN[6]	RIN[6]
LCD_DATA0	BIN[0]	RIN[0]	BIN[7]	RIN[7]

(2) Bit Allocation of LCD Signal for RGB666 Output

Table 39.8 shows the RGB signal input allocated to the LCD signal output for RGB666 output.

R/G/BIN[7:0] are the RGB internal signals after dither process.

Table 39.8 Bit Allocation of RGB Signal Input for RGB666 Output

OUT_FORMAT	1	1	1	1
OUT_ENDIAN_ON	0	0	1	1
OUT_SWAP_ON	0	1	0	1
LCD_DATA23	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA22	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA21	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA20	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA19	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA18	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA17	RIN[7]	BIN[7]	RIN[2]	BIN[2]
LCD_DATA16	RIN[6]	BIN[6]	RIN[3]	BIN[3]
LCD_DATA15	RIN[5]	BIN[5]	RIN[4]	BIN[4]
LCD_DATA14	RIN[4]	BIN[4]	RIN[5]	BIN[5]
LCD_DATA13	RIN[3]	BIN[3]	RIN[6]	BIN[6]
LCD_DATA12	RIN[2]	BIN[2]	RIN[7]	BIN[7]
LCD_DATA11	GIN[7]	GIN[7]	GIN[2]	GIN[2]
LCD_DATA10	GIN[6]	GIN[6]	GIN[3]	GIN[3]
LCD_DATA9	GIN[5]	GIN[5]	GIN[4]	GIN[4]
LCD_DATA8	GIN[4]	GIN[4]	GIN[5]	GIN[5]
LCD_DATA7	GIN[3]	GIN[3]	GIN[6]	GIN[6]
LCD_DATA6	GIN[2]	GIN[2]	GIN[7]	GIN[7]
LCD_DATA5	BIN[7]	RIN[7]	BIN[2]	RIN[2]
LCD_DATA4	BIN[6]	RIN[6]	BIN[3]	RIN[3]
LCD_DATA3	BIN[5]	RIN[5]	BIN[4]	RIN[4]
LCD_DATA2	BIN[4]	RIN[4]	BIN[5]	RIN[5]
LCD_DATA1	BIN[3]	RIN[3]	BIN[6]	RIN[6]
LCD_DATA0	BIN[2]	RIN[2]	BIN[7]	RIN[7]

(3) Bit Allocation of LCD Signal for RGB565 Output

Table 39.9 shows the RGB signal input allocated to the LCD signal output for RGB565 output.

R/G/BIN[7:0] are the RGB internal signals after dither process.

Table 39.9 Bit Allocation of RGB Signal Input for RGB565 Output

OUT_FORMAT	2	2	2	2
OUT_ENDIAN_ON	0	0	1	1
OUT_SWAP_ON	0	1	0	1
LCD_DATA23	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA22	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA21	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA20	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA19	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA18	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA17	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA16	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA15	RIN[7]	BIN[7]	RIN[3]	BIN[3]
LCD_DATA14	RIN[6]	BIN[6]	RIN[4]	BIN[4]
LCD_DATA13	RIN[5]	BIN[5]	RIN[5]	BIN[5]
LCD_DATA12	RIN[4]	BIN[4]	RIN[6]	BIN[6]
LCD_DATA11	RIN[3]	BIN[3]	RIN[7]	BIN[7]
LCD_DATA10	GIN[7]	GIN[7]	GIN[2]	GIN[2]
LCD_DATA9	GIN[6]	GIN[6]	GIN[3]	GIN[3]
LCD_DATA8	GIN[5]	GIN[5]	GIN[4]	GIN[4]
LCD_DATA7	GIN[4]	GIN[4]	GIN[5]	GIN[5]
LCD_DATA6	GIN[3]	GIN[3]	GIN[6]	GIN[6]
LCD_DATA5	GIN[2]	GIN[2]	GIN[7]	GIN[7]
LCD_DATA4	BIN[7]	RIN[7]	BIN[3]	RIN[3]
LCD_DATA3	BIN[6]	RIN[6]	BIN[4]	RIN[4]
LCD_DATA2	BIN[5]	RIN[5]	BIN[5]	RIN[5]
LCD_DATA1	BIN[4]	RIN[4]	BIN[6]	RIN[6]
LCD_DATA0	BIN[3]	RIN[3]	BIN[7]	RIN[7]

(4) Bit Allocation of LCD Signal for Serial RGB Output

For serial RGB output, RGB signal input shown Table 39.10 is allocated to rgb internal signals and the signals are converted from parallel to serial format and output as LCD signals. R/G/BIN[7:0] are the RGB internal signals after dither process.

The internal signals r[7:0], g[7:0], and b[7:0] are serially output to LCD_DATA7 to LCD_DATA0.

Table 39.10 Bit Allocation of RGB Signal Input for Serial RGB Output

OUT_FORMAT	3	3	3	3
OUT_ENDIAN_ON	0	0	1	1
OUT_SWAP_ON	0	1	0	1
r[7]	RIN[7]	BIN[7]	RIN[0]	BIN[0]
r[6]	RIN[6]	BIN[6]	RIN[1]	BIN[1]
r[5]	RIN[5]	BIN[5]	RIN[2]	BIN[2]
r[4]	RIN[4]	BIN[4]	RIN[3]	BIN[3]
r[3]	RIN[3]	BIN[3]	RIN[4]	BIN[4]
r[2]	RIN[2]	BIN[2]	RIN[5]	BIN[5]
r[1]	RIN[1]	BIN[1]	RIN[6]	BIN[6]
r[0]	RIN[0]	BIN[0]	RIN[7]	BIN[7]
g[7]	GIN[7]	GIN[7]	GIN[0]	GIN[0]
g[6]	GIN[6]	GIN[6]	GIN[1]	GIN[1]
g[5]	GIN[5]	GIN[5]	GIN[2]	GIN[2]
g[4]	GIN[4]	GIN[4]	GIN[3]	GIN[3]
g[3]	GIN[3]	GIN[3]	GIN[4]	GIN[4]
g[2]	GIN[2]	GIN[2]	GIN[5]	GIN[5]
g[1]	GIN[1]	GIN[1]	GIN[6]	GIN[6]
g[0]	GIN[0]	GIN[0]	GIN[7]	GIN[7]
b[7]	BIN[7]	RIN[7]	BIN[0]	RIN[0]
b[6]	BIN[6]	RIN[6]	BIN[1]	RIN[1]
b[5]	BIN[5]	RIN[5]	BIN[2]	RIN[2]
b[4]	BIN[4]	RIN[4]	BIN[3]	RIN[3]
b[3]	BIN[3]	RIN[3]	BIN[4]	RIN[4]
b[2]	BIN[2]	RIN[2]	BIN[5]	RIN[5]
b[1]	BIN[1]	RIN[1]	BIN[6]	RIN[6]
b[0]	BIN[0]	RIN[0]	BIN[7]	RIN[7]

(5) Parallel to Serial Conversion

As shown in Table 39.11, four types of parallel to serial conversions are possible by controlling clock speed mode and selecting the scan direction ('n' in the table are natural numbers).

Table 39.11 Specifications of Serial RGB Output

OUT_FRQ_SEL	1	1	2	2
OUT_DIR_SEL	0	1	0	1
Line (2n-1)	Repeated (r → g → b)	Repeated (b → g → r)	Repeated (r → g → b → X)	Repeated (X → b → g → r)
Line 2n	Repeated (g → b → r)	Repeated (r → b → g)	Repeated (r → g → b → X)	Repeated (X → b → g → r)

Figure 39.5 and Figure 39.6 show the timing of parallel to serial conversion in triple speed and quadruple speed modes, respectively.

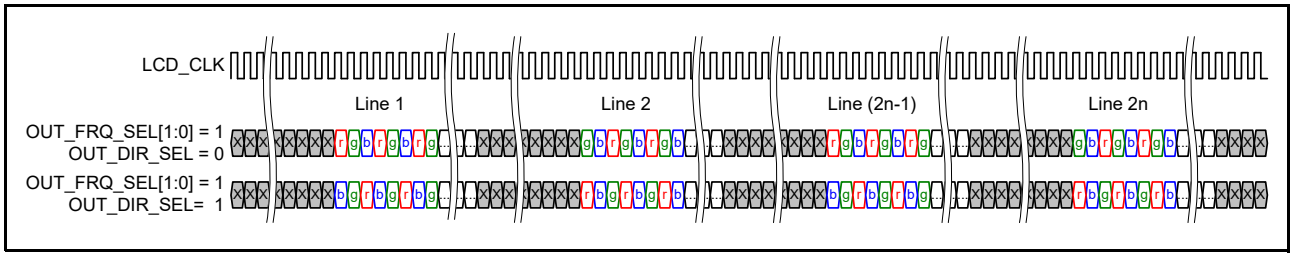


Figure 39.5 Timing of Parallel to Serial Conversion in Triple Speed Mode

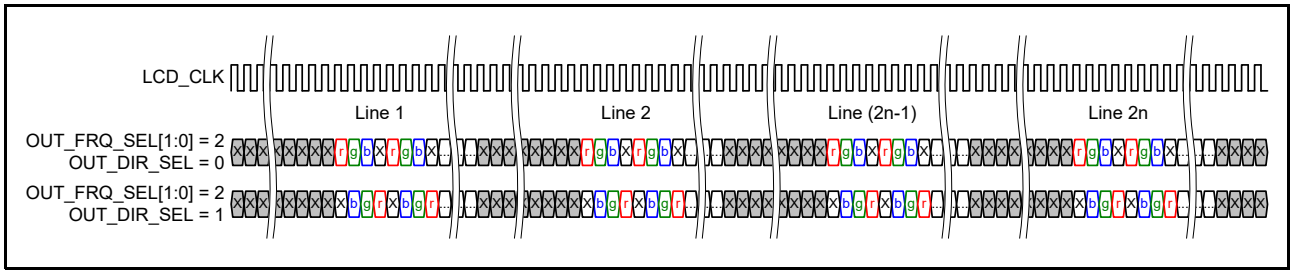


Figure 39.6 Timing of Parallel to Serial Conversion in Quadruple Speed Mode

During serial output, the phase timing with the HE signal can be adjusted by OUT_PHASE[0:1].

Figure 39.7 shows the timing of the clock phases of the serial RGB output (triple speed mode).

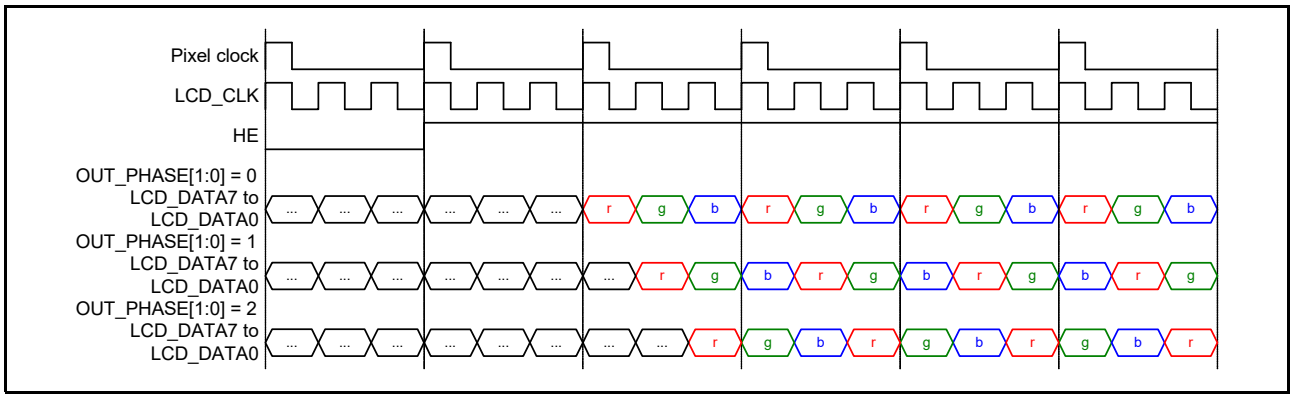


Figure 39.7 Timing of Clock Phases of Serial RGB Output (Triple Speed Mode)

Figure 39.8 shows the timing of the clock phases of the serial RGB output (quadruple speed mode).

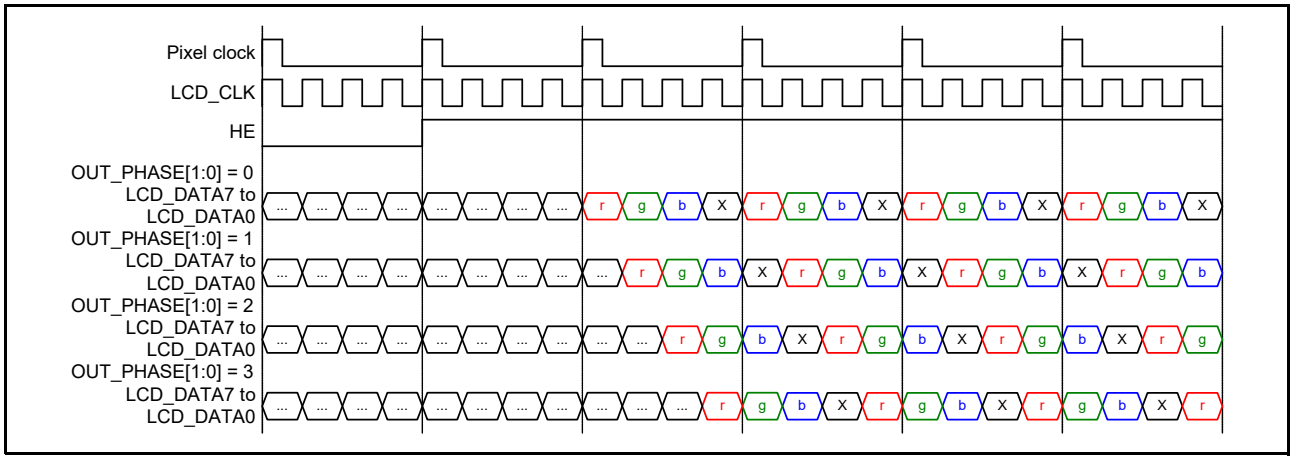


Figure 39.8 Timing of Clock Phases of Serial RGB Output (Quadruple Speed Mode)

Table 39.12 Output Format Conversion

Register Name	Bit Name	Initial Value	Description
OUT_SET	OUT_FORMAT[1:0]	0	Output Format Select 0: RBG888 1: RGB666 2: RGB565 3: Serial RGB
OUT_SET	OUT_ENDIAN_ON	0	Bit Endian Change On/Off Control 0: Off 1: On
OUT_SET	OUT_SWAP_ON	0	B/R Signal Swap On/Off Control 0: Off 1: On
OUT_SET	OUT_FRQ_SEL[1:0]	0	Clock Frequency Control 0: 100% speed — (parallel RGB) 1: Triple speed — (serial RGB) 2: Quadruple speed — (serial RGB) 3: Setting prohibited
OUT_SET	OUT_DIR_SEL	0	Scan Direction Select 0: Forward scan 1: Reverse scan
OUT_SET	OUT_PHASE[1:0]	0	Clock Phase Adjustment for Serial RGB Output Triple speed mode 0: 0 (clk) 1: 1 (clk) 2: 2 (clk) 3: Setting prohibited Quadruple speed mode 0: 0 (clk) 1: 1 (clk) 2: 2 (clk) 3: 3 (clk)

39.1.9 LCD TCON

The LCD TCON generates various timing signals for driving the LCD panel.

Specifically, the timing include two vertical panel driver signals, five horizontal panel driver signals, and one composite signal of the vertical and horizontal panel driver signals. Table 39.13 lists the timing signals that are generated by LCD TCON

Table 39.13 Signals Generated by LCD TCON

Signal Name	Type	Description
STVA/VS	Vertical	• Gate start signal The pulse width, pulse position, and pulse polarity of the signal can be controlled. • Vsync signal The width, position, and polarity of the sync signal can be controlled.
STVB/VE	Vertical	• Gate start signal The pulse width, pulse position, and pulse polarity of the signal can be controlled. • Vertical enable signal The width, position, and polarity of the sync signal can be controlled.
STH/SP/HS	Horizontal	• Source start signal The pulse width, pulse position, and pulse polarity of the signal can be controlled. • Hsync signal The width, position, and polarity of the sync signal can be controlled.
STB/LP/HE	Horizontal	• Source strobe signal The pulse width, pulse position, and pulse polarity of the signal can be controlled. • Horizontal enable signal The width, position, and polarity of the enable signal can be controlled.
CPV/GCK	Horizontal	• Gate clock signal The pulse width, pulse position, and pulse polarity of the signal can be controlled.
POLA	Horizontal	• VCOM voltage polarity control signal The polarity inversion position, and polarity inversion operation (1 × 1, 1 × 2, 2 × 2) can be controlled.
POLB	Horizontal	• VCOM voltage polarity control signal The polarity inversion position, and polarity inversion operation (1 × 1, 1 × 2, 2 × 2) can be controlled.
DE	Horizontal/Vertical	• Data enable signal The width, position, and polarity of the enable signal can be controlled.

(1) Horizontal Reference Offset Control

The horizontal reference offset control enables generation of a reference signal with a clock delay equivalent to the value of TCON_OFFSET[10:0] from the rising edge of the Hsync signal. If a signal that spans across the Hsync signal needs to be generated, such a signal is generated with reference to the offset reference signal.

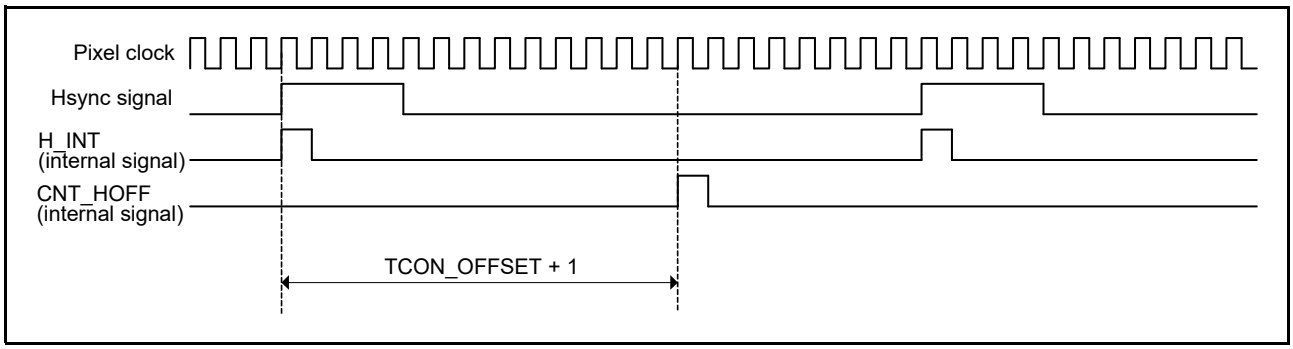


Figure 39.9 Generation of Offset Horizontal Reference (H_OFF) Signal

Table 39.14 Horizontal Reference Signal Selection

Signal Name	Bit Name	Initial Value	Description
TCON_TIM	TCON_OFFSET[10:0]	0	Offset Hsync Signal Timing Sets the clock cycle count from the rising edge of the Hsync signal.
TCON_TIM_STH2	TCON_STH_HS_SEL	0	STH Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
TCON_TIM_STB2	TCON_STB_HS_SEL	0	STB Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
TCON_TIM_CPV2	TCON_CPV_HS_SEL	0	CPV Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
TCON_TIM_POLA2	TCON_POLA_HS_SEL	0	POLA Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
TCON_TIM_POLB2	TCON_POLB_HS_SEL	0	POLB Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference

Note: When generating the POLA and POLB signals in reverse mode, the bits TCON_POLA_HS_SEL and TCON_POLB_HS_SEL should be set to 0.

(2) Horizontal Panel Driver Signal Generation (A)

Horizontal synchronous panel driver signal generation (A) involves generation of a timing signal that changes twice in a horizontal period according to the values of TCON_xxxx_HS[10:0] and TCON_xxxx_HW[10:0] bits, which set the first changing timing and the second changing timing, respectively.

The internal counter performs the following operations.

1. Resets the counter value at the rising edge of the Hsync signal as the reference.
2. Increments the counter value at the rising edge of the panel clock.

A fixed output value of 0 can be obtained by setting 0 in TCON_xxxx_HW[10:0], which set the second changing timing.

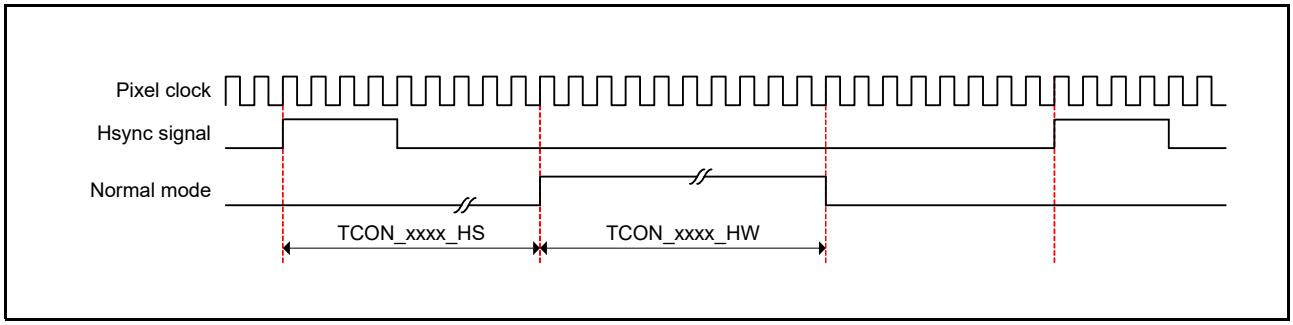


Figure 39.10 Horizontal Panel Driver Signal (in Normal Mode)

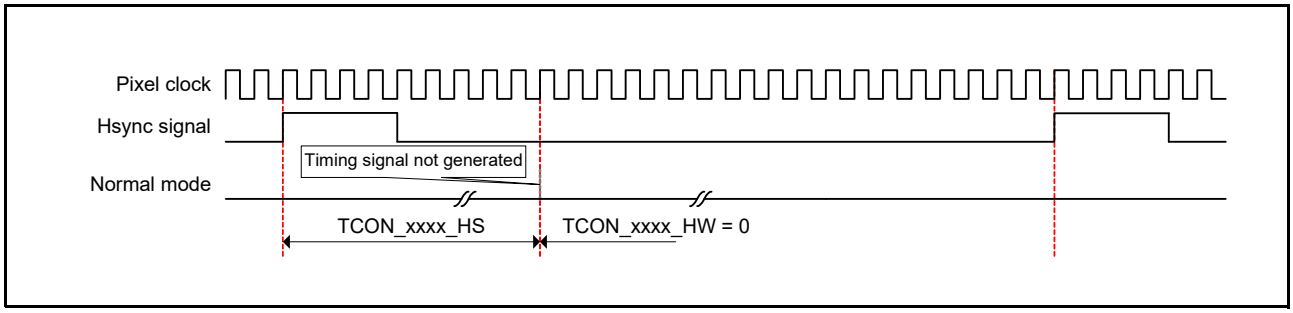


Figure 39.11 Horizontal Panel Driver Signal (in Normal Mode and When TCON_xxxx_HW_ = 0)

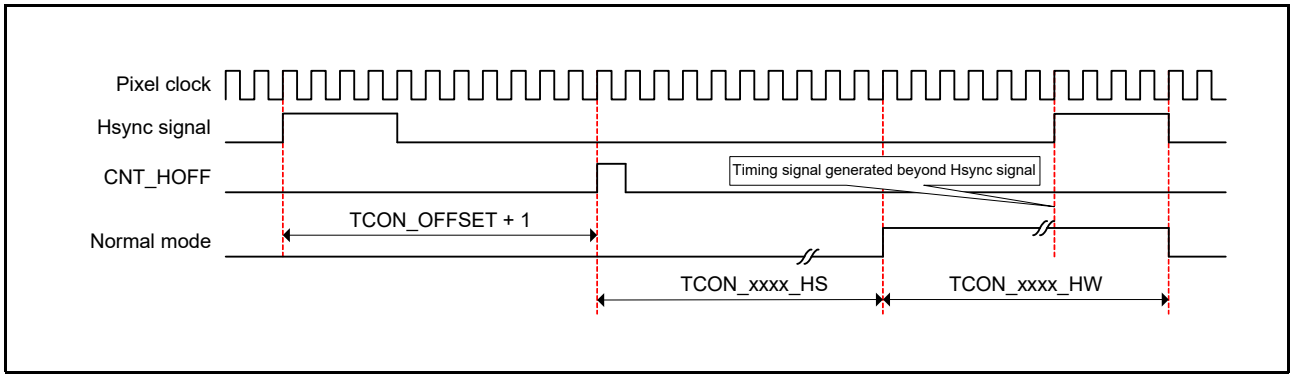


Figure 39.12 Horizontal Panel Driver Signal (in Normal Mode and When Offset Horizontal Reference is Used)

Table 39.15 Settings for Horizontal Panel Driver Signal Generation (A)

Register Name	Bit Name	Initial Value	Description
TCON_TIM_STH1	TCON_STH_HS[10:0]	0	STH Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STH_HS from the rising edge of the Hsync signal (clock cycles)
TCON_TIM_STH1	TCON_STH_HW[10:0]	96	STH Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STH_HW (clock cycles)
TCON_TIM_STB1	TCON_STB_HS[10:0]	144	STB Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STB_HS from the rising edge of the Hsync signal (clock cycles)
TCON_TIM_STB1	TCON_STB_HW[10:0]	640	STB Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STB_HW (clock cycles)
TCON_TIM_CPV1	TCON_CPV_HS[10:0]	0	CPV Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_CPV_HS from the rising edge of the Hsync signal (clock cycles)
TCON_TIM_CPV1	TCON_CPV_HW[10:0]	0	CPV Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_CPV_HW (clock cycles)

(3) Horizontal Panel Driver Signal Generation (B)

In addition to the normal mode operation described in (2), reverse mode operation, that is, horizontal panel driver signal generation (B) is provided. In reverse mode, operation starts at the rising edge of the Vsync signal as the reference and a signal is generated such that its polarity is inverted every horizontal period in the timing set by the TCON_xxxx_HS[10:0] bits, which set the first changing timing.

In reverse mode, regardless of whether the number of lines in the vertical direction is odd or even, the polarity of the signals generated is inverted every horizontal period. The following three reverse modes are selectable for polarity inversion operation.

Table 39.16 Horizontal Panel Driver Signal Generation Modes

Register Name	Bit Name	Initial Value	Description
TCON_TIM_POLA2	TCON_POLA_MD [1:0]	1	POLA Signal Generation Mode Select 0: Normal mode Generates the signal that changes twice a horizontal period. 1: 1 × 1 reverse mode Generates the signal whose polarity is inverted every horizontal period. 2: 1 × 2 reverse mode Generates the signal whose polarity is inverted in the first horizontal period and is subsequently inverted every two horizontal periods. 3: 2 × 2 reverse mode Generates the signal whose polarity is inverted every two horizontal periods.

Table 39.16 Horizontal Panel Driver Signal Generation Modes

Register Name	Bit Name	Initial Value	Description
TCON_TIM_POLB2	TCON_POLB_MD [1:0]	1	POLB Signal Generation Mode Select 0: Normal mode Generates the signal that changes twice a horizontal period. 1: 1 × 1 reverse mode Generates the signal whose polarity is inverted every horizontal period. 2: 1 × 2 reverse mode Generates the signal whose polarity is inverted in the first horizontal period and is subsequently inverted every two horizontal periods. 3: 2 × 2 reverse mode Generates the signal whose polarity is inverted every two horizontal periods.

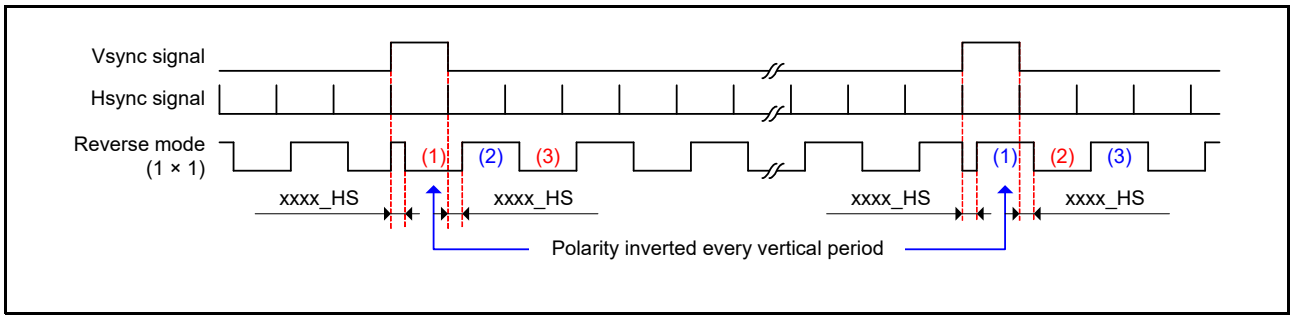


Figure 39.13 Horizontal Panel Driver Signal (in 1 × 1 Reverse Mode)

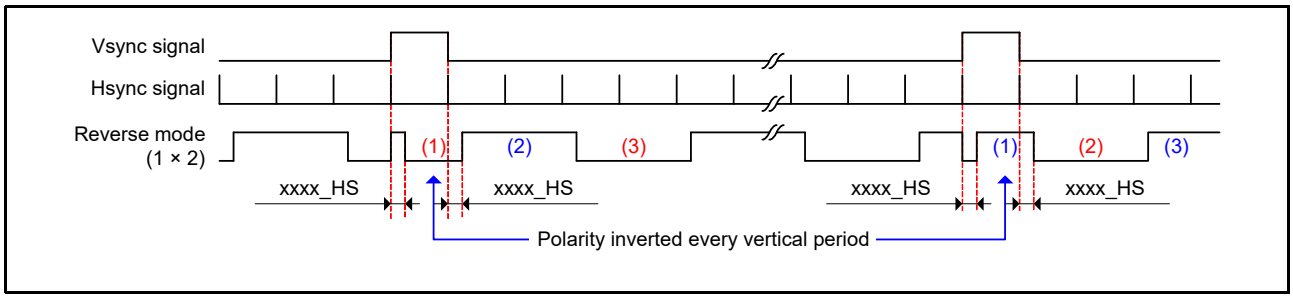


Figure 39.14 Horizontal Panel Driver Signal (in 1 × 2 Reverse Mode)

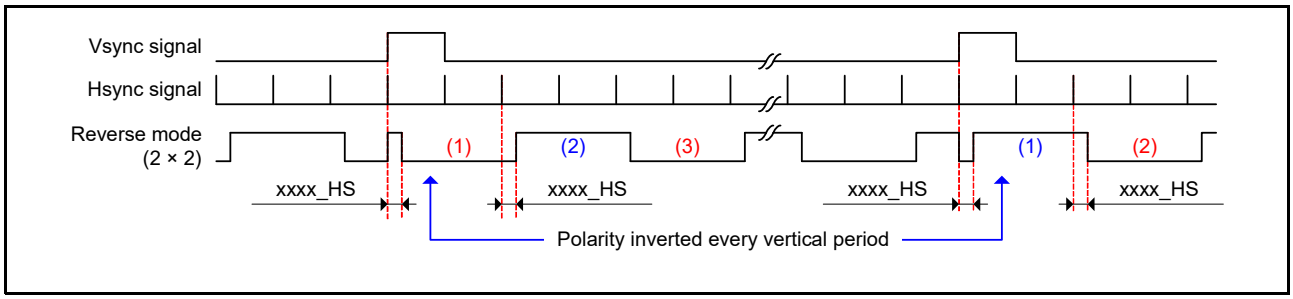


Figure 39.15 Horizontal Panel Driver Signal (in 2 × 2 Reverse Mode)

Table 39.17 Settings of Horizontal Panel Driver Signal Generation (B)

Register Name	Bit Name	Initial Value	Description
TCON_TIM_POLA1	TCON_POLA_HS [10:0]	0	POLA Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_POLA_HS + 1 from the rising edge of the Hsync signal (clock cycles) Note: When 1 × 1, 1 × 2, or 2 × 2 reverse mode is selected, these bits should be set to 1 or greater.
TCON_TIM_POLA1	TCON_POLA_HW [10:0]	0	POLA Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_POLA_HW (clock cycles)
TCON_TIM_POLB1	TCON_POLB_HS [10:0]	0	POLBA Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_POLB_HS + 1 from the rising edge of the Hsync signal (clock cycles) Note: When 1 × 1, 1 × 2, or 2 × 2 reverse mode is selected, these bits should be set to1 or greater.
TCON_TIM_POLB1	TCON_POLB_HW [10:0]	0	POLB Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_POLB_HW (clock cycles)

(4) Vertical Panel Driver Signal Generation

The vertical synchronous panel driver signal generation involves the following operations.

1. Initialization at the rising edge of the Vsync signal
2. Generation of a timing signal that changes twice in a vertical period according to the values of the internal counter, and TCON_xxxx_VS[10:0] and TCON_xxxx_VW[10:0] bits, which set the first changing timing and the second changing timing, respectively.

The internal counter increments the counter value in the following two cases.

1. At the rising edge of the Hsync signal
2. At the point reached after a clock delay specified by the value of TCON_HALF[10:0] from the rising edge of the Hsync signal (normally, 1/2fH is set).

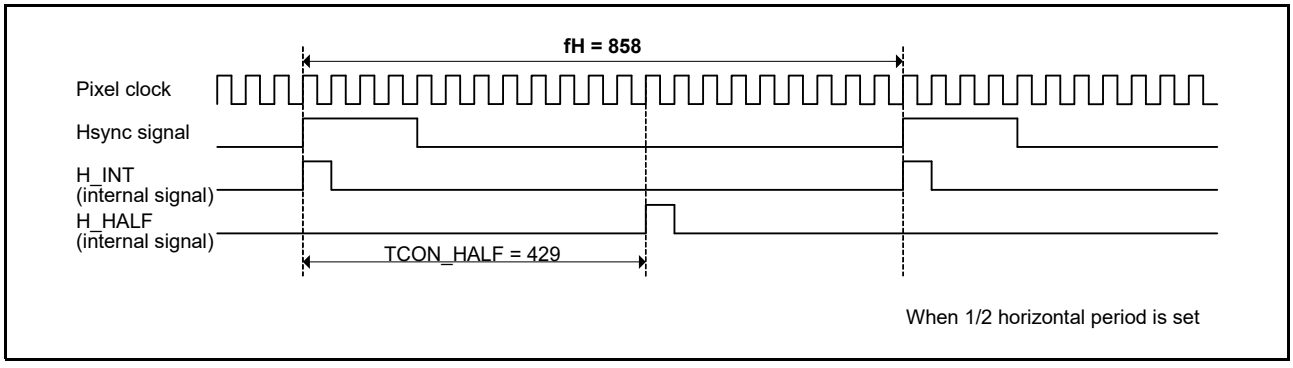


Figure 39.16 1/2 Pulse (H_HALF) Signal Generation

Table 39.18 Settings of 1/2 Pulse (H_HALF) Signal Generation

Register Name	Bit Name	Initial Value	Description
TCON_TIM	TCON_HALF[10:0]	400	1/2fH Timing Specifies the clock count from the rising edge of the Hsync signal as the counting timing of horizontal counter

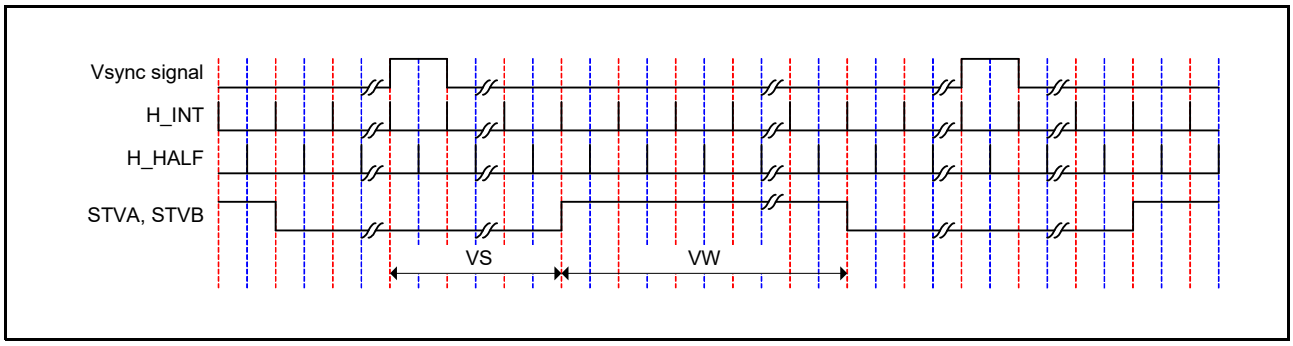


Figure 39.17 Vertical Panel Driver Signal (H_INT Reference Operation)

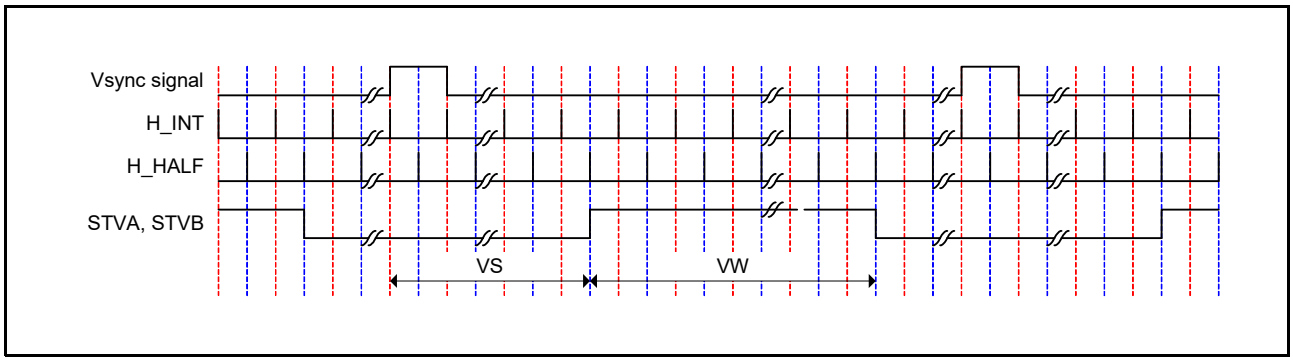


Figure 39.18 Vertical Panel Driver Signal (H_HALF Reference Operation)

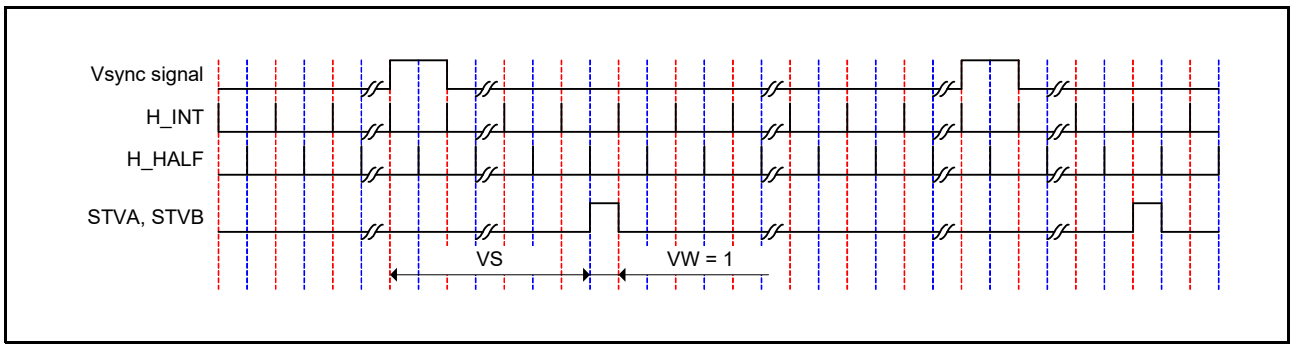


Figure 39.19 Vertical Panel Driver Signal (H_INT and H_HALF Reference Operation)

Table 39.19 Vertical Panel Driver Signal Generation

Register Name	Bit Name	Initial Value	Description
TCON_TIM_STVA1	TCON_STVA_VS[10:0]	0	STVA Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STVA_HS from the rising edge of the Vsync signal (1/2fH cycles)
TCON_TIM_STVA1	TCON_STVA_VW[10:0]	4	STVA Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STVA_HW (1/2fH cycles)
TCON_TIM_STVB1	TCON_STVB_VS[10:0]	70	STVB Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STVB_HS from the rising edge of the Vsync signal (1/2fH cycles)
TCON_TIM_STVB1	TCON_STVB_VW[10:0]	960	STVB Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STVB_HW (1/2fH cycles)

(5) DE Timing Signal Generation

DE timing signal generation involves generation of data enable signal (DE) that indicates the valid period of the video signal by synthesizing the horizontal panel driver (HE) signal and the vertical panel driver (VE) signal (AND).

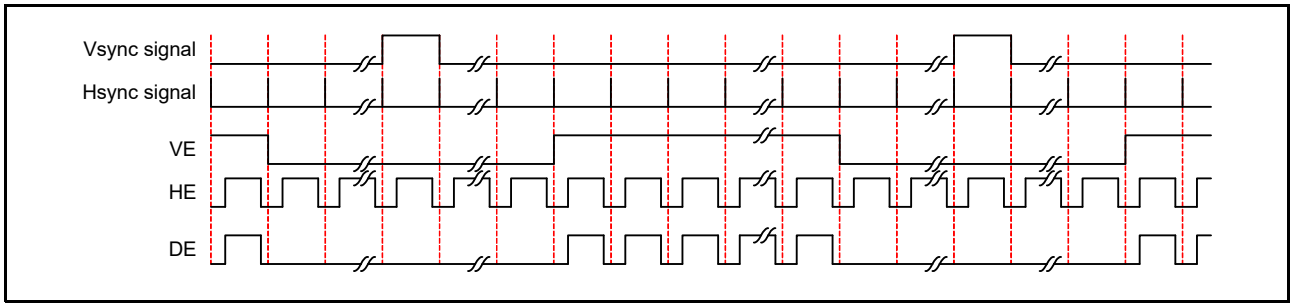


Figure 39.20 Data Enable Signal Generation

(6) Polarity Inversion

Polarity inversion enables inversion of polarity of each signal generated by the signal generating circuit.

Table 39.20 Panel Driver Signal Polarity Inversion Control

Register Name	Bit Name	Initial Value	Description
TCON_TIM_STVA2	TCON_STVA_INV	1	Polarity Inversion Control of STVA Signal 0: Not inverted 1: Inverted
TCON_TIM_STVB2	TCON_STVB_INV	0	Polarity Inversion Control of STVB Signal 0: Not inverted 1: Inverted
TCON_TIM_STH2	TCON_STH_INV	1	Polarity Inversion Control of STH Signal 0: Not inverted 1: Inverted
TCON_TIM_STB2	TCON_STB_INV	0	Polarity Inversion Control of STB Signal 0: Not inverted 1: Inverted
TCON_TIM_CPV2	TCON_CPV_INV	0	Polarity Inversion Control of CPV Signal 0: Not inverted 1: Inverted
TCON_TIM_POLA2	TCON_POLA_INV	0	Polarity Inversion Control of POLA Signal 0: Not inverted 1: Inverted
TCON_TIM_POLB2	TCON_POLB_INV	0	Polarity Inversion Control of POLB Signal 0: Not inverted 1: Inverted
TCON_TIM_DE	TCON_DE_INV	0	Polarity Inversion Control of DE Signal 0: Not inverted 1: Inverted

(7) Output Selection

An output pin is selected for every signal subjected to polarity inversion control.

Table 39.21 Panel Driver Signal Output Selection

Register Name	Bit Name	Initial Value	Description
TCON_TIM_STVA2	TCON_STVA_SEL [2:0]	0	Output Signal Select for LCD_TCON0 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_STVB2	TCON_STVB_SEL [2:0]	1	Output Signal Select for LCD_TCON1 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Table 39.21 Panel Driver Signal Output Selection

Register Name	Bit Name	Initial Value	Description
TCON_TIM_STH2	TCON_STH_SEL [2:0]	2	Output Signal Select for LCD_TCON2 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_STB2	TCON_STB_SEL [2:0]	7	Output Signal Select for LCD_TCON3 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_CPV2	TCON_CPV_SEL [2:0]	4	Output Signal Select for LCD_TCON4 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_POLA2	TCON_POLA_SEL [2:0]	5	Output Signal Select for LCD_TCON5 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_POLB2	TCON_POLB_SEL [2:0]	6	Output Signal Select for LCD_TCON6 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

(8) Output Phase Selection

The output phase can be individually selected for the video output signal and the various timing output signals based on the LCD_CLK (panel clock).

Table 39.22 Panel Output Signal Phase Selection

Register Name	Bit Name	Initial Value	Description
OUT_CLK_PHASE	OUTCNT_LCD_EDGE	0	Output Phase Control of LCD_DATA23 to LCD_DATA0 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_STVA_EDGE	0	Output Phase Control of LCD_TCON0 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_STVB_EDGE	0	Output Phase Control of LCD_TCON1 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_STH_EDGE	0	Output Phase Control of LCD_TCON2 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_STB_EDGE	0	Output Phase Control of LCD_TCON3 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_CPV_EDGE	0	Output Phase Control of LCD_TCON4 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_POLA_EDGE	0	Output Phase Control of LCD_TCON5 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_POLB_EDGE	0	Output Phase Control of LCD_TCON6 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin

39.2 Register Descriptions

Table 39.23 to Table 39.25 show the register configuration.

- Symbols used in Register Description:

Initial value: Register value after a reset
—: Undefined value
R/W: Readable/writable. The written value can be read.
R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.
R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.
R: Read-only. The write value should always be 0.
—/W: Write-only. The read value is undefined.

Table 39.23 Gamma Correction Block Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Register update control register G in gamma correction block	GAM_G_UPDATE	R/WC1	H'0000 0000	H'FCFF 7800	32
Function switch register in gamma correction block	GAM_SW	R/W	H'0000 0000	H'FCFF 7804	32
Table setting register G1 in gamma correction block	GAM_G_LUT1	R/W	H'0400 0400	H'FCFF 7808	32
Table setting register G2 in gamma correction block	GAM_G_LUT2	R/W	H'0400 0400	H'FCFF 780C	32
Table setting register G3 in gamma correction block	GAM_G_LUT3	R/W	H'0400 0400	H'FCFF 7810	32
Table setting register G4 in gamma correction block	GAM_G_LUT4	R/W	H'0400 0400	H'FCFF 7814	32
Table setting register G5 in gamma correction block	GAM_G_LUT5	R/W	H'0400 0400	H'FCFF 7818	32
Table setting register G6 in gamma correction block	GAM_G_LUT6	R/W	H'0400 0400	H'FCFF 781C	32
Table setting register G7 in gamma correction block	GAM_G_LUT7	R/W	H'0400 0400	H'FCFF 7820	32
Table setting register G8 in gamma correction block	GAM_G_LUT8	R/W	H'0400 0400	H'FCFF 7824	32
Table setting register G9 in gamma correction block	GAM_G_LUT9	R/W	H'0400 0400	H'FCFF 7828	32
Table setting register G10 in gamma correction block	GAM_G_LUT10	R/W	H'0400 0400	H'FCFF 782C	32
Table setting register G11 in gamma correction block	GAM_G_LUT11	R/W	H'0400 0400	H'FCFF 7830	32
Table setting register G12 in gamma correction block	GAM_G_LUT12	R/W	H'0400 0400	H'FCFF 7834	32
Table setting register G13 in gamma correction block	GAM_G_LUT13	R/W	H'0400 0400	H'FCFF 7838	32
Table setting register G14 in gamma correction block	GAM_G_LUT14	R/W	H'0400 0400	H'FCFF 783C	32
Table setting register G15 in gamma correction block	GAM_G_LUT15	R/W	H'0400 0400	H'FCFF 7840	32
Table setting register G16 in gamma correction block	GAM_G_LUT16	R/W	H'0400 0400	H'FCFF 7844	32
Area setting register G1 in gamma correction block	GAM_G_AREA1	R/W	H'0008 1018	H'FCFF 7848	32
Area setting register G2 in gamma correction block	GAM_G_AREA2	R/W	H'2028 3038	H'FCFF 784C	32

Table 39.23 Gamma Correction Block Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Area setting register G3 in gamma correction block	GAM_G_AREA3	R/W	H'4048 5058	H'FCFF 7850	32
Area setting register G4 in gamma correction block	GAM_G_AREA4	R/W	H'6068 7078	H'FCFF 7854	32
Area setting register G5 in gamma correction block	GAM_G_AREA5	R/W	H'8088 9098	H'FCFF 7858	32
Area setting register G6 in gamma correction block	GAM_G_AREA6	R/W	H'A0A8 B0B8	H'FCFF 785C	32
Area setting register G7 in gamma correction block	GAM_G_AREA7	R/W	H'C0C8 D0D8	H'FCFF 7860	32
Area setting register G8 in gamma correction block	GAM_G_AREA8	R/W	H'E0E8 F0F8	H'FCFF 7864	32
Register update control register B in gamma correction block	GAM_B_UPDATE	R/WC1	H'0000 0000	H'FCFF 7880	32
Table setting register B1 in gamma correction block	GAM_B_LUT1	R/W	H'0400 0400	H'FCFF 7888	32
Table setting register B2 in gamma correction block	GAM_B_LUT2	R/W	H'0400 0400	H'FCFF 788C	32
Table setting register B3 in gamma correction block	GAM_B_LUT3	R/W	H'0400 0400	H'FCFF 7890	32
Table setting register B4 in gamma correction block	GAM_B_LUT4	R/W	H'0400 0400	H'FCFF 7894	32
Table setting register B5 in gamma correction block	GAM_B_LUT5	R/W	H'0400 0400	H'FCFF 7898	32
Table setting register B6 in gamma correction block	GAM_B_LUT6	R/W	H'0400 0400	H'FCFF 789C	32
Table setting register B7 in gamma correction block	GAM_B_LUT7	R/W	H'0400 0400	H'FCFF 78A0	32
Table setting register B8 in gamma correction block	GAM_B_LUT8	R/W	H'0400 0400	H'FCFF 78A4	32
Table setting register B9 in gamma correction block	GAM_B_LUT9	R/W	H'0400 0400	H'FCFF 78A8	32
Table setting register B10 in gamma correction block	GAM_B_LUT10	R/W	H'0400 0400	H'FCFF 78AC	32
Table setting register B11 in gamma correction block	GAM_B_LUT11	R/W	H'0400 0400	H'FCFF 78B0	32
Table setting register B12 in gamma correction block	GAM_B_LUT12	R/W	H'0400 0400	H'FCFF 78B4	32
Table setting register B13 in gamma correction block	GAM_B_LUT13	R/W	H'0400 0400	H'FCFF 78B8	32
Table setting register B14 in gamma correction block	GAM_B_LUT14	R/W	H'0400 0400	H'FCFF 78BC	32
Table setting register B15 in gamma correction block	GAM_B_LUT15	R/W	H'0400 0400	H'FCFF 78C0	32
Table setting register B16 in gamma correction block	GAM_B_LUT16	R/W	H'0400 0400	H'FCFF 78C4	32
Area setting register B1 in gamma correction block	GAM_B_AREA1	R/W	H'0008 1018	H'FCFF 78C8	32
Area setting register B2 in gamma correction block	GAM_B_AREA2	R/W	H'2028 3038	H'FCFF 78CC	32
Area setting register B3 in gamma correction block	GAM_B_AREA3	R/W	H'4048 5058	H'FCFF 78D0	32

Table 39.23 Gamma Correction Block Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Area setting register B4 in gamma correction block	GAM_B_AREA4	R/W	H'6068 7078	H'FCFF 78D4	32
Area setting register B5 in gamma correction block	GAM_B_AREA5	R/W	H'8088 9098	H'FCFF 78D8	32
Area setting register B6 in gamma correction block	GAM_B_AREA6	R/W	H'A0A8 B0B8	H'FCFF 78DC	32
Area setting register B7 in gamma correction block	GAM_B_AREA7	R/W	H'C0C8 D0D8	H'FCFF 78E0	32
Area setting register B8 in gamma correction block	GAM_B_AREA8	R/W	H'E0E8 F0F8	H'FCFF 78E4	32
Register update control register R in gamma correction block	GAM_R_UPDATE	R/WC1	H'0000 0000	H'FCFF 7900	32
Table setting register R1 in gamma correction block	GAM_R_LUT1	R/W	H'0400 0400	H'FCFF 7908	32
Table setting register R2 in gamma correction block	GAM_R_LUT2	R/W	H'0400 0400	H'FCFF 790C	32
Table setting register R3 in gamma correction block	GAM_R_LUT3	R/W	H'0400 0400	H'FCFF 7910	32
Table setting register R4 in gamma correction block	GAM_R_LUT4	R/W	H'0400 0400	H'FCFF 7914	32
Table setting register R5 in gamma correction block	GAM_R_LUT5	R/W	H'0400 0400	H'FCFF 7918	32
Table setting register R6 in gamma correction block	GAM_R_LUT6	R/W	H'0400 0400	H'FCFF 791C	32
Table setting register R7 in gamma correction block	GAM_R_LUT7	R/W	H'0400 0400	H'FCFF 7920	32
Table setting register R8 in gamma correction block	GAM_R_LUT8	R/W	H'0400 0400	H'FCFF 7924	32
Table setting register R9 in gamma correction block	GAM_R_LUT9	R/W	H'0400 0400	H'FCFF 7928	32
Table setting register R10 in gamma correction block	GAM_R_LUT10	R/W	H'0400 0400	H'FCFF 792C	32
Table setting register R11 in gamma correction block	GAM_R_LUT11	R/W	H'0400 0400	H'FCFF 7930	32
Table setting register R12 in gamma correction block	GAM_R_LUT12	R/W	H'0400 0400	H'FCFF 7934	32
Table setting register R13 in gamma correction block	GAM_R_LUT13	R/W	H'0400 0400	H'FCFF 7938	32
Table setting register R14 in gamma correction block	GAM_R_LUT14	R/W	H'0400 0400	H'FCFF 793C	32
Table setting register R15 in gamma correction block	GAM_R_LUT15	R/W	H'0400 0400	H'FCFF 7940	32
Table setting register R16 in gamma correction block	GAM_R_LUT16	R/W	H'0400 0400	H'FCFF 7944	32
Area setting register R1 in gamma correction block	GAM_R_AREA1	R/W	H'0008 1018	H'FCFF 7948	32
Area setting register R2 in gamma correction block	GAM_R_AREA2	R/W	H'2028 3038	H'FCFF 794C	32
Area setting register R3 in gamma correction block	GAM_R_AREA3	R/W	H'4048 5058	H'FCFF 7950	32
Area setting register R4 in gamma correction block	GAM_R_AREA4	R/W	H'6068 7078	H'FCFF 7954	32

Table 39.23 Gamma Correction Block Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Area setting register R5 in gamma correction block	GAM_R_AREA5	R/W	H'8088 9098	H'FCFF 7958	32
Area setting register R6 in gamma correction block	GAM_R_AREA6	R/W	H'A0A8 B0B8	H'FCFF 795C	32
Area setting register R7 in gamma correction block	GAM_R_AREA7	R/W	H'C0C8 D0D8	H'FCFF 7960	32
Area setting register R8 in gamma correction block	GAM_R_AREA8	R/W	H'E0E8 F0F8	H'FCFF 7964	32

Table 39.24 TCON Block Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
TCON register update control register	TCON_UPDATE	R/WC1	H'0000 0000	H'FCFF 7980	32
TCON reference timing setting register	TCON_TIM	R/W	H'0190 0000	H'FCFF 7984	32
TCON vertical timing setting register A1	TCON_TIM_STVA1	R/W	H'0000 0004	H'FCFF 7988	32
TCON vertical timing setting register A2	TCON_TIM_STVA2	R/W	H'0000 0010	H'FCFF 798C	32
TCON vertical timing setting register B1	TCON_TIM_STVB1	R/W	H'0046 03C0	H'FCFF 7990	32
TCON vertical timing setting register B2	TCON_TIM_STVB2	R/W	H'0000 0001	H'FCFF 7994	32
TCON horizontal timing setting register STH1	TCON_TIM_STH1	R/W	H'0000 0060	H'FCFF 7998	32
TCON horizontal timing setting register STH2	TCON_TIM_STH2	R/W	H'0000 0012	H'FCFF 799C	32
TCON horizontal timing setting register STB1	TCON_TIM_STB1	R/W	H'0090 0280	H'FCFF 79A0	32
TCON horizontal timing setting register STB2	TCON_TIM_STB2	R/W	H'0000 0007	H'FCFF 79A4	32
TCON horizontal timing setting register CPV1	TCON_TIM_CPV1	R/W	H'0000 0000	H'FCFF 79A8	32
TCON horizontal timing setting register CPV2	TCON_TIM_CPV2	R/W	H'0000 0004	H'FCFF 79AC	32
TCON horizontal timing setting register POLA1	TCON_TIM_POLA1	R/W	H'0000 0000	H'FCFF 79B0	32
TCON horizontal timing setting register POLA2	TCON_TIM_POLA2	R/W	H'0000 1005	H'FCFF 79B4	32
TCON horizontal timing setting register POLB1	TCON_TIM_POLB1	R/W	H'0000 0000	H'FCFF 79B8	32
TCON horizontal timing setting register POLB2	TCON_TIM_POLB2	R/W	H'0000 1006	H'FCFF 79BC	32
TCON data enable polarity setting register	TCON_TIM_DE	R/W	H'0000 0000	H'FCFF 79C0	32

Table 39.25 Output Controller Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Register update control register in output controller	OUT_UPDATE	R/WC1	H'0000 0000	H'FCFF 7A00	32
Output interface register	OUT_SET	R/W	H'001F 0000	H'FCFF 7A04	32
Brightness (DC) correction register 1	OUT_BRIGHT1	R/W	H'0000 0200	H'FCFF 7A08	32
Brightness (DC) correction register 2	OUT_BRIGHT2	R/W	H'0200 0200	H'FCFF 7A0C	32
Contrast (gain) correction register	OUT_CONTRAST	R/W	H'0080 8080	H'FCFF 7A10	32
Panel dither register	OUT_PDTHA	R/W	H'0000 3021	H'FCFF 7A14	32
Output phase control register	OUT_CLK_PHASE	R/W	H'0000 0000	H'FCFF 7A24	32

39.2.1 Register Update Control Register G in Gamma Correction Block (GAM_G_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GAM_G_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GAM_G_VEN	0	R/WC1	Gamma Correction (G) Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

39.2.2 Function Switch Register in Gamma Correction Block (GAM_SW)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GAM_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GAM_ON	0	R/W	Gamma Correction On/Off Control 0: Off 1: On

Note: This register is updated when GAM_G_VEN in GAM_G_UPDATE is 1.

39.2.3 Table Setting Register G1 to G16 in Gamma Correction Block (GAM_G_LUT1 to GAM_G_LUT16)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GAM_G_GAIN_xx[10:0]										
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GAM_G_GAIN_yy[10:0]										
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	*	1024	R/W	GAM_G_LUT1: Gain Adjustment of Area 0 of G Signal GAM_G_LUT2: Gain Adjustment of Area 2 of G Signal GAM_G_LUT3: Gain Adjustment of Area 4 of G Signal GAM_G_LUT4: Gain Adjustment of Area 6 of G Signal GAM_G_LUT5: Gain Adjustment of Area 8 of G Signal GAM_G_LUT6: Gain Adjustment of Area 10 of G Signal GAM_G_LUT7: Gain Adjustment of Area 12 of G Signal GAM_G_LUT8: Gain Adjustment of Area 14 of G Signal GAM_G_LUT9: Gain Adjustment of Area 16 of G Signal GAM_G_LUT10: Gain Adjustment of Area 18 of G Signal GAM_G_LUT11: Gain Adjustment of Area 20 of G Signal GAM_G_LUT12: Gain Adjustment of Area 22 of G Signal GAM_G_LUT13: Gain Adjustment of Area 24 of G Signal GAM_G_LUT14: Gain Adjustment of Area 26 of G Signal GAM_G_LUT15: Gain Adjustment of Area 28 of G Signal GAM_G_LUT16: Gain Adjustment of Area 30 of G Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times])
26 to 16	*	1024	R/W	*: Bit Name GAM_G_LUT1: GAM_G_GAIN_00[10:0] GAM_G_LUT2: GAM_G_GAIN_02[10:0] GAM_G_LUT3: GAM_G_GAIN_04[10:0] GAM_G_LUT4: GAM_G_GAIN_06[10:0] GAM_G_LUT5: GAM_G_GAIN_08[10:0] GAM_G_LUT6: GAM_G_GAIN_10[10:0] GAM_G_LUT7: GAM_G_GAIN_12[10:0] GAM_G_LUT8: GAM_G_GAIN_14[10:0] GAM_G_LUT9: GAM_G_GAIN_16[10:0] GAM_G_LUT10: GAM_G_GAIN_18[10:0] GAM_G_LUT11: GAM_G_GAIN_20[10:0] GAM_G_LUT12: GAM_G_GAIN_22[10:0] GAM_G_LUT13: GAM_G_GAIN_24[10:0] GAM_G_LUT14: GAM_G_GAIN_26[10:0] GAM_G_LUT15: GAM_G_GAIN_28[10:0] GAM_G_LUT16: GAM_G_GAIN_30[10:0]
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	*	1024	R/W	GAM_G_LUT1: Gain Adjustment of Area 1 of G Signal GAM_G_LUT2: Gain Adjustment of Area 3 of G Signal GAM_G_LUT3: Gain Adjustment of Area 5 of G Signal GAM_G_LUT4: Gain Adjustment of Area 7 of G Signal GAM_G_LUT5: Gain Adjustment of Area 9 of G Signal GAM_G_LUT6: Gain Adjustment of Area 11 of G Signal GAM_G_LUT7: Gain Adjustment of Area 13 of G Signal GAM_G_LUT8: Gain Adjustment of Area 15 of G Signal GAM_G_LUT9: Gain Adjustment of Area 17 of G Signal GAM_G_LUT10: Gain Adjustment of Area 19 of G Signal

Bit	Bit Name	Initial Value	R/W	Description
10 to 0	*	1024	R/W	GAM_G_LUT11: Gain Adjustment of Area 21 of G Signal GAM_G_LUT12: Gain Adjustment of Area 23 of G Signal GAM_G_LUT13: Gain Adjustment of Area 25 of G Signal GAM_G_LUT14: Gain Adjustment of Area 27 of G Signal GAM_G_LUT15: Gain Adjustment of Area 29 of G Signal GAM_G_LUT16: Gain Adjustment of Area 31 of G Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times]) *: Bit Name GAM_G_LUT1: GAM_G_GAIN_01[10:0] GAM_G_LUT2: GAM_G_GAIN_03[10:0] GAM_G_LUT3: GAM_G_GAIN_05[10:0] GAM_G_LUT4: GAM_G_GAIN_07[10:0] GAM_G_LUT5: GAM_G_GAIN_09[10:0] GAM_G_LUT6: GAM_G_GAIN_11[10:0] GAM_G_LUT7: GAM_G_GAIN_13[10:0] GAM_G_LUT8: GAM_G_GAIN_15[10:0] GAM_G_LUT9: GAM_G_GAIN_17[10:0] GAM_G_LUT10: GAM_G_GAIN_19[10:0] GAM_G_LUT11: GAM_G_GAIN_21[10:0] GAM_G_LUT12: GAM_G_GAIN_23[10:0] GAM_G_LUT13: GAM_G_GAIN_25[10:0] GAM_G_LUT14: GAM_G_GAIN_27[10:0] GAM_G_LUT15: GAM_G_GAIN_29[10:0] GAM_G_LUT16: GAM_G_GAIN_31[10:0]

Note: This register is updated when GAM_G_VEN in GAM_G_UPDATE is 1.

39.2.4 Area Setting Register G1 in Gamma Correction Block (GAM_G_AREA1)

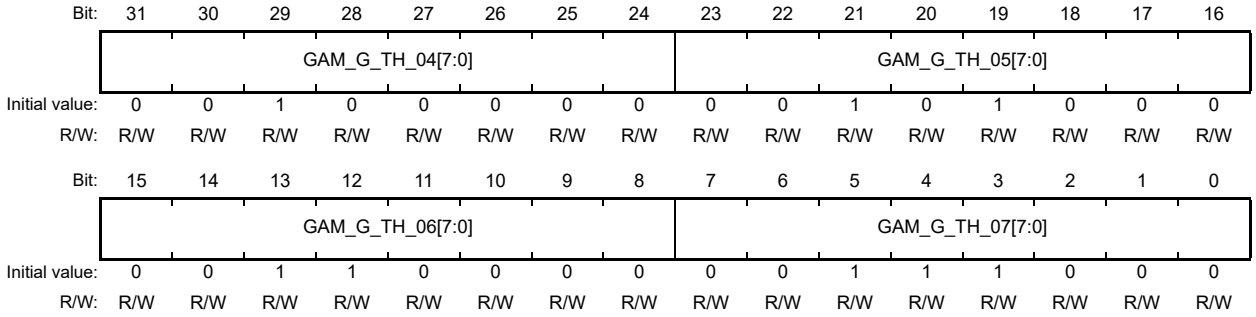
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GAM_G_TH_01[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_02[7:0]								GAM_G_TH_03[7:0]							
Initial value:	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GAM_G_TH_01[7:0]	8	R/W	Start Threshold of Area 1 of G Signal Unsigned (0 to 255 [LSB]) 0 < Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_02[7:0]	16	R/W	Start Threshold of Area 2 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_03[7:0]	24	R/W	Start Threshold of Area 3 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_G_VEN in GAM_G_UPDATE is 1.

39.2.5 Area Setting Register G2 in Gamma Correction Block (GAM_G_AREA2)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_04[7:0]	32	R/W	Start Threshold of Area 4 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_05[7:0]	40	R/W	Start Threshold of Area 5 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_06[7:0]	48	R/W	Start Threshold of Area 6 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_07[7:0]	56	R/W	Start Threshold of Area 7 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_G_VEN in GAM_G_UPDATE is 1.

39.2.6 Area Setting Register G3 in Gamma Correction Block (GAM_G_AREA3)

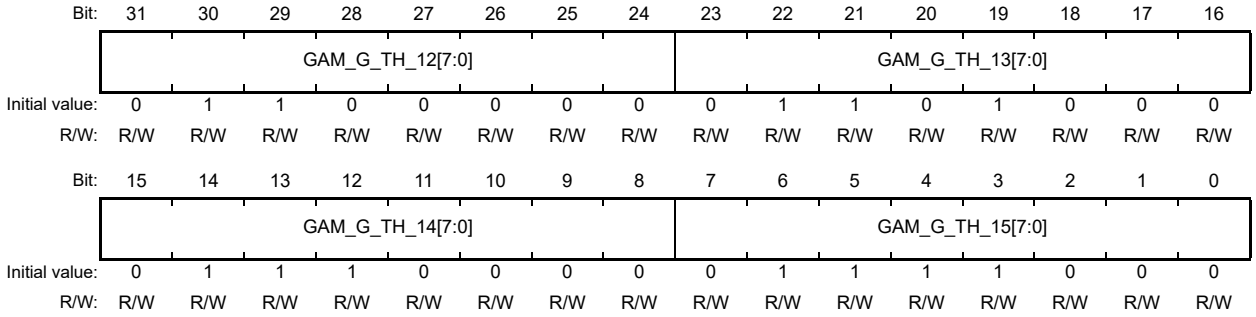
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_08[7:0]								GAM_G_TH_09[7:0]							
Initial value:	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_10[7:0]								GAM_G_TH_11[7:0]							
Initial value:	0	1	0	1	0	0	0	0	0	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_08[7:0]	64	R/W	Start Threshold of Area 8 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_09[7:0]	72	R/W	Start Threshold of Area 9 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_10[7:0]	80	R/W	Start Threshold of Area 10 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_11[7:0]	88	R/W	Start Threshold of Area 11 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_G_VEN in GAM_G_UPDATE is 1.

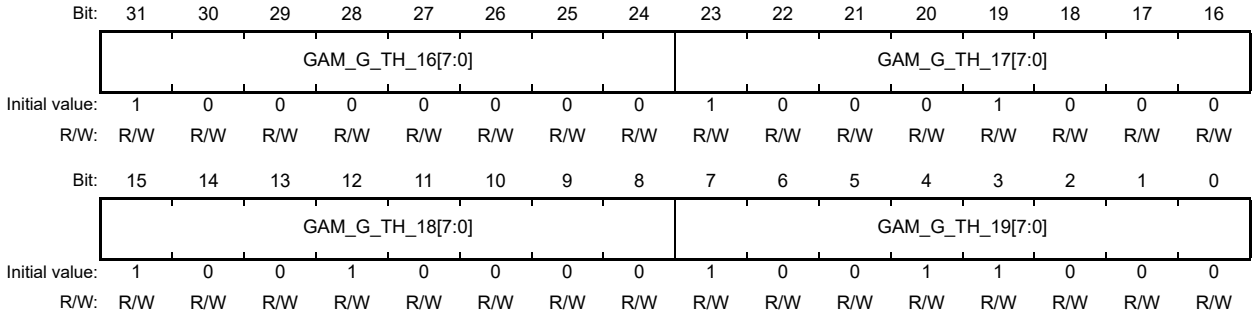
39.2.7 Area Setting Register G4 in Gamma Correction Block (GAM_G_AREA4)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_12[7:0]	96	R/W	Start Threshold of Area 12 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_13[7:0]	104	R/W	Start Threshold of Area 13 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_14[7:0]	112	R/W	Start Threshold of Area 14 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_15[7:0]	120	R/W	Start Threshold of Area 15 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_G_VEN in GAM_G_UPDATE is 1.

39.2.8 Area Setting Register G5 in Gamma Correction Block (GAM_G_AREA5)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_16 [7:0]	128	R/W	Start Threshold of Area 16 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_17 [7:0]	136	R/W	Start Threshold of Area 17 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_18 [7:0]	144	R/W	Start Threshold of Area 18 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_19 [7:0]	152	R/W	Start Threshold of Area 19 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_G_VEN in GAM_G_UPDATE is 1.

39.2.9 Area Setting Register G6 in Gamma Correction Block (GAM_G_AREA6)

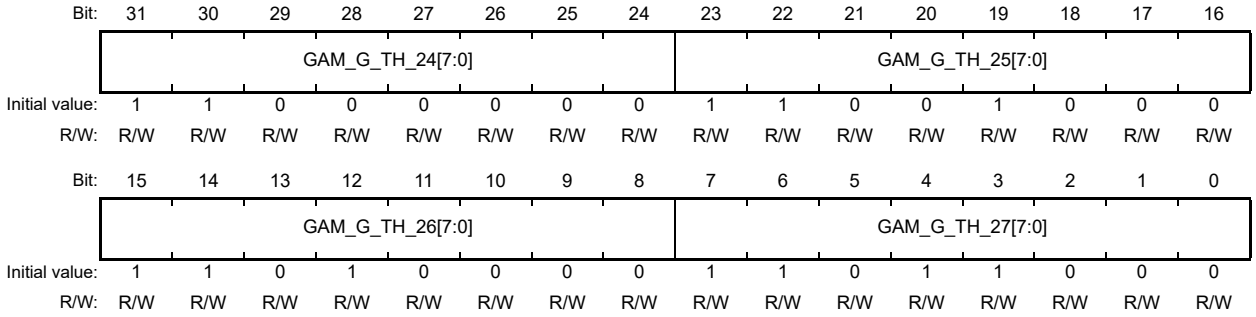
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_20[7:0]								GAM_G_TH_21[7:0]							
Initial value:	1	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_22[7:0]								GAM_G_TH_23[7:0]							
Initial value:	1	0	1	1	0	0	0	0	1	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_20 [7:0]	160	R/W	Start Threshold of Area 20 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_21 [7:0]	168	R/W	Start Threshold of Area 21 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_22 [7:0]	176	R/W	Start Threshold of Area 22 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_23 [7:0]	184	R/W	Start Threshold of Area 23 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_G_VEN in GAM_G_UPDATE is 1.

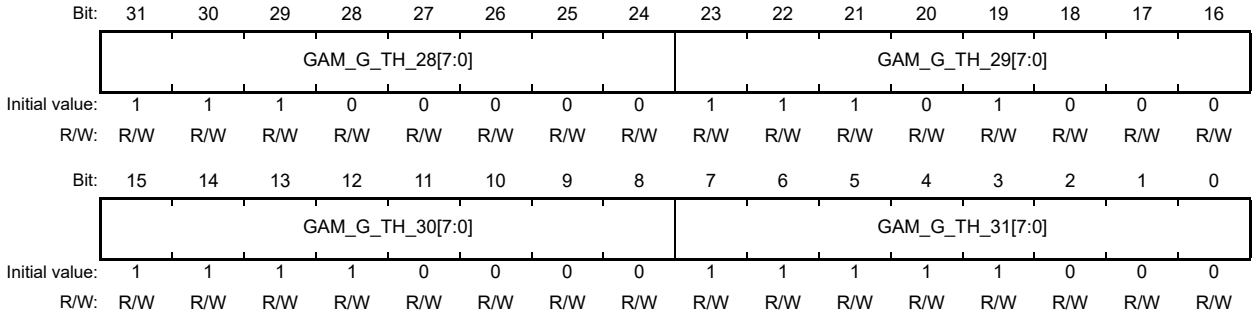
39.2.10 Area Setting Register G7 in Gamma Correction Block (GAM_G_AREA7)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_24 [7:0]	192	R/W	Start Threshold of Area 24 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_25 [7:0]	200	R/W	Start Threshold of Area 25 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_26 [7:0]	208	R/W	Start Threshold of Area 26 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_27 [7:0]	216	R/W	Start Threshold of Area 27 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_G_VEN in GAM_G_UPDATE is 1.

39.2.11 Area Setting Register G8 in Gamma Correction Block (GAM_G_AREA8)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_28 [7:0]	224	R/W	Start Threshold of Area 28 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_29 [7:0]	232	R/W	Start Threshold of Area 29 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_30 [7:0]	240	R/W	Start Threshold of Area 30 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_31 [7:0]	248	R/W	Start Threshold of Area 31 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area ≤ 255

Note: This register is updated when GAM_G_VEN in GAM_G_UPDATE is 1.

39.2.12 Register Update Control Register B in Gamma Correction Block (GAM_B_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GAM_B_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GAM_B_VEN	0	R/WC1	Gamma Correction (B) Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

39.2.13 Table Setting Register B1 to B16 in Gamma Correction Block (GAM_B_LUT1 to GAM_B_LUT16)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GAM_B_GAIN_xx[10:0]										
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GAM_B_GAIN_yy[10:0]										
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	*	1024	R/W	GAM_B_LUT1: Gain Adjustment of Area 0 of B Signal GAM_B_LUT2: Gain Adjustment of Area 2 of B Signal GAM_B_LUT3: Gain Adjustment of Area 4 of B Signal GAM_B_LUT4: Gain Adjustment of Area 6 of B Signal GAM_B_LUT5: Gain Adjustment of Area 8 of B Signal GAM_B_LUT6: Gain Adjustment of Area 10 of B Signal GAM_B_LUT7: Gain Adjustment of Area 12 of B Signal GAM_B_LUT8: Gain Adjustment of Area 14 of B Signal GAM_B_LUT9: Gain Adjustment of Area 16 of B Signal GAM_B_LUT10: Gain Adjustment of Area 18 of B Signal GAM_B_LUT11: Gain Adjustment of Area 20 of B Signal GAM_B_LUT12: Gain Adjustment of Area 22 of B Signal GAM_B_LUT13: Gain Adjustment of Area 24 of B Signal GAM_B_LUT14: Gain Adjustment of Area 26 of B Signal GAM_B_LUT15: Gain Adjustment of Area 28 of B Signal GAM_B_LUT16: Gain Adjustment of Area 30 of B Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times])
26 to 16	*	1024	R/W	*: Bit Name GAM_B_LUT1: GAM_B_GAIN_00[10:0] GAM_B_LUT2: GAM_B_GAIN_02[10:0] GAM_B_LUT3: GAM_B_GAIN_04[10:0] GAM_B_LUT4: GAM_B_GAIN_06[10:0] GAM_B_LUT5: GAM_B_GAIN_08[10:0] GAM_B_LUT6: GAM_B_GAIN_10[10:0] GAM_B_LUT7: GAM_B_GAIN_12[10:0] GAM_B_LUT8: GAM_B_GAIN_14[10:0] GAM_B_LUT9: GAM_B_GAIN_16[10:0] GAM_B_LUT10: GAM_B_GAIN_18[10:0] GAM_B_LUT11: GAM_B_GAIN_20[10:0] GAM_B_LUT12: GAM_B_GAIN_22[10:0] GAM_B_LUT13: GAM_B_GAIN_24[10:0] GAM_B_LUT14: GAM_B_GAIN_26[10:0] GAM_B_LUT15: GAM_B_GAIN_28[10:0] GAM_B_LUT16: GAM_B_GAIN_30[10:0]
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 0	*	1024	R/W	GAM_B_LUT1: Gain Adjustment of Area 1 of B Signal GAM_B_LUT2: Gain Adjustment of Area 3 of B Signal GAM_B_LUT3: Gain Adjustment of Area 5 of B Signal GAM_B_LUT4: Gain Adjustment of Area 7 of B Signal GAM_B_LUT5: Gain Adjustment of Area 9 of B Signal GAM_B_LUT6: Gain Adjustment of Area 11 of B Signal GAM_B_LUT7: Gain Adjustment of Area 13 of B Signal GAM_B_LUT8: Gain Adjustment of Area 15 of B Signal GAM_B_LUT9: Gain Adjustment of Area 17 of B Signal GAM_B_LUT10: Gain Adjustment of Area 19 of B Signal GAM_B_LUT11: Gain Adjustment of Area 21 of B Signal GAM_B_LUT12: Gain Adjustment of Area 23 of B Signal GAM_B_LUT13: Gain Adjustment of Area 25 of B Signal GAM_B_LUT14: Gain Adjustment of Area 27 of B Signal GAM_B_LUT15: Gain Adjustment of Area 29 of B Signal GAM_B_LUT16: Gain Adjustment of Area 31 of B Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times]) *: Bit Name GAM_B_LUT1: GAM_B_GAIN_01[10:0] GAM_B_LUT2: GAM_B_GAIN_03[10:0] GAM_B_LUT3: GAM_B_GAIN_05[10:0] GAM_B_LUT4: GAM_B_GAIN_07[10:0] GAM_B_LUT5: GAM_B_GAIN_09[10:0] GAM_B_LUT6: GAM_B_GAIN_11[10:0] GAM_B_LUT7: GAM_B_GAIN_13[10:0] GAM_B_LUT8: GAM_B_GAIN_15[10:0] GAM_B_LUT9: GAM_B_GAIN_17[10:0] GAM_B_LUT10: GAM_B_GAIN_19[10:0] GAM_B_LUT11: GAM_B_GAIN_21[10:0] GAM_B_LUT12: GAM_B_GAIN_23[10:0] GAM_B_LUT13: GAM_B_GAIN_25[10:0] GAM_B_LUT14: GAM_B_GAIN_27[10:0] GAM_B_LUT15: GAM_B_GAIN_29[10:0] GAM_B_LUT16: GAM_B_GAIN_31[10:0]

Note: This register is updated when GAM_B_VEN in GAM_B_UPDATE is 1.

39.2.14 Area Setting Register B1 in Gamma Correction Block (GAM_B_AREA1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GAM_B_TH_01[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_02[7:0]								GAM_B_TH_03[7:0]							
Initial value:	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GAM_B_TH_01[7:0]	8	R/W	Start Threshold of Area 1 of B Signal Unsigned (0 to 255 [LSB]) 0 < Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_02[7:0]	16	R/W	Start Threshold of Area 2 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_03[7:0]	24	R/W	Start Threshold of Area 3 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_B_VEN in GAM_B_UPDATE is 1.

39.2.15 Area Setting Register B2 in Gamma Correction Block (GAM_B_AREA2)

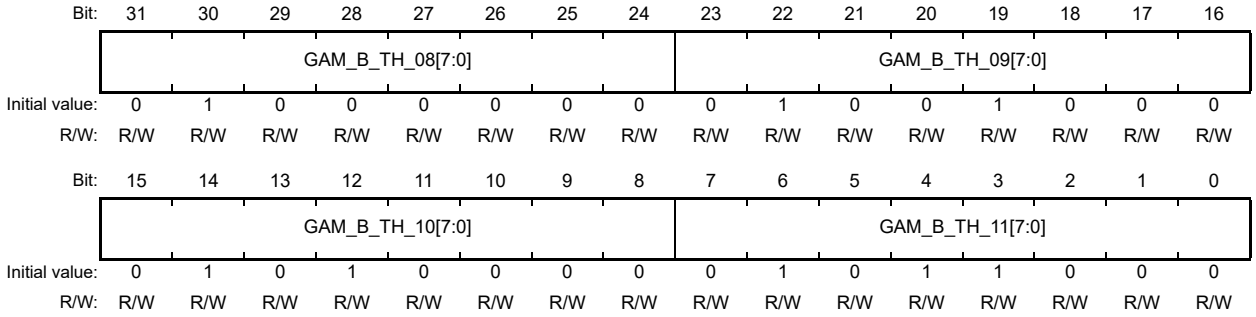
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_04[7:0]								GAM_B_TH_05[7:0]							
Initial value:	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_06[7:0]								GAM_B_TH_07[7:0]							
Initial value:	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_04[7:0]	32	R/W	Start Threshold of Area 4 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_05[7:0]	40	R/W	Start Threshold of Area 5 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_06[7:0]	48	R/W	Start Threshold of Area 6 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_07[7:0]	56	R/W	Start Threshold of Area 7 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_B_VEN in GAM_B_UPDATE is 1.

39.2.16 Area Setting Register B3 in Gamma Correction Block (GAM_B_AREA3)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_08[7:0]	64	R/W	Start Threshold of Area 8 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_09[7:0]	72	R/W	Start Threshold of Area 9 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_10[7:0]	80	R/W	Start Threshold of Area 10 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_11[7:0]	88	R/W	Start Threshold of Area 11 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_B_VEN in GAM_B_UPDATE is 1.

39.2.17 Area Setting Register B4 in Gamma Correction Block (GAM_B_AREA4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_12[7:0]								GAM_B_TH_13[7:0]							
Initial value:	0	1	1	0	0	0	0	0	0	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_14[7:0]								GAM_B_TH_15[7:0]							
Initial value:	0	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_12[7:0]	96	R/W	Start Threshold of Area 12 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_13[7:0]	104	R/W	Start Threshold of Area 13 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_14[7:0]	112	R/W	Start Threshold of Area 14 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_15[7:0]	120	R/W	Start Threshold of Area 15 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_B_VEN in GAM_B_UPDATE is 1.

39.2.18 Area Setting Register B5 in Gamma Correction Block (GAM_B_AREA5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_16[7:0]								GAM_B_TH_17[7:0]							
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_18[7:0]								GAM_B_TH_19[7:0]							
Initial value:	1	0	0	1	0	0	0	0	1	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_16[7:0]	128	R/W	Start Threshold of Area 16 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_17[7:0]	136	R/W	Start Threshold of Area 17 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_18[7:0]	144	R/W	Start Threshold of Area 18 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_19[7:0]	152	R/W	Start Threshold of Area 19 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_B_VEN in GAM_B_UPDATE is 1.

39.2.19 Area Setting Register B6 in Gamma Correction Block (GAM_B_AREA6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_20[7:0]								GAM_B_TH_21[7:0]							
Initial value:	1	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_22[7:0]								GAM_B_TH_23[7:0]							
Initial value:	1	0	1	1	0	0	0	0	1	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_20[7:0]	160	R/W	Start Threshold of Area 20 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_21[7:0]	168	R/W	Start Threshold of Area 21 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_22[7:0]	176	R/W	Start Threshold of Area 22 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_23[7:0]	184	R/W	Start Threshold of Area 23 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_B_VEN in GAM_B_UPDATE is 1.

39.2.20 Area Setting Register B7 in Gamma Correction Block (GAM_B_AREA7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_24[7:0]								GAM_B_TH_25[7:0]							
Initial value:	1	1	0	0	0	0	0	0	1	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_26[7:0]								GAM_B_TH_27[7:0]							
Initial value:	1	1	0	1	0	0	0	0	1	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_24[7:0]	192	R/W	Start Threshold of Area 24 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_25[7:0]	200	R/W	Start Threshold of Area 25 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_26[7:0]	208	R/W	Start Threshold of Area 26 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_27[7:0]	216	R/W	Start Threshold of Area 27 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_B_VEN in GAM_B_UPDATE is 1.

39.2.21 Area Setting Register B8 in Gamma Correction Block (GAM_B_AREA8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_28[7:0]								GAM_B_TH_29[7:0]							
Initial value:	1	1	1	0	0	0	0	0	1	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_30[7:0]								GAM_B_TH_31[7:0]							
Initial value:	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_28[7:0]	224	R/W	Start Threshold of Area 28 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_29[7:0]	232	R/W	Start Threshold of Area 29 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_30[7:0]	240	R/W	Start Threshold of Area 30 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_31[7:0]	248	R/W	Start Threshold of Area 31 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area ≤ 255

Note: This register is updated when GAM_B_VEN in GAM_B_UPDATE is 1.

39.2.22 Register Update Control Register R in Gamma Correction Block (GAM_R_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GAM_R_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GAM_R_VEN	0	R/WC1	Gamma Correction (R) Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

39.2.23 Table Setting Register R1 to R16 in Gamma Correction Block (GAM_R_LUT1 to GAM_R_LUT16)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GAM_R_GAIN_xx[10:0]										
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GAM_R_GAIN_yy[10:0]										
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	*	1024	R/W	GAM_R_LUT1: Gain Adjustment of Area 0 of R Signal GAM_R_LUT2: Gain Adjustment of Area 2 of R Signal GAM_R_LUT3: Gain Adjustment of Area 4 of R Signal GAM_R_LUT4: Gain Adjustment of Area 6 of R Signal GAM_R_LUT5: Gain Adjustment of Area 8 of R Signal GAM_R_LUT6: Gain Adjustment of Area 10 of R Signal GAM_R_LUT7: Gain Adjustment of Area 12 of R Signal GAM_R_LUT8: Gain Adjustment of Area 14 of R Signal GAM_R_LUT9: Gain Adjustment of Area 16 of R Signal GAM_R_LUT10: Gain Adjustment of Area 18 of R Signal GAM_R_LUT11: Gain Adjustment of Area 20 of R Signal GAM_R_LUT12: Gain Adjustment of Area 22 of R Signal GAM_R_LUT13: Gain Adjustment of Area 24 of R Signal GAM_R_LUT14: Gain Adjustment of Area 26 of R Signal GAM_R_LUT15: Gain Adjustment of Area 28 of R Signal GAM_R_LUT16: Gain Adjustment of Area 30 of R Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times]) *: Bit Name GAM_R_LUT1: GAM_R_GAIN_00[10:0] GAM_R_LUT2: GAM_R_GAIN_02[10:0] GAM_R_LUT3: GAM_R_GAIN_04[10:0] GAM_R_LUT4: GAM_R_GAIN_06[10:0] GAM_R_LUT5: GAM_R_GAIN_08[10:0] GAM_R_LUT6: GAM_R_GAIN_10[10:0] GAM_R_LUT7: GAM_R_GAIN_12[10:0] GAM_R_LUT8: GAM_R_GAIN_14[10:0] GAM_R_LUT9: GAM_R_GAIN_16[10:0] GAM_R_LUT10: GAM_R_GAIN_18[10:0] GAM_R_LUT11: GAM_R_GAIN_20[10:0] GAM_R_LUT12: GAM_R_GAIN_22[10:0] GAM_R_LUT13: GAM_R_GAIN_24[10:0] GAM_R_LUT14: GAM_R_GAIN_26[10:0] GAM_R_LUT15: GAM_R_GAIN_28[10:0] GAM_R_LUT16: GAM_R_GAIN_30[10:0]
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 0	*	1024	R/W	GAM_R_LUT1: Gain Adjustment of Area 1 of R Signal GAM_R_LUT2: Gain Adjustment of Area 3 of R Signal GAM_R_LUT3: Gain Adjustment of Area 5 of R Signal GAM_R_LUT4: Gain Adjustment of Area 7 of R Signal GAM_R_LUT5: Gain Adjustment of Area 9 of R Signal GAM_R_LUT6: Gain Adjustment of Area 11 of R Signal GAM_R_LUT7: Gain Adjustment of Area 13 of R Signal GAM_R_LUT8: Gain Adjustment of Area 15 of R Signal GAM_R_LUT9: Gain Adjustment of Area 17 of R Signal GAM_R_LUT10: Gain Adjustment of Area 19 of R Signal GAM_R_LUT11: Gain Adjustment of Area 21 of R Signal GAM_R_LUT12: Gain Adjustment of Area 23 of R Signal GAM_R_LUT13: Gain Adjustment of Area 25 of R Signal GAM_R_LUT14: Gain Adjustment of Area 27 of R Signal GAM_R_LUT15: Gain Adjustment of Area 29 of R Signal GAM_R_LUT16: Gain Adjustment of Area 31 of R Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times]) *: Bit Name GAM_R_LUT1: GAM_R_GAIN_01[10:0] GAM_R_LUT2: GAM_R_GAIN_03[10:0] GAM_R_LUT3: GAM_R_GAIN_05[10:0] GAM_R_LUT4: GAM_R_GAIN_07[10:0] GAM_R_LUT5: GAM_R_GAIN_09[10:0] GAM_R_LUT6: GAM_R_GAIN_11[10:0] GAM_R_LUT7: GAM_R_GAIN_13[10:0] GAM_R_LUT8: GAM_R_GAIN_15[10:0] GAM_R_LUT9: GAM_R_GAIN_17[10:0] GAM_R_LUT10: GAM_R_GAIN_19[10:0] GAM_R_LUT11: GAM_R_GAIN_21[10:0] GAM_R_LUT12: GAM_R_GAIN_23[10:0] GAM_R_LUT13: GAM_R_GAIN_25[10:0] GAM_R_LUT14: GAM_R_GAIN_27[10:0] GAM_R_LUT15: GAM_R_GAIN_29[10:0] GAM_R_LUT16: GAM_R_GAIN_31[10:0]

Note: This register is updated when GAM_R_VEN in GAM_R_UPDATE is 1.

39.2.24 Area Setting Register R1 in Gamma Correction Block (GAM_R_AREA1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GAM_R_TH_01[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_02[7:0]								GAM_R_TH_03[7:0]							
Initial value:	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GAM_R_TH_01[7:0]	8	R/W	Start Threshold of Area 1 of R Signal Unsigned (0 to 255 [LSB]) 0 < Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_02[7:0]	16	R/W	Start Threshold of Area 2 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_03[7:0]	24	R/W	Start Threshold of Area 3 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_R_VEN in GAM_R_UPDATE is 1.

39.2.25 Area Setting Register R2 in Gamma Correction Block (GAM_R_AREA2)

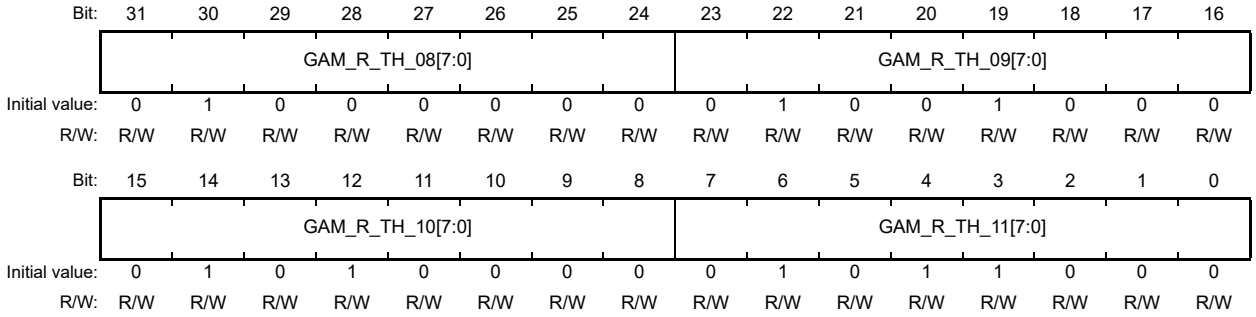
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_04[7:0]								GAM_R_TH_05[7:0]							
Initial value:	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_06[7:0]								GAM_R_TH_07[7:0]							
Initial value:	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_04[7:0]	32	R/W	Start Threshold of Area 4 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_05[7:0]	40	R/W	Start Threshold of Area 5 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_06[7:0]	48	R/W	Start Threshold of Area 6 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_07[7:0]	56	R/W	Start Threshold of Area 7 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_R_VEN in GAM_R_UPDATE is 1.

39.2.26 Area Setting Register R3 in Gamma Correction Block (GAM_R_AREA3)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_08[7:0]	64	R/W	Start Threshold of Area 8 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_09[7:0]	72	R/W	Start Threshold of Area 9 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_10[7:0]	80	R/W	Start Threshold of Area 10 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_11[7:0]	88	R/W	Start Threshold of Area 11 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_R_VEN in GAM_R_UPDATE is 1.

39.2.27 Area Setting Register R4 in Gamma Correction Block (GAM_R_AREA4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_12[7:0]								GAM_R_TH_13[7:0]							
Initial value:	0	1	1	0	0	0	0	0	0	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_14[7:0]								GAM_R_TH_15[7:0]							
Initial value:	0	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_12[7:0]	96	R/W	Start Threshold of Area 12 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_13[7:0]	104	R/W	Start Threshold of Area 13 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_14[7:0]	112	R/W	Start Threshold of Area 14 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_15[7:0]	120	R/W	Start Threshold of Area 15 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_R_VEN in GAM_R_UPDATE is 1.

39.2.28 Area Setting Register R5 in Gamma Correction Block (GAM_R_AREA5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_16[7:0]								GAM_R_TH_17[7:0]							
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_18[7:0]								GAM_R_TH_19[7:0]							
Initial value:	1	0	0	1	0	0	0	0	1	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_16[7:0]	128	R/W	Start Threshold of Area 16 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_17[7:0]	136	R/W	Start Threshold of Area 17 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_18[7:0]	144	R/W	Start Threshold of Area 18 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_19[7:0]	152	R/W	Start Threshold of Area 19 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_R_VEN in GAM_R_UPDATE is 1.

39.2.29 Area Setting Register R6 in Gamma Correction Block (GAM_R_AREA6)

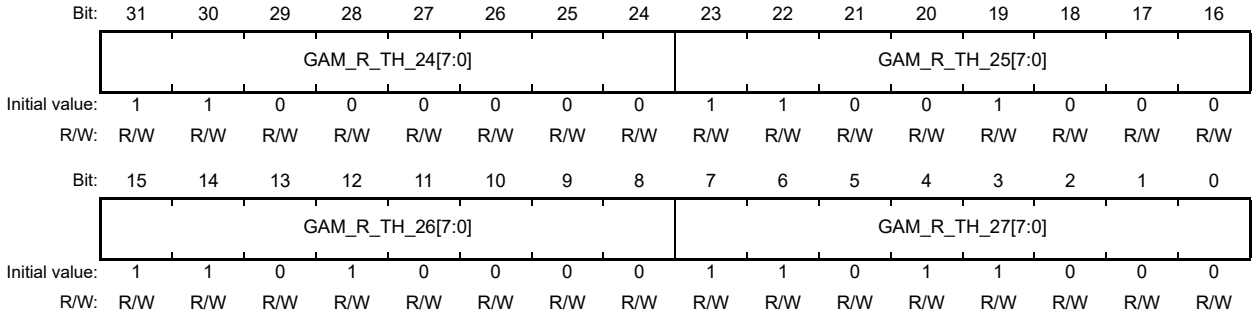
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_20[7:0]								GAM_R_TH_21[7:0]							
Initial value:	1	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_22[7:0]								GAM_R_TH_23[7:0]							
Initial value:	1	0	1	1	0	0	0	0	1	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_20[7:0]	160	R/W	Start Threshold of Area 20 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_21[7:0]	168	R/W	Start Threshold of Area 21 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_22[7:0]	176	R/W	Start Threshold of Area 22 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_23[7:0]	184	R/W	Start Threshold of Area 23 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_R_VEN in GAM_R_UPDATE is 1.

39.2.30 Area Setting Register R7 in Gamma Correction Block (GAM_R_AREA7)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_24[7:0]	192	R/W	Start Threshold of Area 24 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_25[7:0]	200	R/W	Start Threshold of Area 25 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_26[7:0]	208	R/W	Start Threshold of Area 26 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_27[7:0]	216	R/W	Start Threshold of Area 27 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area

Note: This register is updated when GAM_R_VEN in GAM_R_UPDATE is 1.

39.2.31 Area Setting Register R8 in Gamma Correction Block (GAM_R_AREA8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_28[7:0]								GAM_R_TH_29[7:0]							
Initial value:	1	1	1	0	0	0	0	0	1	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_30[7:0]								GAM_R_TH_31[7:0]							
Initial value:	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_28[7:0]	224	R/W	Start Threshold of Area 28 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_29[7:0]	232	R/W	Start Threshold of Area 29 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_30[7:0]	240	R/W	Start Threshold of Area 30 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_31[7:0]	248	R/W	Start Threshold of Area 31 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area < Threshold of current area ≤ 255

Note: This register is updated when GAM_R_VEN in GAM_R_UPDATE is 1.

39.2.32 TCON Register Update Control Register (TCON_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TCON_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TCON_VEN	0	R/WC1	LCD TCON Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

39.2.33 TCON Reference Timing Setting Register (TCON_TIM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_HALF[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_OFFSET[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_HALF[10:0]	400	R/W	1/2fH Timing Specifies the clock count from the rising edge of the Hsync signal as the counting timing of horizontal counter.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_OFFSET[10:0]	0	R/W	Offset Hsync Signal Timing Sets the clock cycle count from the rising edge of the Hsync signal.

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.34 TCON Vertical Timing Setting Register A1 (TCON_TIM_STVA1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_STVA_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_STVA_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_STVA_VS [10:0]	0	R/W	STVA Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STVA_VS from the rising edge of the Vsync signal (1/2fH cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_STVA_VW [10:0]	4	R/W	STVA Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STVA_VW (1/2fH cycles).

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.35 TCON Vertical Timing Setting Register A2 (TCON_TIM_STVA2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TCON_STVA_INV	—	TCON_STVA_SEL [2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_STVA_INV	1	R/W	Polarity Inversion Control of STVA Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_STVA_SEL [2:0]	0	R/W	Output Signal Select for LCD_TCON0 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.36 TCON Vertical Timing Setting Register B1 (TCON_TIM_STVB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_STVB_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_STVB_VW[10:0]										
Initial value:	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_STVB_VS [10:0]	70	R/W	STVB Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STVB_VS from the rising edge of the Vsync signal (1/2fH cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_STVB_VW [10:0]	960	R/W	STVB Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STVB_VW (1/2fH cycles).

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.37 TCON Vertical Timing Setting Register B2 (TCON_TIM_STVB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TCON_STVB_INV	—	TCON_STVB_SEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_STVB_INV	0	R/W	Polarity Inversion Control of STVB Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_STVB_SEL [2:0]	1	R/W	Output Signal Select for LCD_TCON1 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.38 TCON Horizontal Timing Setting Register STH1 (TCON_TIM_STH1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_STH_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_STH_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_STH_HS [10:0]	0	R/W	STH Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STH_HS + 1 from the rising edge of the Hsync signal (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_STH_HW [10:0]	96	R/W	STH Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STH_HW (clock cycles).

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.39 TCON Horizontal Timing Setting Register STH2 (TCON_TIM_STH2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TCON_STH_HS_SEL	—	—	—	TCON_STH_INV	—	TCON_STH_SEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	TCON_STH_HS_SEL	0	R/W	STH Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_STH_INV	1	R/W	Polarity Inversion Control of STH Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_STH_SEL[2:0]	2	R/W	Output Signal Select for LCD_TCON2 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.40 TCON Horizontal Timing Setting Register STB1 (TCON_TIM_STB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_STB_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_STB_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_STB_HS [10:0]	144	R/W	STB Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STB_HS + 1 from the rising edge of the Hsync signal (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_STB_HW [10:0]	640	R/W	STB Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STB_HW (clock cycles).

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.41 TCON Horizontal Timing Setting Register STB2 (TCON_TIM_STB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TCON_STB_HS_SEL	—	—	—	TCON_STB_INV	—	TCON_STB_SEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	TCON_STB_HS_SEL	0	R/W	STB Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_STB_INV	0	R/W	Polarity Inversion Control of STB Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_STB_SEL[2:0]	7	R/W	Output Signal Select for LCD_TCON3 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.42 TCON Horizontal Timing Setting Register CPV1 (TCON_TIM_CPV1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_CPV_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_CPV_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_CPV_HS [10:0]	0	R/W	CPV Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_CPV_HS + 1 from the rising edge of the Hsync signal (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_CPV_HW [10:0]	0	R/W	CPV Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_CPV_HW (clock cycles).

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.43 TCON Horizontal Timing Setting Register CPV2 (TCON_TIM_CPV2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TCON_CPV_HS_SEL	—	—	—	TCON_CPV_INV	—	TCON_CPV_SEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	TCON_CPV_HS_SEL	0	R/W	CPV Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_CPV_INV	0	R/W	Polarity Inversion Control of CPV Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_CPV_SEL[2:0]	4	R/W	Output Signal Select for LCD_TCON4 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.44 TCON Horizontal Timing Setting Register POLA1 (TCON_TIM_POLA1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_POLA_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_POLA_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_POLA_HS [10:0]	0	R/W	POLA Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_POLA_HS from the rising edge of the Hsync signal (clock cycles). Note: When 1 × 1, 1 × 2, or 2 × 2 reverse mode is selected, these bits should be set to1 or greater.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_POLA_HW [10:0]	0	R/W	POLA Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_POLA_HW (clock cycles).

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.45 TCON Horizontal Timing Setting Register POLA2 (TCON_TIM_POLA2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TCON_POLA_MD[1:0]		—	—	—	TCON_POLA_HS_SEL	—	—	—	TCON_POLA_INV	—	TCON_POLA_SEL[2:0]		
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1
R/W:	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	TCON_POLA_MD [1:0]	1	R/W	POLA Signal Generation Mode Select 0: Normal mode Generates the signal that changes twice a horizontal period. 1: 1 × 1 reverse mode Generates the signal whose polarity is inverted every horizontal period. 2: 1 × 2 reverse mode Generates the signal whose polarity is inverted in the first horizontal period and is subsequently inverted every two horizontal periods. 3: 2 × 2 reverse mode Generates the signal whose polarity is inverted every two horizontal periods.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	TCON_POLA_ HS_SEL	0	R/W	POLA Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_POLA_INV	0	R/W	Polarity Inversion Control of POLA Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_POLA_ SEL[2:0]	5	R/W	Output Signal Select for LCD_TCON5 pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.46 TCON Horizontal Timing Setting Register POLB1 (TCON_TIM_POLB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_POLB_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_POLB_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_POLB_HS [10:0]	0	R/W	POLB Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_POLB_HS from the rising edge of the Hsync signal (clock cycles). Note: When 1 × 1, 1 × 2, or 2 × 2 reverse mode is selected, these bits should be set to1 or greater.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_POLB_HW [10:0]	0	R/W	POLB Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_POLB_HW (clock cycles).

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.47 TCON Horizontal Timing Setting Register POLB2 (TCON_TIM_POLB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TCON_POLB_MD[1:0]		—	—	—	TCON_POLB_HS_SEL	—	—	—	TCON_POLB_IN	—	TCON_POLB_SEL[2:0]		
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0
R/W:	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	TCON_POLB_MD[1:0]	1	R/W	POLB Signal Generation Mode Select 0: Normal mode Generates the signal that changes twice a horizontal period. 1: 1 × 1 reverse mode Generates the signal whose polarity is inverted every horizontal period. 2: 1 × 2 reverse mode Generates the signal whose polarity is inverted in the first horizontal period and is subsequently inverted every two horizontal periods. 3: 2 × 2 reverse mode Generates the signal whose polarity is inverted every two horizontal periods.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	TCON_POLB_HS_SEL	0	R/W	POLB Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_POLB_INV	0	R/W	Polarity Inversion Control of POLB Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_POLB_SEL[2:0]	6	R/W	Output Signal Select for LCD_TCON6 pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.48 TCON Data Enable Polarity Setting Register (TCON_TIM_DE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TCON_DE_INV
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TCON_DE_INV	0	R/W	Polarity Inversion Control of DE Signal 0: Not inverted 1: Inverted

Note: This register is updated when TCON_VEN in TCON_UPDATE is 1.

39.2.49 Register Update Control Register in Output Controller (OUT_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OUTCNT_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OUTCNT_VEN	0	R/WC1	Brightness/Contrast, Dither Process, Output Interface Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

39.2.50 Output Interface Register (OUT_SET)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	OUT_ENDIAN_ON	—	—	—	OUT_SWAP_ON	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	OUT_FORMAT [1:0]		—	—	OUT_FRQ_SEL [1:0]		—	—	—	OUT_DIR_SEL	—	—	OUT_PHASE [1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	OUT_ENDIAN_ON	0	R/W	Bit Endian Change On/Off Control 0: Off 1: On
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	OUT_SWAP_ON	0	R/W	B/R Signal Swap On/Off Control 0: Off 1: On
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	OUT_FORMAT[1:0]	0	R/W	Output Format Select 0: RBG888 1: RGB666 2: RGB565 3: Serial RGB
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	OUT_FRQ_SEL [1:0]	0	R/W	Clock Frequency Control 0: 100% speed — (parallel RGB) 1: Triple speed — (serial RGB) 2: Quadruple speed — (serial RGB) 3: Setting prohibited
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OUT_DIR_SEL	0	R/W	Scan Direction Select 0: Forward scan 1: Reverse scan
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	OUT_PHASE[1:0]	0	R/W	Clock Phase Adjustment During Serial RGB Output Triple speed mode 0: 0 (clk) 1: 1 (clk) 2: 2 (clk) 3: Setting prohibited Quadruple speed mode 0: 0 (clk) 1: 1 (clk) 2: 2 (clk) 3: 3 (clk)

Note: This register is updated when OUTCNT_VEN in OUT_UPDATE is 1.

39.2.51 Brightness (DC) Correction Register 1 (OUT_BRIGHT1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PBRT_G[9:0]									
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	PBRT_G[9:0]	512	R/W	Brightness (DC) Adjustment of G Signal Unsigned (0 (-512) to 512 (0) to 1023 (+511) [LSB], 512 [LSB] with offset)

Note: This register is updated when OUTCNT_VEN in OUT_UPDATE is 1.

39.2.52 Brightness (DC) Correction Register 2 (OUT_BRIGHT2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PBRT_B[9:0]									
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PBRT_G[9:0]									
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 16	PBRT_B[9:0]	512	R/W	Brightness (DC) Adjustment of B Signal Unsigned (0 (-512) to 512 (0) to 1023 (+511) [LSB], 512 [LSB] with offset)
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	PBRT_R[9:0]	512	R/W	Brightness (DC) Adjustment of R Signal Unsigned (0 (-512) to 512 (0) to 1023 (+511) [LSB], 512 [LSB] with offset)

Note: This register is updated when OUTCNT_VEN in OUT_UPDATE is 1.

39.2.53 Contrast (Gain) Correction Register (OUT_CONTRAST)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CONT_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CONT_B[7:0]								CONT_R[7:0]							
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	CONT_G[7:0]	128	R/W	Contrast (Gain) Adjustment of G Signal 0/128 to 255/128 (approx.2 times)
15 to 8	CONT_B[7:0]	128	R/W	Contrast (Gain) Adjustment of B Signal 0/128 to 255/128 (approx.2 times)
7 to 0	CONT_R[7:0]	128	R/W	Contrast (Gain) Adjustment of R Signal 0/128 to 255/128 (approx.2 times)

Note: This register is updated when OUTCNT_VEN in OUT_UPDATE is 1.

39.2.54 Panel Dither Register (OUT_PDTHA)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	PDTH_SEL[1:0]		—	—	PDTH_FORMAT[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PDTH_PA[1:0]		—	—	PDTH_PB[1:0]		—	—	PDTH_PC[1:0]		—	—	PDTH_PD[1:0]	
Initial value:	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	1
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	PDTH_SEL[1:0]	0	R/W	Panel Dither Operation Mode 0: Truncate 1: Round-off 2: 2 × 2 pattern dither 3: Random pattern dither
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	PDTH_FORMAT [1:0]	0	R/W	Panel Dither Output Format Select 0: RGB888 1: RGB666 2: RGB565 3: Setting prohibited
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	PDTH_PA[1:0]	3	R/W	Pattern Value (A) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	PDTH_PB[1:0]	0	R/W	Pattern Value (B) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	PDTH_PC[1:0]	2	R/W	Pattern Value (C) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	PDTH_PD[1:0]	1	R/W	Pattern Value (D) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])

Note: This register is updated when OUTCNT_VEN in OUT_UPDATE is 1.

39.2.55 Output Phase Control Register (OUT_CLK_PHASE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	OUTCNT_FRONT_GAM	—	—	—	OUTCNT_LCD_EDGE	—	OUTCNT_STVA_EDGE	OUTCNT_STVB_EDGE	OUTCNT_STH_EDGE	OUTCNT_STB_EDGE	OUTCNT_CPV_EDGE	OUTCNT_POLA_EDGE	OUTCNT_POLB_EDGE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	OUTCNT_FRONT_GAM	0	R/W	Correction Circuit Sequence Control 0: Brightness → contrast → gamma correction 1: Gamma correction → brightness → contrast
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	OUTCNT_LCD_EDGE	0	R/W	Output Phase Control of LCD_DATA23 to LCD_DATA0 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	OUTCNT_STVA_EDGE	0	R/W	Output Phase Control of LCD_TCON0 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
5	OUTCNT_STVB_EDGE	0	R/W	Output Phase Control of LCD_TCON1 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
4	OUTCNT_STH_EDGE	0	R/W	Output Phase Control of LCD_TCON2 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
3	OUTCNT_STB_EDGE	0	R/W	Output Phase Control of LCD_TCON3 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
2	OUTCNT_CPV_EDGE	0	R/W	Output Phase Control of LCD_TCON4 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
1	OUTCNT_POLA_EDGE	0	R/W	Output Phase Control of LCD_TCON5 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
0	OUTCNT_POLB_EDGE	0	R/W	Output Phase Control of LCD_TCON6 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin

Note: This register is updated when OUTCNT_VEN in OUT_UPDATE is 1.

39.3 Usage Methods

39.3.1 Gamma Correction Adjustment Method

The characteristics of G, B, R of each panel to be connected should be measured and gamma correction should be made to suit the panel.

Since the gamma correction adjustment depends on to the characteristics of the panel, there are no recommended setting values.

39.3.2 Dither Usage Method

Dither is used when pseudo contour is appeared on the display screen.

Table 39.26 Dither Settings

Bit Name	Setting Value
PDTH_FORMAT[1:0]	Selects the format. For RGB888: 0 For RGB666: 1 For RGB565: 2
PDTH_SEL[1:0]	When 2 × 2 pattern dither is to be used: 2
PDTH_PA[1:0]	Normally 3 (initial value)
PDTH_PB[1:0]	Normally 0 (initial value)
PDTH_PC[1:0]	Normally 2 (initial value)
PDTH_PD[1:0]	Normally 1 (initial value)

39.3.3 Output Format Adjustment Method

The setting example of the typical output format is shown in Table 39.27 and Table 39.28.

It is necessary to carry out the setting of synchronization system signals of each output format similarly as the setting of output after scaling.

Table 39.27 Setting Example of Synchronizing Signal

Register Name	Bit Name	VGA	SVGA	Description
TCON_TIM	TCON_HALF[10:0]	400	528	Sets the half value of 1H time period in clock units.
Vsync signal				
TCON_TIM_STVA1	TCON_STVA_VS[10:0]	0	0	Sets the pulse generation start position from the rising edge of the internal Vsync signal. A 1/2H time period is set as 1.
TCON_TIM_STVA1	TCON_STVA_VW[10:0]	4	8	Sets the changing point from the above pulse generation start position. A 1/2H time period is set as 1.
TCON_TIM_STVA2	TCON_STVA_INV	0	0	Sets the output polarity of the above pulse. For inverted output: 1
TCON_TIM_STVA2	TCON_STVA_SEL[2:0]	0	0	For STVA output selection: 0
Vertical enable signal				
TCON_TIM_STVB1	TCON_STVB_VS[10:0]	68	44	Sets the pulse generation start position from the rising edge of the internal Vsync signal. A 1/2H time period is set as 1.
TCON_TIM_STVB1	TCON_STVB_VW[10:0]	960	1200	Sets the changing point from the above pulse generation start position. A 1/2H time period is set as 1.
TCON_TIM_STVB2	TCON_STVB_INV	0	0	Sets the output polarity of the above pulse. For inverted output: 1
TCON_TIM_STVB2	TCON_STVB_SEL[2:0]	1	1	For STVB output selection: 1
Hsync signal				
TCON_TIM_STH1	TCON_STH_HS[10:0]	0	0	Sets the pulse generation start position from the rising edge of the internal Hsync signal.
TCON_TIM_STH1	TCON_STH_HW[10:0]	96	128	Sets the changing point from the above pulse generation start position.
TCON_TIM_STH2	TCON_STH_INV	0	0	Sets the output polarity of the above pulse. For inverted output: 1
TCON_TIM_STH2	TCON_STH_SEL[2:0]	2	2	For STH output selection: 2
Horizontal enable signal				
TCON_TIM_STB1	TCON_STB_HS[10:0]	128	192	Sets the pulse generation start position from the rising edge of the internal Hsync signal.
TCON_TIM_STB1	TCON_STB_HW[10:0]	640	800	Sets the changing point from the above pulse generation start position.
TCON_TIM_STB2	TCON_STB_INV	0	0	Sets the output polarity of the above pulse. For inverted output: 1
TCON_TIM_STB2	TCON_STB_SEL[2:0]	3	3	For STB output selection: 3

Table 39.28 Setting Example of Data System

Register Name	Bit Name	RGB888	Serial RGB (Triple Speed)	Description
OUT_SET	OUT_ENDIAN_ON	0	0	When bit endian is changed: 1
OUT_SET	OUT_SWAP_ON	0	0	When B/R is to be swapped: 1
OUT_SET	OUT_PIXEL_INV_ON	0	0	When the function of reducing number of simultaneous changes is to be used: 1
OUT_SET	OUT_SUM_MOVE[4:0]	31	31	When OUT_PIXEL_INV_ON = 1, this register becomes valid. Sets the simultaneous changing threshold.
OUT_SET	OUT_FORMAT[1:0]	0	3	Sets the output format. For RGB888: 0 For RGB666: 1 For RGB565: 2 For serial RGB: 3
OUT_SET	OUT_FRQ_SEL[1:0]	0	1	Sets the output clock. For RGB888, RGB666, RGB565: 0 For triple speed serial RGB output: 1 For quadruple serial RGB output: 2
OUT_SET	OUT_DIR_SEL	0	0	When data arrangement of the serial RGB output is to be reversed: 1
OUT_SET	OUT_PHASE[1:0]	0	0	Sets when output phase of the serial RGB is shifted. Not delayed Delayed by one clock cycle: 1 Delayed by two clock cycles: 2 Delayed by three clock cycles: 3 (applicable only for quadruple speed mode)

40. Video Display Controller 6 (8): System Controller

40.1 System Controller

40.1.1 Overview of Functions

The system controller provides interrupt control, panel clock control, CLUT table read select signal status flag output functions.

40.1.2 Interrupt Control

Ten interrupt signals are output from the scaler and image synthesizer. The system controller controls whether to output these interrupt signals.

One is written to the corresponding INT_STA* bit when an interrupt signal is to be accepted. After 1 has been written to the bit, however, its value is still read out as 0 until the interrupt signal is accepted. Once the interrupt signal has been accepted, the INT_STA* bit is read out as 1.

The bit for an accepted interrupt signal is cleared by writing 0 to the INT_STA* bit. If further interrupt signals are to be accepted after the INT_STA* bit has been cleared, 1 is again written to the bit.

Table 40.1 Interrupt Signals

Request Source Name	Bit Name	Function
S0_VI_VSYNC	INT_STA 0	Vsync signal input to scaler 0
S0_LO_VSYNC	INT_STA 1	Vsync signal output from scaler 0
S0_VSYNCERR	INT_STA 2	Missing Vsync signal for scaler 0
GR3_VLINE	INT_STA 3	Specified line signal for panel output in graphics 3
S0_VFIELD	INT_STA 4	Field end signal for recording function in scaler 0
IV1_VBUFERR	INT_STA 5	Frame buffer write overflow signal for scaler 0
IV3_VBUFERR	INT_STA 6	Frame buffer read underflow signal for graphics 0
IV5_VBUFERR	INT_STA 7	Frame buffer read underflow signal for graphics 2
IV6_VBUFERR	INT_STA 8	Frame buffer read underflow signal for graphics 3
S0_WLINE	INT_STA 9	Write specification line signal input to scaling-down control block in scaler 0 (not usable in this product)

Table 40.2 Interrupt Clear/Hold Settings

Register Name	Bit Name	Initial Value	Description
SYSCNT_INT1	INT_STA0	0	S0_VI_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA1	0	S0_LO_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA2	0	S0_VSYNCERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA3	0	GR3_VLINE Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA4	0	S0_VFIELD Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA5	0	IV1_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA6	0	IV3_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA7	0	IV5_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT2	INT_STA8	0	IV6_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT2	INT_STA9	0	S0_WLINE Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. (prohibited setting in this product) 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.

Table 40.3 Interrupt Output On/Off Settings

Register Name	Bit Name	Initial Value	Description
SYSCNT_INT4	INT_OUT0_ON	0	S0_VI_VSYNC Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT1_ON	0	S0_LO_VSYNC Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT2_ON	0	S0_VSYNCERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT3_ON	0	GR3_VLINE Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT4_ON	0	S0_VFIELD Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT5_ON	0	IV1_VBUFERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT6_ON	0	IV3_VBUFERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT7_ON	0	IV5_VBUFERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT5	INT_OUT8_ON	0	IV6_VBUFERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT5	INT_OUT9_ON	0	S0_WLINE Interrupt Output On/Off 0: Off 1: On (prohibited setting in this product)

40.1.3 Panel Clock Control

The video image clock, external clock, or peripheral clock 1 is selectable as the source of the panel clock for supply to this module. The module is also equipped with a divider for scaling the frequency by factors from 1/1 to 1/32. The bits listed in Table 40.4 control the panel clock.

Table 40.4 Panel Clock Control

Register Name	Bit Name	Initial Value	Description
SYSCNT_PANEL_CLK	PANEL_ICKSEL [1:0]	0	Divided Clock Source Select 0: Video image clock (this is DV_CLK if INP_SEL = 1) 1: External clock (LCD0_EXTCLK) 2: Setting prohibited 3: Peripheral clock 1 (P1φ)
SYSCNT_PANEL_CLK	PANEL_OCKSEL [1:0]	0	Panel Clock Output Select 0: Divided clock generated from clock source selected by PANEL_CLK[1:0] 1: LVDS PLL clock 2: LVDS PLL clock divided by 7 3: Setting prohibited
SYSCNT_PANEL_CLK	PANEL_ICKEN	0	Panel Clock Operation Enable 0: Disables operation of the panel clock operation block. 1: Enables operation of the panel clock operation block. Note: Set this bit to 0 before changing the value of the PANEL_ICKSEL or PANEL_DCDR bits.*
SYSCNT_PANEL_CLK	PANEL_DCDR [5:0]	1	Clock Frequency Division Ratio Refer to Table 40.5 for details on the settings. Note: Settings other than those in Table 40.5 are prohibited.

Note: * Set this bit to 0 before changing the value of the LVDS_CLK_EN bit in LVDS_CLKSELR.

Table 40.5 I/O Clock Frequency and Divisors

		I/O Clock Frequency (MHz)		
DCDR[5:0]	Clock Ratio	27.00	54.00	66.67
000001	1/1	27.00	54.00	66.67
000010	1/2	13.50	27.00	33.33
000011	1/3	9.00	18.00	22.22
000100	1/4	6.75	13.50	16.67
000101	1/5	5.40	10.80	13.33
000110	1/6	4.50	9.00	11.11
000111	1/7	3.86	7.71	9.52
001000	1/8	3.38	6.75	8.33
001001	1/9	3.00	6.00	7.41
001100	1/12	2.25	4.50	5.56
010000	1/16	1.69	3.38	4.17
011000	1/24	1.13	2.25	2.78
100000	1/32	0.84	1.69	2.08

40.1.4 CLUT Table Read Select Signal Status Flag

The CLUT read select signal status can be read using the flags shown in Table 40.6.

Table 40.6 CLUT Table Read Select Signal Status Flags

Register Name	Bit Name	Initial Value	Description
SYSCNT_CLUT	GR0_CLT_SEL_ST	—	Graphics 0 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.
SYSCNT_CLUT	GR2_CLT_SEL_ST	—	Graphics 2 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.
SYSCNT_CLUT	GR3_CLT_SEL_ST	—	Graphics 3 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.

40.2 Register Descriptions

The following register sets are allocated in the SH register map area.

- Symbols used in Register Descriptions

Initial value: Register value after a reset
—: Undefined value
R/W: Readable/writable. The written value can be read.
R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.
R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.
R: Read-only. The write value should always be 0.
—/W: Write-only. The read value is undefined.

Table 40.7 Register Configuration of System Controller (CH0)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt control register 1	SYSCNT_INT1	R/W	H'0000 0000	H'FCFF 7A80	32
Interrupt control register 2	SYSCNT_INT2	R/W	H'0000 0000	H'FCFF 7A84	32
Interrupt control register 4	SYSCNT_INT4	R/W	H'0000 0000	H'FCFF 7A8C	32
Interrupt control register 5	SYSCNT_INT5	R/W	H'0000 0000	H'FCFF 7A90	32
Panel clock control register	SYSCNT_PANEL_CLK	R/W	H'0001	H'FCFF 7A98	16
CLUT table read select signal status register	SYSCNT_CLUT	R	H'0000	H'FCFF 7A9A	16

40.2.1 Interrupt Control Register 1 (SYSCNT_INT1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	INT_STA7	—	—	—	INT_STA6	—	—	—	INT_STA5	—	—	—	INT_STA4
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	INT_STA3	—	—	—	INT_STA2	—	—	—	INT_STA1	—	—	—	INT_STA0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	INT_STA7	0	R/W	IV5_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	INT_STA6	0	R/W	IV3_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	INT_STA5	0	R/W	IV1_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	INT_STA4	0	R/W	S0_VFIELD Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	INT_STA3	0	R/W	GR3_VLINE Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INT_STA2	0	R/W	S0_VSYNCERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	INT_STA1	0	R/W	S0_LO_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INT_STA0	0	R/W	S0_VI_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.

40.2.2 Interrupt Control Register 2 (SYSCNT_INT2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	$\overline{\text{INT\_STA9}}$	—	—	—	$\overline{\text{INT\_STA8}}$
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INT_STA9	0	R/W	S0_WLINE Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. (prohibited setting in this product) 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INT_STA8	0	R/W	IV6_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.

40.2.3 Interrupt Control Register 4 (SYSCNT_INT4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	INT_OUT7_ON	—	—	—	INT_OUT6_ON	—	—	—	INT_OUT5_ON	—	—	—	INT_OUT4_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	INT_OUT3_ON	—	—	—	INT_OUT2_ON	—	—	—	INT_OUT1_ON	—	—	—	INT_OUT0_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	INT_OUT7_ON	0	R/W	IV5_VBUFERR Interrupt Output On/Off 0: Off 1: On
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	INT_OUT6_ON	0	R/W	IV3_VBUFERR Interrupt Output On/Off 0: Off 1: On
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	INT_OUT5_ON	0	R/W	IV1_VBUFERR Interrupt Output On/Off 0: Off 1: On
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	INT_OUT4_ON	0	R/W	S0_VFIELD Interrupt Output On/Off 0: Off 1: On
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	INT_OUT3_ON	0	R/W	GR3_VLINE Interrupt Output On/Off 0: Off 1: On
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INT_OUT2_ON	0	R/W	S0_VSYNCERR Interrupt Output On/Off 0: Off 1: On
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INT_OUT1_ON	0	R/W	S0_LO_VSYNC Interrupt Output On/Off 0: Off 1: On
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INT_OUT0_ON	0	R/W	S0_VI_VSYNC Interrupt Output On/Off 0: Off 1: On

40.2.4 Interrupt Control Register 5 (SYSCNT_INT5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	INT_OUT9_ON	—	—	—	INT_OUT8_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INT_OUT9_ON	0	R/W	S0_WLINE Interrupt Output On/Off 0: Off 1: On (prohibited setting in this product)
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INT_OUT8_ON	0	R/W	IV6_VBUFERR Interrupt Output On/Off 0: Off 1: On

40.2.5 Panel Clock Control Register (SYSCNT_PANEL_CLK)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PANEL_ICKSEL [1:0]		PANEL_OCKSEL [1:0]		—	PANEL_ICKEN	—	—	PANEL_DCDR[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R/W	R/W	R/W	R/W	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	PANEL_ICKSEL [1:0]	All 0	R/W	Divided Clock Source Select 0: Video image clock (this is DV_CLK if INP_SEL = 1) 1: External clock (LCD0_EXTCLK) 2: Setting prohibited 3: Peripheral clock 1 (P1φ)
11, 10	PANEL_OCKSEL [1:0]	All 0	R/W	Panel Clock Output Select 0: Divided clock generated from clock source selected by PANEL_CLK[1:0] 1: LVDS PLL clock 2: LVDS PLL clock divided by 7 3: Setting prohibited
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	PANEL_ICKEN	0	R/W	Panel Clock Operation Enable 0: Disables operation of the panel clock operation block. 1: Enables operation of the panel clock operation block. Note: Set this bit to 0 before changing the value of the PANEL_ICKSEL or PANEL_DCDR bits.*1
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	PANEL_DCDR [5:0]	1	R/W	Clock Frequency Division Ratio Refer to Table 40.5 for details on the settings. Note: Settings other than those in Table 40.5 are prohibited.

Note: * Set this bit to 0 before changing the value of the LVDS_CLK_EN bit in LVDS_CLKSELR.

40.2.6 CLUT Table Read Select Signal Status Register (SYSCNT_CLUT)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	GR3_CLT_SEL_ST	—	—	—	GR2_CLT_SEL_ST	—	—	—	—	—	—	—	GR0_CLT_SEL_ST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	GR3_CLT_SEL_ST	—	R	Graphics 3 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR2_CLT_SEL_ST	—	R	Graphics 2 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR0_CLT_SEL_ST	—	R	Graphics 0 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.

41. LVDS Output Interface

This LSI has a module for providing a low voltage differential signaling (LVDS) output interface. This module converts the RGB signals output from video display controller 6 to the LVDS format that has six bits each for R, G, and B signals, and outputs the converted signals. It has a dedicated phase-locked loop (PLL) circuit, which can generate a clock with a desired frequency by dividing or multiplying the input clock frequency.

The clock output from this PLL can be used as the panel clock for video display controller 6 even when LVDS output is not used. This PLL is called LVDS PLL in this section.

41.1 Features

- Four pairs of differential output conforming to the TIA/EIA-644 standard (three pairs for data and one pair for the clock)
- When LVDS output is not used, the LVDS pins can be used as CMOS input/output pins.
- LVDS PLL for generating a clock with a desired frequency
- LVDS PLL power-down function

41.2 Block Diagram

Figure 41.1 shows a block diagram of the LVDS output interface configuration.

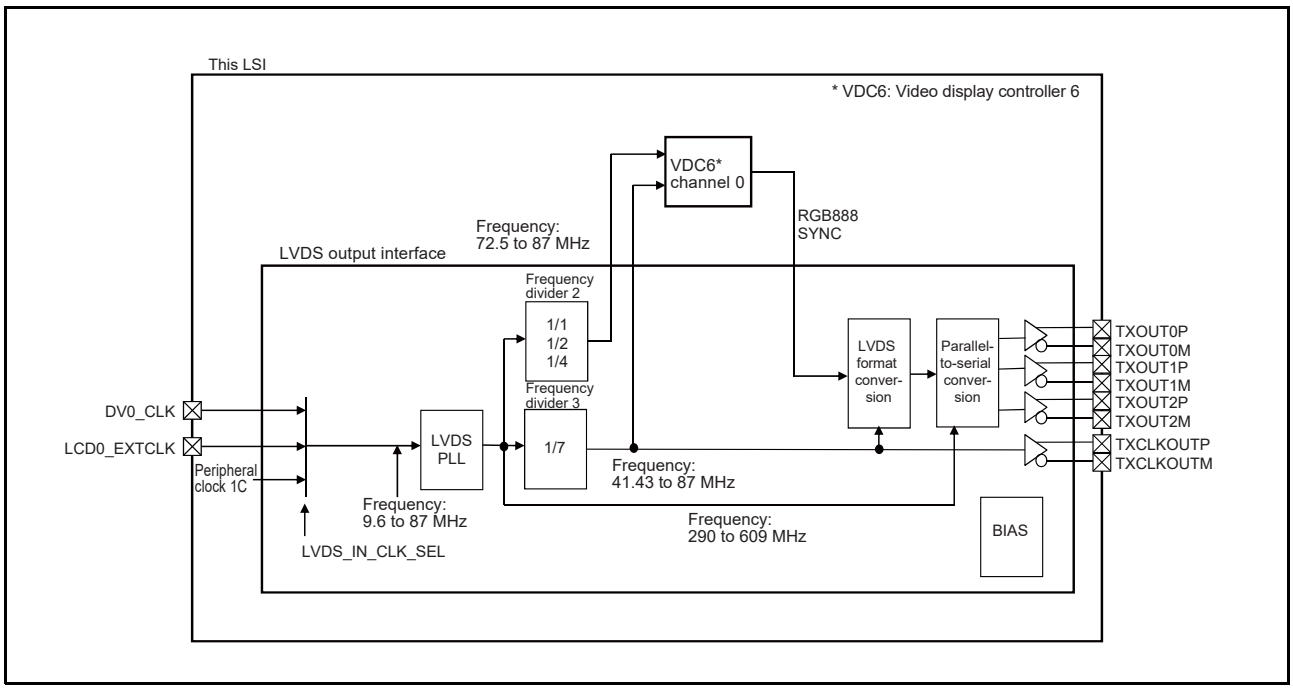


Figure 41.1 Block Diagram of LVDS Output Interface Configuration

Figure 41.2 shows a block diagram of the LVDS PLL circuit.

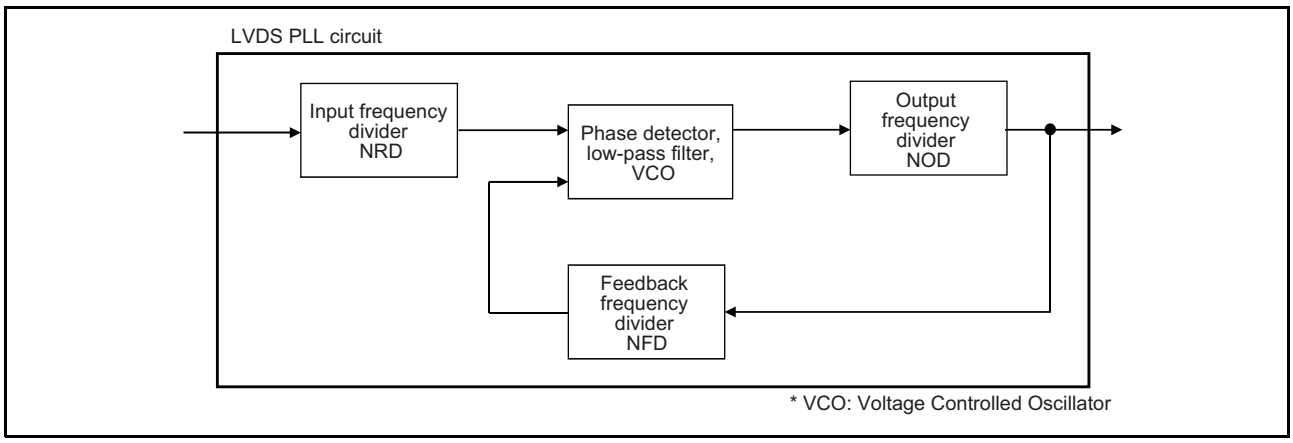


Figure 41.2 Block Diagram of LVDS PLL Circuit

41.3 Input/Output Pins

Table 41.1 shows the pin configuration.

Table 41.1 Pin Configuration

Pin Name	Symbol	I/O	Function
LVDS analog power supply pin	LVDSAPVcc	Input	Power supply pin for LVDS output
LVDS PLL power supply pin	LVDSPLLVcc	Input	Power supply pin for LVDS PLL
LVDS data output pin 0P	TXOUT0P	Output	LVDS data output pins
LVDS data output pin 0M	TXOUT0M	Output	
LVDS data output pin 1P	TXOUT1P	Output	
LVDS data output pin 1M	TXOUT1M	Output	
LVDS data output pin 2P	TXOUT2P	Output	
LVDS data output pin 2M	TXOUT2M	Output	
LVDS clock output pin CP	TXCLKOUTP	Output	LVDS clock output pins
LVDS clock output pin CM	TXCLKOUTM	Output	

41.4 Register Descriptions

Table 41.2 shows the register configuration.

Table 41.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
LVDS register update control register	LVDS_UPDATE	R/W	H'00000000	H'FCFF7A30	32
LVDS format conversion register L	LVDSFCL	R/W	H'00000321	H'FCFF7A34	32
LVDS clock select register	LCLKSELR	R/W	H'00000000	H'FCFF7A50	32
LVDSPLL setting register	LPLLSETR	R/W	H'00000001	H'FCFF7A54	32

41.4.1 LVDS Register Update Control Register (LVDS_UPDATE)

LVDS_UPDATE controls the timing for updating LVDS registers.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LVDS_UPDATE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	LVDS_ UPDATE	0	R/W	LVDS Register Update 0: Registers are not updated. 1: Registers are updated.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

41.4.2 LVDS Format Conversion Register L (LVDSFCL)

LVDSFCL is used to make necessary settings for conversion of the RGB signals output from video display controller 6 to the LVDS format.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SYNC_MODE	—	—	—	—	SYNC_POL[1:0]		—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R/W	R/W	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	LVDS_SEL2[3:0]				LVDS_SEL1[3:0]				LVDS_SEL0[3:0]			
Initial value:	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	1
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	SYNC_MODE	0	R/W	SYNC Mode Switch Switches the SYNC mode. 0: LVDS_TCON0 and LVDS_TCON2 are transmitted through LVDS signals. 1: Fixed values are transmitted as LVDS_TCON0 and LVDS_TCON2 through LVDS signals. The fixed values are specified in the SYNC_POL bits.
27 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23, 22	SYNC_POL [1:0]	00	R/W	LVDS_TCON0 and LVDS_TCON2 Value The values specified in these bits are output as the LVDS_TCON0 and LVDS_TCON2 values through LVDS signals when the SYNC_MODE bit is set to 1. SYNC_POL[1]: Specifies the LVDS_TCON0 output value. SYNC_POL[0]: Specifies the LVDS_TCON2 output value.
21 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	LVDS_SEL2 [3:0]	0011	R/W	TXOUT2P/M Data Format Setting These bits specify the format of the data to be output through the TXOUT2P and TXOUT2M pins. Be sure to set these bits to 3 in this LSI.
7 to 4	LVDS_SEL1 [3:0]	0010	R/W	TXOUT1P/M Data Format Setting These bits specify the format of the data to be output through the TXOUT1P and TXOUT1M pins. Be sure to set these bits to 2 in this LSI.
3 to 0	LVDS_SEL0 [3:0]	0001	R/W	TXOUT0P/M Data Format Setting These bits specify the format of the data to be output through the TXOUT0P and TXOUT0M pins. Be sure to set these bits to 1 in this LSI.

Note: This register is updated by setting the LVDS_UPDATE bit in the LVDS register update control register (LVDS_UPDATE) to 1.

41.4.3 LVDS Clock Select Register (LCLKSELR)

LCLKSELR specifies the input clock and the frequency dividing values for frequency divider 2.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	LVDS_IN_CLK_SEL[1:0]		—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LVDSPLL_TST[5:0]						LVDS_ODIV_SET[1:0]		—	—	—	LVDS_CLK_EN	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 24	LVDS_IN_CLK_SEL[1:0]	All 0	R/W	These bits select the clock input to LVDSPLL. 0: DV0_CLK 1: LCD0_EXTCLK 2: Peripheral clock 1C (P1φ) 3: Setting prohibited
23 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 10	LVDSPLL_TST[5:0]	All 0	R/W	Internal Parameter Setting for LVDS PLL These bits specify the internal parameters for the LVDS PLL (the resistance of the LPF circuit and the current through the charge pump circuit). Note: Do not rewrite from 000000 in this LSI.
9, 8	LVDS_ODIV_SET[1:0]	All 0	R/W	These bits specify the frequency dividing value (NODIV) for frequency divider 2. NODIV = 1, 2, or 4 0: NODIV = 1 1: NODIV = 2 2: NODIV = 4 3: Setting prohibited
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	LVDS_CLK_EN	0	R/W	Enables the LVDS PLL output. 0: Output from the PLL is disabled. 1: Output from the PLL is enabled. Note: Be sure to clear this bit to 0 before making settings in any other register described in this section. In addition, before modifying this bit, be sure to clear the PANEL_ICKEN bits in SYSCNT_PANEL_CLK in video display controller 6 to 0.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

41.4.4 LVDS PLL Setting Register (LPLLSETR)

LPLLSETR specifies the frequency dividing value and controls power-down in the LVDS PLL.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	LVDSPLL_FD[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	LVDSPLL_RD[2:0]			—	—	LVDSPLL_OD[1:0]		—	—	—	LVDSPLL_PD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 16	LVDSPLL_FD[6:0]	All 0	R/W	These bits specify the frequency dividing value (NFD) for the feedback frequency in the LVDS PLL. NFD = 23 to 63 22 to 62: NFD = LVDSPLL_FD value + 1 0 to 21 and 63 to 127: Setting prohibited
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	LVDSPLL_RD[2:0]	All 0	R/W	These bits specify the frequency dividing value (NRD) for the input frequency in the LVDS PLL. NRD = 1 to 8 0 to 7: NRD = LVDSPLL_RD value + 1
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	LVDSPLL_OD[1:0]	All 0	R/W	These bits specify the frequency dividing value (NOD) for the output frequency in the LVDS PLL. NOD = 1, 2, 4, or 8 0 to 3: NOD = 2 ^ LVDSPLL_OD value Note: Be sure to set this bit to 1 in this LSI.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	LVDSPLL_PD	1	R/W	Controls power-down for the LVDS PLL. 0: Normal operation 1: Power-down state

41.5 Operation

This module receives the RGB888 signals output from video display controller 6 and the SYNC signal, converts the video signals to the LVDS format, and outputs them. The precision of the RGB signals is reduced to six bits each. The operating clock is generated in the LVDS PLL.

41.5.1 LVDS PLL Settings

The input clock select circuit, frequency dividers 2 and 3, and LVDS PLL circuit are used to generate operating clocks. For input and output of each circuit, allowable frequency ranges are determined; be sure to make appropriate settings to generate input and output frequencies within the ranges.

Before modifying various setting registers, check that the panel clock operation in video display controller 6 is disabled (SYSCNT_PANEL_CLK.PANEL_ICKEN = 0) and the LVDS PLL output is disabled (LPLLSETR.LVDS_CLK_EN = 0).

Figure 41.3 shows a block diagram of the LVDS PLL and its peripheral circuits.

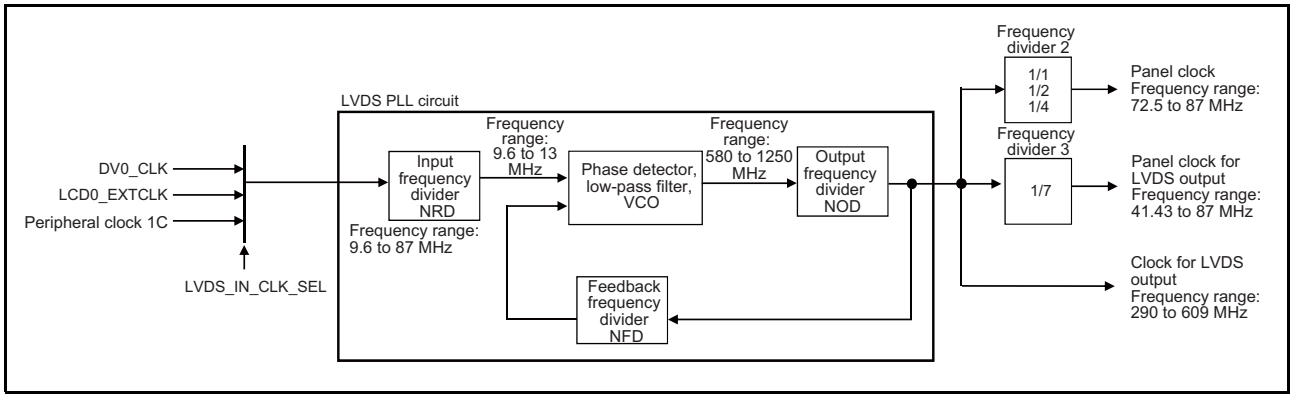


Figure 41.3 Block Diagram of LVDS PLL and Peripheral Circuits

(1) Clock Input to LVDS PLL

The clock input to the LVDS PLL can be selected from among DV0_CLK, LCD0_EXTCLK, and peripheral clock 1C (P1φ) clocks through the LVDS_IN_CLK_SEL bit setting in LVDS_CLKSELR.

The frequency of this input clock should be within the range from 9.6 to 87 MHz.

(2) LVDS PLL Frequency Setting

The LVDS PLL has three internal frequency dividers: input frequency divider NRD, feedback frequency divider NFD, and output frequency divider NOD.

The frequency of the clock output from the LVDS PLL is calculated by the following equations. Note that the reference frequency, VCO output frequency, LVDS PLL input frequency, and LVDS PLL output frequency should be within the respective ranges shown below the equations.

FREF = FIN / NRD (Reference frequency)
FVCO = FIN × NFD × NOD / NRD (VCO output frequency)
FOUT = FIN × NFD / NRD (LVDS PLL output frequency)

Note: FREF: Reference frequency. Frequency range = 9.6 MHz to 13 MHz
FVCO: VCO output frequency. Frequency range = 580 MHz to 1250 MHz
FIN: LVDS PLL input frequency. Frequency range = 9.6 MHz to 87 MHz
FOUT: LVDS PLL output frequency. Frequency range = 609 MHz max.

(3) LVDS PLL Frequency Setting Timing

Before making settings for each frequency divider in the PLL circuit described in (2), place the PLL in the power-down state (PD = 1) and wait for at least 0.5 µs. After making frequency divider settings, wait for a retention period of at least 0.5 µs and release the power-down state (PD = 0).

Figure 41.4 shows this timing.

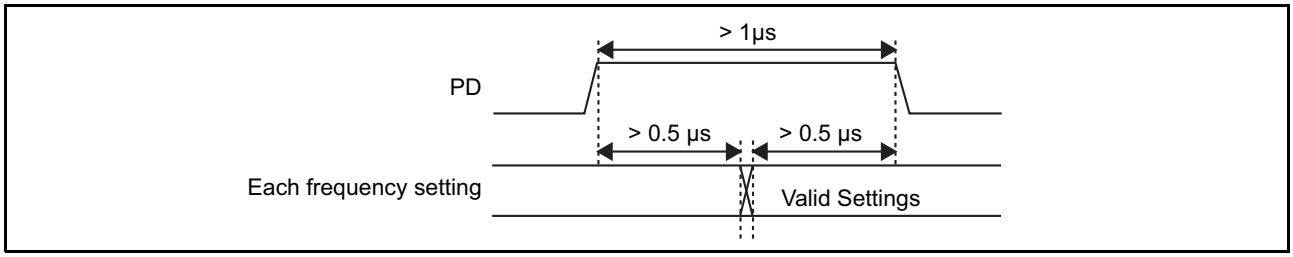


Figure 41.4 PLL Setting Timing

(4) LVDS PLL Output Enable Timing

Output of the clock from the LVDS PLL can be enabled or disabled through the LVDS_CLK_EN bit setting in LCLKSELR.

To allow time for stabilization of oscillation by the PLL, wait for at least 200 us after release from the power-down state before enabling the LVDS PLL output. The LVDS_CLK_EN bit in LCLKSELR should be modified only while the panel clock operation in video display controller 6 is disabled (SYSCNT_PANEL_CLK.PANEL_ICKEN = 0).

(5) LVDS PLL Output Clock Frequency Setting

When using the clock output from the LVDS PLL as the clock for the LVDS output interface, select the clock from frequency divider 3 (divided by 7) as the panel clock for video display controller 6.

When using the clock output from the LVDS PLL as the clock for digital RGB output, select the clock from frequency divider 2 as the panel clock for video display controller 6. In addition, make appropriate settings so that the clock frequency after frequency division is 87 MHz or lower. This limitation on the frequency is not applied when the clock is used for the LVDS output interface.

The panel clock can be selected through the PANEL_OCKSEL bits in SYSCNT_PANEL_CLK in video display controller 6.

41.5.2 LVDS Output Format

This module converts and then outputs the LCD data output from video display controller 6 and the timing signals generated by the LCD TCON.

(1) LVDS Output Format

(a) LCD data

After converted into the LVDS format, LCD data is output from LCD_DATA23 to LCD_DATA18, LCD_DATA15 to LCD_DATA10, and LCD_DATA7 to LCD_DATA2 pins.

(b) Timing signals generated by LCD TCON

After converted into the LVDS format, the timing signals are output from the LCD_TCON0, LCD_TCON2, and LCD_TCON3 pins.

Figure 41.5 shows the bit alignment in the LVDS output format.

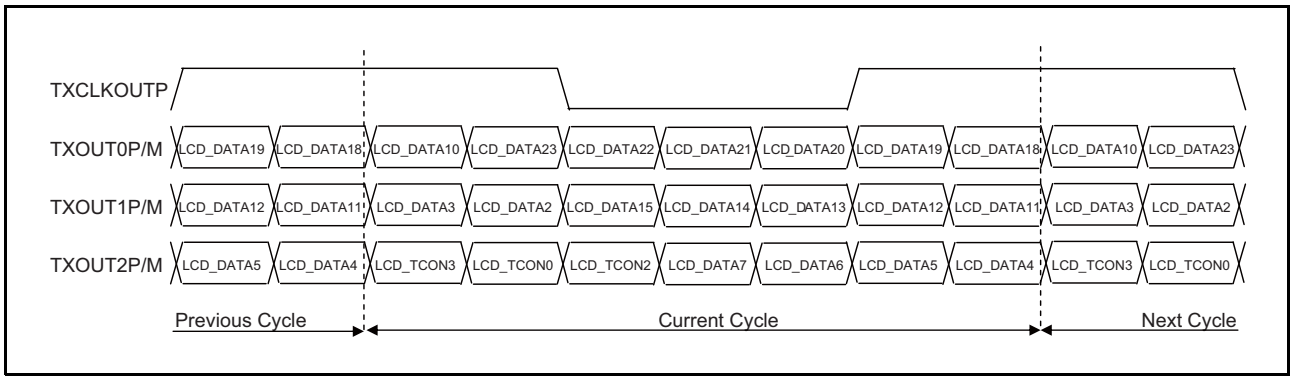


Figure 41.5 Data Map of LVDS Output Format

(2) Setting Example

The following is the setting example of the LVDS output format. The register the setting is applied is register in video display controller 6. For the details of the register, see section 39, Video Display Controller 6 (7): Output Controller.

Table 41.3 Setting Example of LVDS Output Format

Register	Bit	Value	Remark
OUT_SET	OUT_ENDIAN_ON	0	Bit endian change is off.
	OUT_SWAP_ON	0	B/R signal swap is off.
	OUT_FORMAT[1:0]	0	RGB888 is selected as an output format (the upper six-bit data is output in the LVDS format).
TCON_TIM_STVA2	TCON_TIM_STVA2	0	The VS signal is selected for the LCD_TCON0 pin.
TCON_TIM_STVA1	TCON_STVA_VS[10:0]	Arbitrary	The VS signal timing is set.
	TCON_STVA_VW[10:0]	Arbitrary	The VS signal timing is set.
TCON_TIM_STH2	TCON_STH_SEL[2:0]	2	The HS signal is selected for the LCD_TCON2 pin.
TCON_TIM_STH1	TCON_STH_HS[10:0]	Arbitrary	The HS signal timing is set.
	TCON_STH_VS[10:0]	Arbitrary	The HS signal timing is set.
TCON_TIM_STB2	TCON_STB_SEL[2:0]	7	The DE signal is selected for the LCD_TCON3 pin.
TCON_TIM_STVB1	TCON_STVB_VS[10:0]	Arbitrary	The DE signal timing is set.
	TCON_STVB_VW[10:0]	Arbitrary	The DE signal timing is set.
TCON_TIM_STB1	TCON_STB_HS[10:0]	Arbitrary	The DE signal timing is set.
	TCON_STB_HW[10:0]	Arbitrary	The DE signal timing is set.

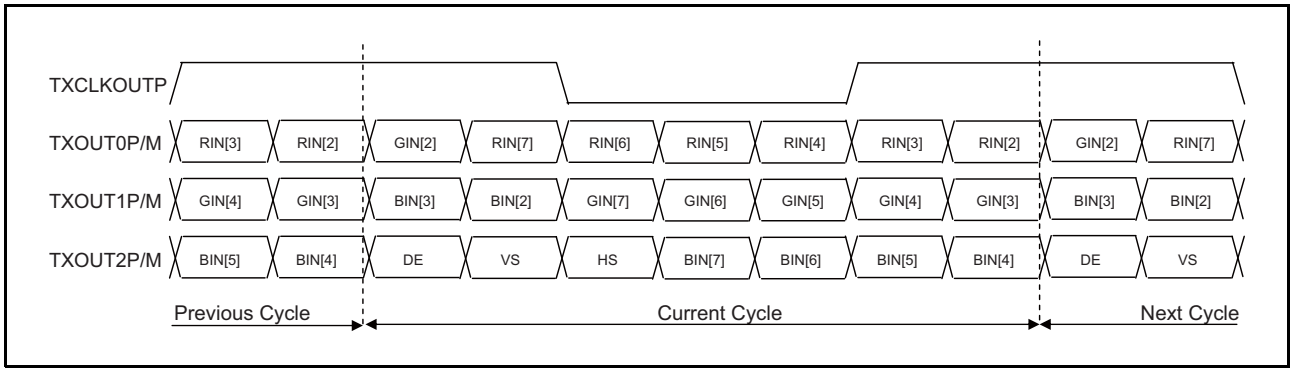


Figure 41.6 Data Map of Setting Example of LVDS Output Format

41.5.3 Procedures for Register Settings

(1) Initial Settings after Power-on Reset

The following shows an example of initial settings after a power-on reset.

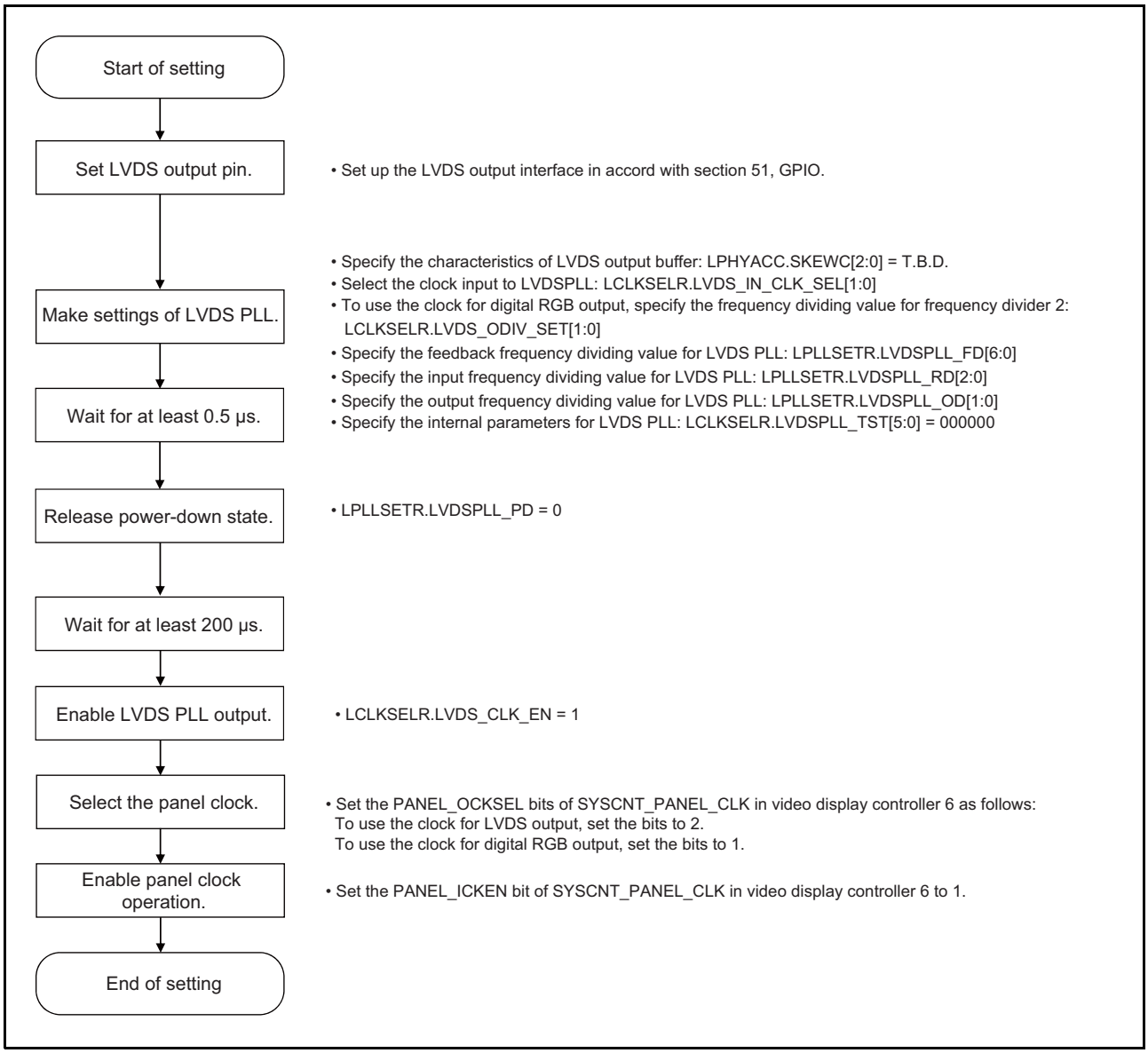


Figure 41.7 Example of Initial Settings after Power-On Reset

(2) Frequency Modification Settings

The following shows an example of modifying frequencies.

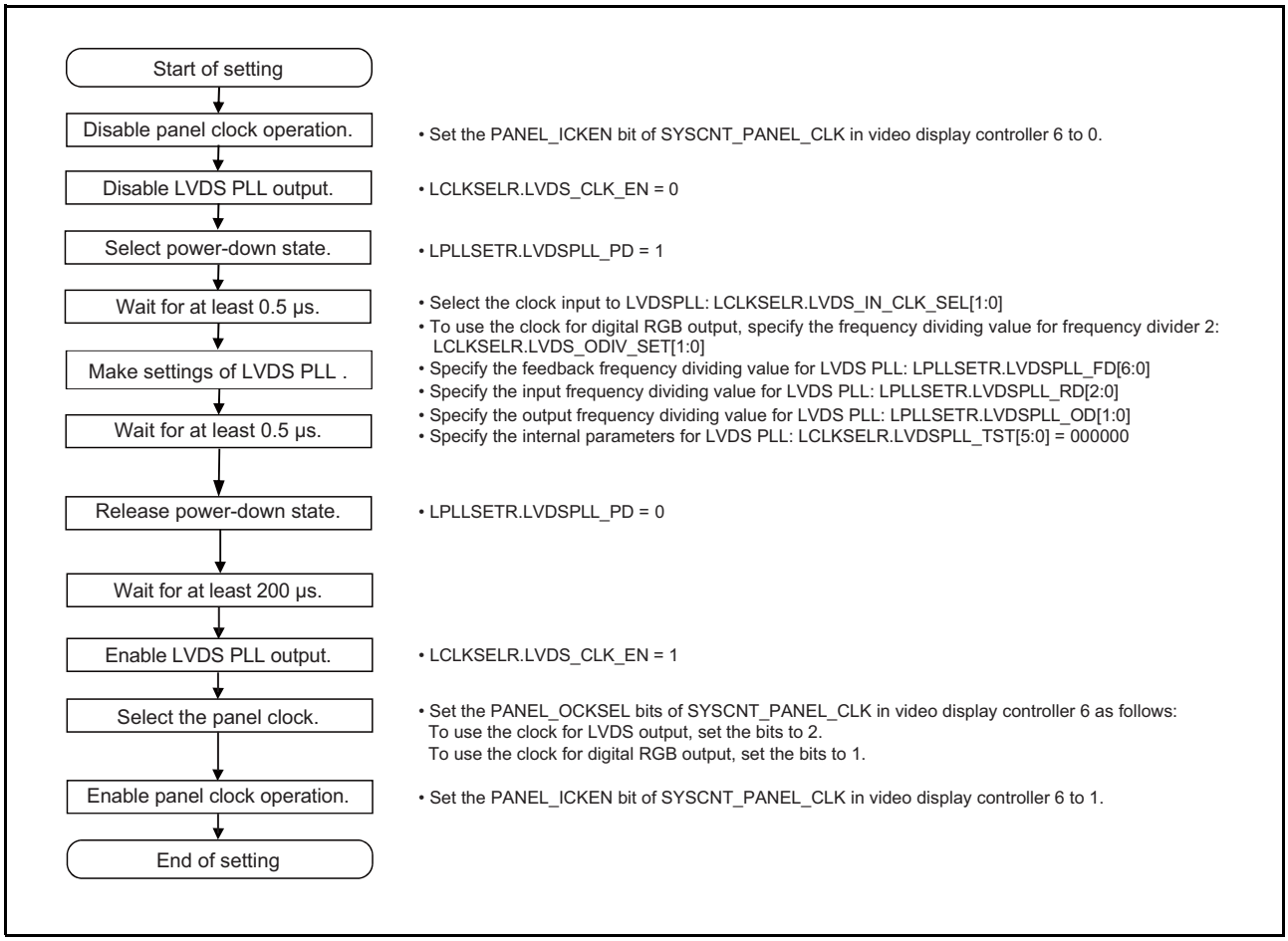


Figure 41.8 Example of Modifying Frequencies for LVDS Output

41.6 Notes

41.6.1 LVDS Output Pin Settings

The LVDS output buffers enter a low power consumption state, even if one of the eight differential output pins is set up other than for an LVDS output interface in accordance with section 51, GPIO. Although the LVDS output buffers return from the low power consumption state after pins are set to operate as an LVDS output interface, this requires up to 0.5 µs. Accordingly, at least 0.5 µs are required before actual LVDS output starts after pins have been set to operate as an LVDS output interface.

42. Image Renderer (IMR-LS2)

The distortion correction engine, image render-light SRAM 2 (IMR-LS2) is a drawing processor with a simple instruction system. The IMR-LS2 refers to captured video data as two-dimensional texture data and draws shapes by performing texture mapping to desired shapes divided into triangular objects. The IMR-LS2 is also capable of working with the video input interface in drawing in real time. Accordingly, the IMR-LS2 is usable in handling various functions such as for flattening the distortion of images from a camera with a fish-eye lens by processing the shape of the images and for correcting other desired types of images.
Details are available upon entry into a non-disclosure agreement.

43. 2D Drawing Engine (DRW)

43.1 Overview

The 2D Drawing Engine (DRW) provides flexible functions that can support almost any object geometry, rather than being bound to only a few specific geometries such as lines, triangles, or circles. The edges of every object can be independently blurred or anti-aliased.
Rasterization is executed on the bounding box of an object from left to right and top to bottom at a rate of one pixel per clock cycle. The 2D Drawing Engine can also raster from bottom to top in certain cases, to optimize the performance. In addition, optimization methods are provided to avoid rasterization of many empty pixels of the bounding box.
The distances to the edges of the object are calculated by a set of edge equations for every pixel of the bounding box. These edge equations can be combined to describe the entire object. If a pixel is inside the object, it is selected for rendering, and if it is outside it is discarded. If a pixel is on an edge, an alpha value for anti-aliasing is chosen proportional to the distance between the pixel (the edge) and another pixel nearest to the edge. Every pixel that is selected for rendering can be textured. The resulting aRGB quadruple can be modified by a general raster operation approach independently for each of the four channels. The aRGB quadruple can then be blended with one of the multiple blend modes of the module.
The 2D Drawing Engine provides two inputs (texture read and framebuffer read) and one o
utput (framebuffer write). The internal color format is always aRGB (8888). The color formats from the inputs are converted to the internal format on read and converted back on write.
Figure 43.1 shows examples of objects that can be drawn in hardware with the 2D Drawing Engine, Figure 43.2 shows a simplified rendering pipeline setup, and Figure 43.3 shows the block diagram of the module.

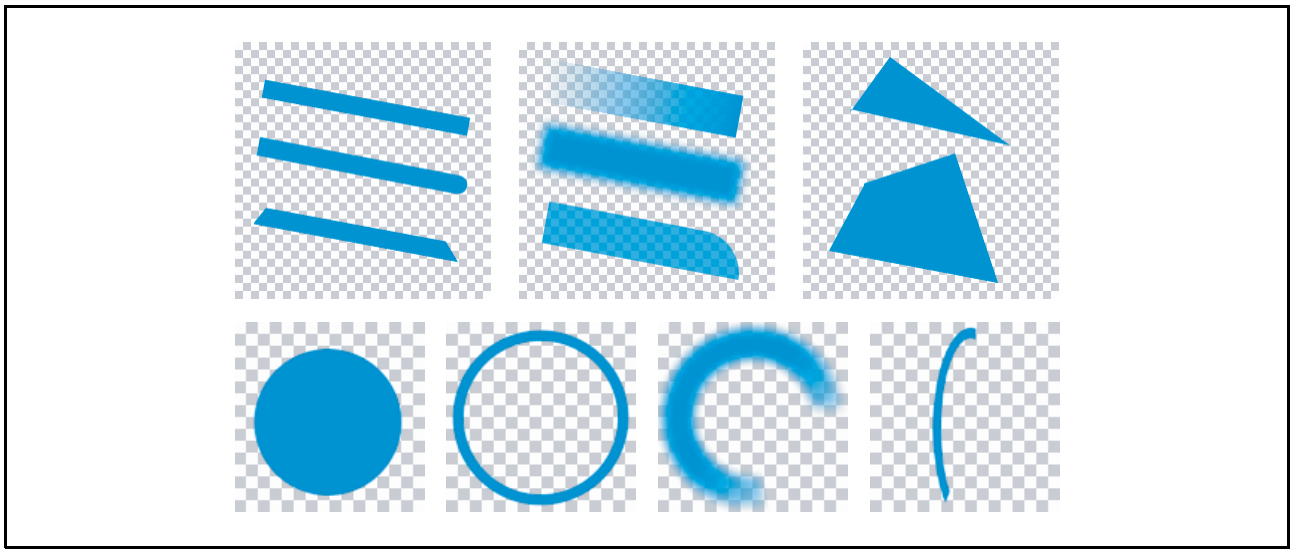


Figure 43.1 Examples of drawing objects

The 2D Drawing Engine also supports a display list mode, which efficiently decouples the CPU and graphics controller to enable rendering in parallel with other CPU activities.

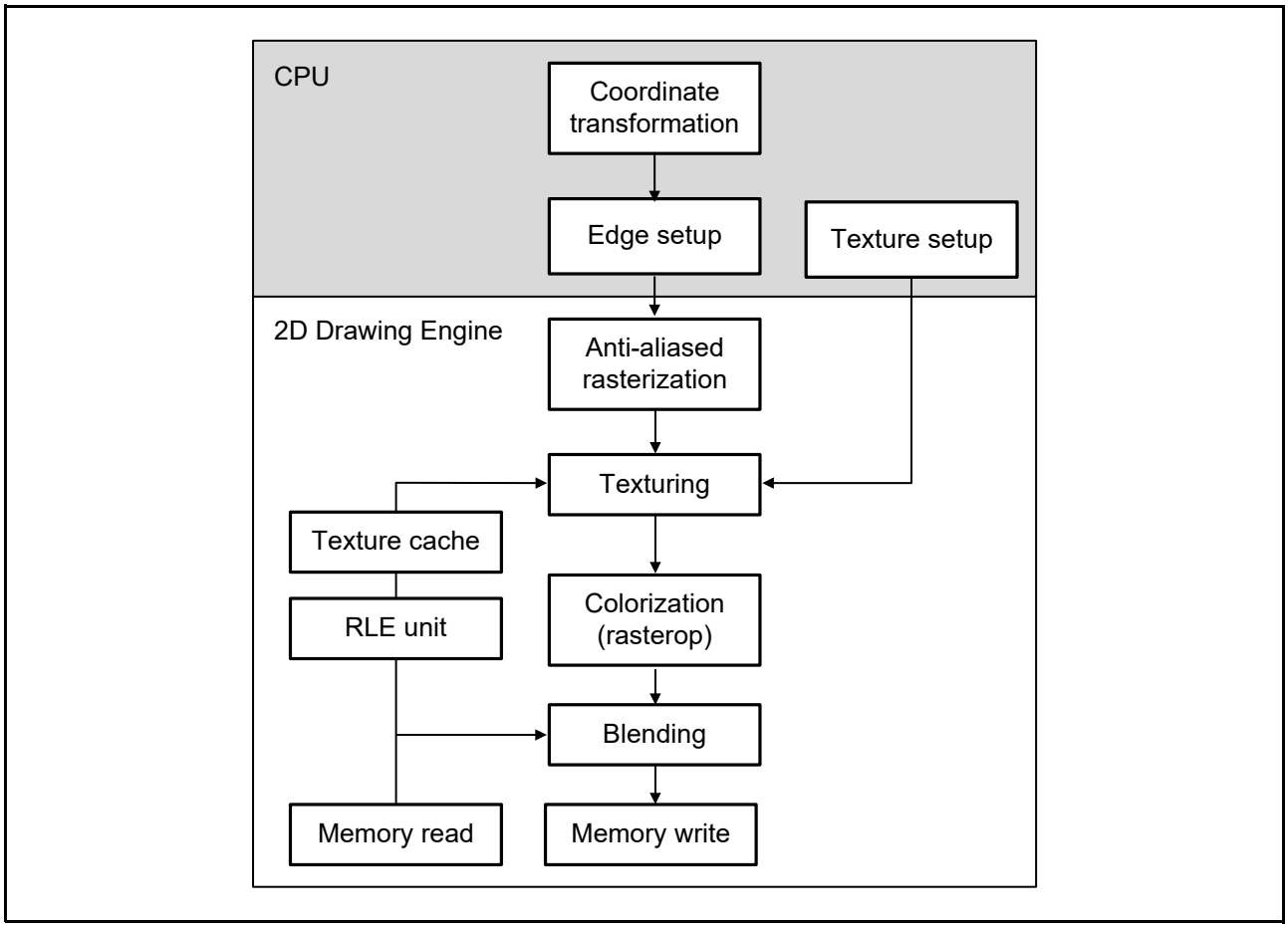


Figure 43.2 Simplified rendering pipeline setup

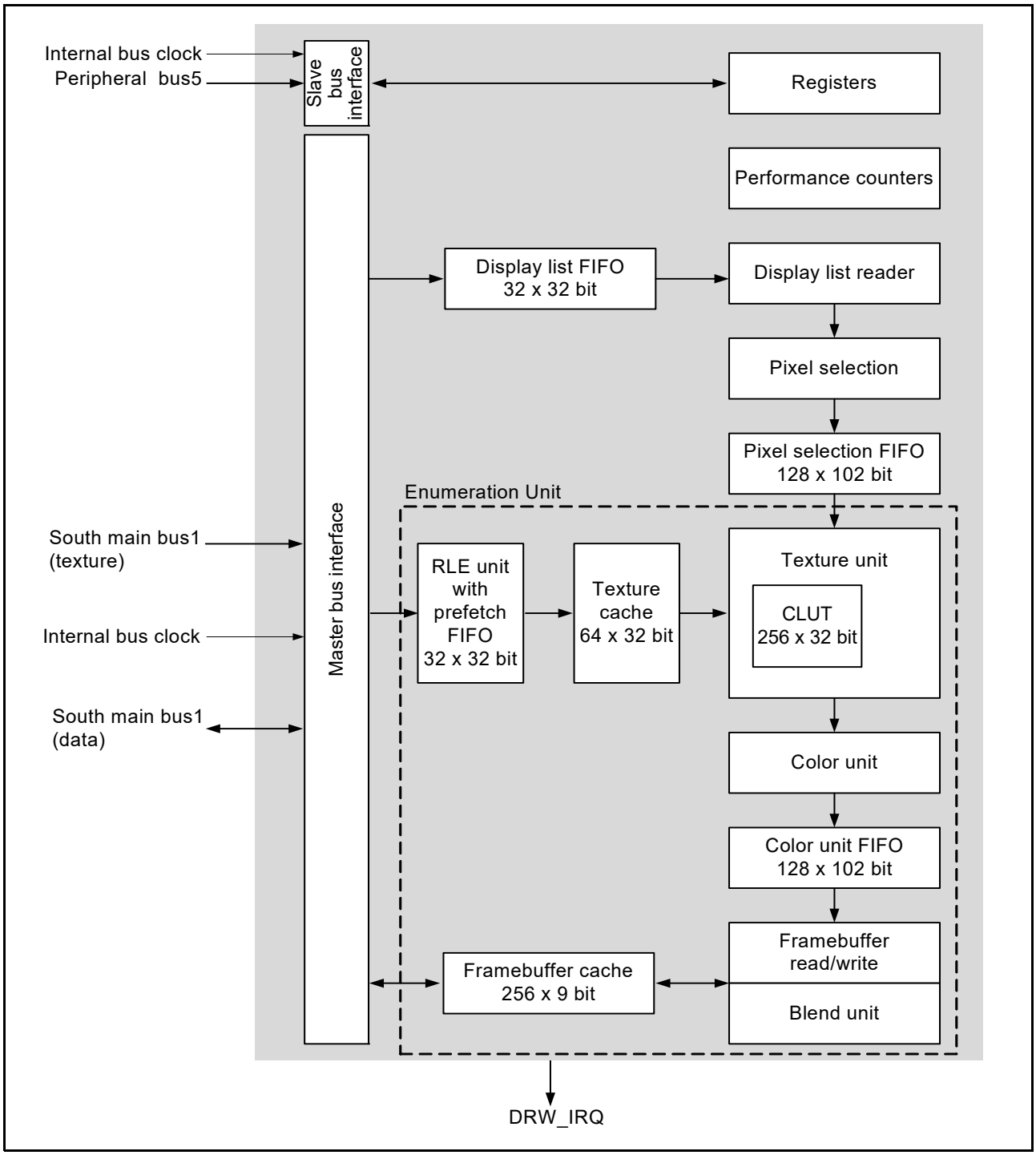


Figure 43.3 Block diagram of the 2D Drawing Engine

The 2D Drawing Engine accesses the South main bus1 as bus master through separate caches for:

- Reading and writing pixel data from and to the framebuffer
- Reading textures
- Reading display lists.

The control registers are accessed through the internal peripheral bus interface.

43.2 Register Descriptions

Symbols used in Register Descriptions:

Initial value: Register value after a power-on reset.
R/W: Readable/writable. The written value can be read.
R: Read-only.
W: Write-only.

Register Name	Abbreviation	R/W	Address	Initial value	Access Size
Geometry control register*	CONTROL	W	E820 A000h	0000 0000h	32
Status control register*	STATUS	R	E820 A000h	0000 0000h	32
Surface control register*	CONTROL2	W	E820 A004h	0000 0000h	32
Hardware version and feature set ID register*	HWREVISION	R	E820 A004h	0FBE 0107h	32
Limiter 1 start value register	L1START	W	E820 A010h	0000 0000h	32
Limiter 2 start value register	L2START	W	E820 A014h	0000 0000h	32
Limiter 3 start value register	L3START	W	E820 A018h	0000 0000h	32
Limiter 4 start value register	L4START	W	E820 A01Ch	0000 0000h	32
Limiter 5 start value register	L5START	W	E820 A020h	0000 0000h	32
Limiter 6 start value register	L6START	W	E820 A024h	0000 0000h	32
Limiter 1 x-axis increment register	L1XADD	W	E820 A028h	0000 0000h	32
Limiter 2 x-axis increment register	L2XADD	W	E820 A02Ch	0000 0000h	32
Limiter 3 x-axis increment register	L3XADD	W	E820 A030h	0000 0000h	32
Limiter 4 x-axis increment register	L4XADD	W	E820 A034h	0000 0000h	32
Limiter 5 x-axis increment register	L5XADD	W	E820 A038h	0000 0000h	32
Limiter 6 x-axis increment register	L6XADD	W	E820 A03Ch	0000 0000h	32
Limiter 1 y-axis increment register	L1YADD	W	E820 A040h	0000 0000h	32
Limiter 2 y-axis increment register	L2YADD	W	E820 A044h	0000 0000h	32
Limiter 3 y-axis increment register	L3YADD	W	E820 A048h	0000 0000h	32
Limiter 4 y-axis increment register	L4YADD	W	E820 A04Ch	0000 0000h	32
Limiter 5 y-axis increment register	L5YADD	W	E820 A050h	0000 0000h	32
Limiter 6 y-axis increment register	L6YADD	W	E820 A054h	0000 0000h	32
Limiter 1 bandwidth parameter register	L1BAND	W	E820 A058h	0000 0000h	32
Limiter 2 bandwidth parameter register	L2BAND	W	E820 A05Ch	0000 0000h	32
Base color register	COLOR1	W	E820 A064h	0000 0000h	32
Secondary color register	COLOR2	W	E820 A068h	0000 0000h	32
Pattern register	PATTERN	W	E820 A074h	0000 0000h	32
Bounding box dimension register	SIZE	W	E820 A078h	0000 0000h	32
Framebuffer pitch and spanstore delay register	PITCH	W	E820 A07Ch	0000 0000h	32
Framebuffer base address register	ORIGIN	W	E820 A080h	0000 0000h	32
U limiter start value register	LUSTART	W	E820 A090h	0000 0000h	32
U limiter x-axis increment register	LUXADD	W	E820 A094h	0000 0000h	32
U limiter y-axis increment register	LUYADD	W	E820 A098h	0000 0000h	32
V limiter start value integer part register	LVSTARTI	W	E820 A09Ch	0000 0000h	32
V limiter start value fractional part register	LVSTARTF	W	E820 A0A0h	0000 0000h	32
V limiter x-axis increment integer part register	LVXADDI	W	E820 A0A4h	0000 0000h	32
V limiter y-axis increment integer part register	LVYADDI	W	E820 A0A8h	0000 0000h	32
V limiter increment fractional parts register	LVYXADDF	W	E820 A0ACh	0000 0000h	32

Register Name	Abbreviation	R/W	Address	Initial value	Access Size
Texels per texture line register	TEXPITCH	W	E820 A0B4h	0000 0000h	32
Texture size or texture address mask register	TEXMASK	W	E820 A0B8h	0000 0000h	32
Texture base address register	TEXORIGIN	W	E820 A0BCh	0000 0000h	32
Interrupt control register	IRQCTL	W	E820 A0C0h	0000 0000h	32
Cache control register	CACHECTL	W	E820 A0C4h	0000 0000h	32
Display list start address register	DLISTSTART	W	E820 A0C8h	0000 0000h	32
Performance counter 1	PERFCOUNT1	R/W	E820 A0CCh	0000 0000h	32
Performance counter 2	PERFCOUNT2	R/W	E820 A0D0h	0000 0000h	32
Performance counters control register	PERFTRIGGER	W	E820 A0D4h	0000 0000h	32
CLUT start address register	TEXCLADDR	W	E820 A0DCh	0000 0000h	32
CLUT data register	TEXCLDATA	W	E820 A0E0h	0000 0000h	32
CLUT offset register	TEXCLOFFSET	W	E820 A0E4h	0000 0000h	32
Color key register	COLKEY	W	E820 A0E8h	0000 0000h	32

Note: A write-only register and a read-only register are allocated to a single address.

43.2.1 Geometry Control Register (CONTROL)

Address(es): DRW.CONTROL E820 A000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	SPAN STORE	SPAN ABORT	UNION CD	UNION AB	UNION 56	UNION 34	UNION 12	BAND2 ENABLE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BAND1 ENABLE	LIM6THRE SHOLD	LIM5THRE SHOLD	LIM4THRE SHOLD	LIM3THRE SHOLD	LIM2THRE SHOLD	LIM1THRE SHOLD	QUAD3 ENABLE	QUAD2 ENABLE	QUAD1 ENABLE	LIM6 ENABLE	LIM5 ENABLE	LIM4 ENABLE	LIM3 ENABLE	LIM2 ENABLE	LIM1 ENABLE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	LIM1 ENABLE	Enable Limiter 1	0: Disable 1: Enable.	W
b1	LIM2 ENABLE	Enable Limiter 2	0: Disable 1: Enable.	W
b2	LIM3 ENABLE	Enable Limiter 3	0: Disable 1: Enable.	W
b3	LIM4 ENABLE	Enable Limiter 4	0: Disable 1: Enable.	W
b4	LIM5 ENABLE	Enable Limiter 5	0: Disable 1: Enable.	W
b5	LIM6 ENABLE	Enable Limiter 6	0: Disable 1: Enable.	W
b6	QUAD1 ENABLE	Enable Quadratic Coupling of Limiters 1 and 2	0: Disable 1: Enable.	W
b7	QUAD2 ENABLE	Enable Quadratic Coupling of Limiters 3 and 4	0: Disable 1: Enable.	W
b8	QUAD3 ENABLE	Enable Quadratic Coupling of Limiters 5 and 6	0: Disable 1: Enable.	W
b9	LIM1 THRESHOLD	Enable Limiter 1 Threshold Mode	0: Disable 1: Enable.	W
b10	LIM2 THRESHOLD	Enable Limiter 2 Threshold Mode	0: Disable 1: Enable.	W
b11	LIM3 THRESHOLD	Enable Limiter 3 Threshold Mode	0: Disable 1: Enable.	W
b12	LIM4 THRESHOLD	Enable Limiter 4 Threshold Mode	0: Disable 1: Enable.	W
b13	LIM5 THRESHOLD	Enable Limiter 5 Threshold Mode	0: Disable 1: Enable.	W
b14	LIM6 THRESHOLD	Enable Limiter 6 Threshold Mode	0: Disable 1: Enable.	W
b15	BAND1 ENABLE	Enable Band Post Process for Limiter 1	0: Disable 1: Enable. (See LnBAND.)	W
b16	BAND2 ENABLE	Enable Band Post Process for Limiter 2	0: Disable 1: Enable. (See LnBAND.)	W
b17	UNION12	Combine Limiter 1 and 2 as Union	0: Select minimum/intersect between limiters 1 and 2 1: Select maximum/union between limiters 1 and 2. (Output is called A.)	W
b18	UNION34	Combine Limiter 3 & 4 as Union	0: Select minimum/intersect between limiters 3 and 4 1: Select maximum/union between limiters 3 and 4. (Output is called B.)	W

Bit	Symbol	Bit name	Description	R/W
b19	UNION56	Combine Limiter 5 & 6 as Union	0: Select minimum/intersect between limiters 5 and 6 1: Select maximum/union between limiters 5 and 6. (Output is called D.)	W
b20	UNIONAB	Combine Outputs A & B as union	0: Select minimum/intersect between limiters A and B 1: Select maximum/union between limiters A and B. (Output is called C.)	W
b21	UNIONCD	Combine Outputs C & D as Union	0: Select minimum/intersect between limiters C and D 1: Select maximum/union between limiters C and D. (Output is final.)	W
b22	SPANABORT	Spanabort	0: Disable 1: Enable. Shape is horizontally convex, only a single span per scan line. See section 43.6.2.6 (2), Spanabort.	W
b23	SPANSTORE	Spanstore	0: Disable 1: Enable. Next line span start is always equal to or right of current-line span start. See section 43.6.2.6 (1), Spanstore.	W
b31 to b24	—	Reserved	The write value should be 0.	W

43.2.2 Surface Control Register (CONTROL2)

Address(es): DRW.CONTROL2 E820 A004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RLEPIXEL WIDTH[1:0]		BDIA	BSIA	CLUT FORMAT	COLKEY ENABLE	CLUT ENABLE	RLE ENABLE	WRITEALPHA[1:0]		WRITEFORMAT [1:0]		READFORMAT [1:0]		TEXTURE FILTERY	TEXTURE FILTERX
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TEXTURE CLAMPY	TEXTURE CLAMPX	BC2	BDI	BSI	BDF	BSF	WRITEFORMAT[2]	BDFA	BSFA	READ FORMAT[3:2]		USEACB	PATTERN SOURCE	TEXTURE ENABLE	PATTERN ENABLE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	PATTERN ENABLE	Pattern Color Enable for Pixel Source	Pixel source is a pattern color (blend of COLOR1 and COLOR2 depending on PATTERN and pattern index). 0: Disable pattern 1: Enable pattern. When patterns are used to fill a primitive an index into the pattern bit mask is generated for each pixel with the U limiter. Depending on the pattern bits the color is selected from COLOR1 and COLOR2 registers. Fractional indices can be interpolated between those two values by using TEXTUREFILTERX = 1. The pattern can be wrapped by using TEXTURECLAMPX = 0, and the mask must be set in the TEXMASK register using the mask for u.	W
b1	TEXTURE ENABLE	Texture Enable for Pixel Source	Pixel source is read from texture and used as an alpha to blend between COLOR1 and COLOR2. 0: Disable texture 1: Enable texture.	W
b2	PATTERN SOURCEL5	Limiter 5 Enable for Pattern Index	Limiter 5 is used as pattern index instead of the default U limiter. Limiter 5 can be combined with limiter 6 to form a quadratic limiter that can be used to make quadratic pattern functions to draw radial patterns.	W
b3	USEACB	Alpha Blend Mode	0: Use WRITEALPHA[1:0] mode 1: Use full alpha channel blending mode.	W
b5, b4	READ FORMAT[3:2]	Texture Format Descriptor	Bits [3] and [2] of the texture buffer format. See the detailed description of the READFORMAT[1:0] bit in this section.	W
b6	BSFA	Blend Source Factor for Alpha Channel	Valid in alpha channel blending mode (USEACB = 1). 0: Use 1.0 as blend source factor for alpha channel 1: Use alpha as blend source factor for alpha channel.	W
b7	BDFA	Blend Destination Factor for Alpha Channel	Valid in alpha channel blending mode (USEACB = 1). 0: Use 1.0 as blend destination factor for alpha channel 1: Use alpha as blend destination factor for alpha channel.	W
b8	WRITE FORMAT[2]	Writeback Framebuffer Format	Bit [2] of framebuffer pixel format. See the description of WRITEFORMAT[1:0] in this section.	W
b9	BSF	Blend Source Factor	Source factor is alpha (factor is 1 per default). 0: Use 1.0 as blend source factor 1: Use alpha as blend source factor.	W
b10	BDF	Blend Destination Factor	Destination factor is alpha (factor is 1 per default). 0: Use 1.0 as blend destination factor 1: Use alpha as blend destination factor.	W
b11	BSI	Blend Source Factor Inverted	Source factor is inverted (meaning 1-a or 1-1 depending on BSF). 0: Use blend factor as specified through BSF 1: Invert blend source factor (1-x).	W
b12	BDI	Blend Destination Factor Inverted	Destination factor is inverted (meaning 1-a or 1-1 depending on BDF). 0: Use blend factor as specified through BDF 1: Invert blend destination factor (1-x).	W

Bit	Symbol	Bit name	Description	R/W
b13	BC2	Blend color 2	Select of blend color 2 instead of framebuffer pixel. 0: Use pixel from framebuffer as destination (DST) 1: Use color 2 as destination (DST).	W
b14	TEXTURE CLAMPX	Calculating U Limiter Outside Used Texture	This bit describes what happens when the U limiter (x direction in texture space) calculates a u value outside of the used texture. 0: Texture wrap mode: Integer part of the calculated value from the U limiter is AND gated with TEXUMASK, resulting in a repetition of texture in the x/u direction 1: Texture clamp mode: Texture color at the border of the texture is taken, resulting in a repetition of texture border color in the x/u direction.	W
b15	TEXTURE CLAMPY	Calculating V Limiter Outside Used Texture	This bit describes what happens when the V limiter (y direction in texture space) calculates a v value outside of the used texture. 0: Texture wrap mode: Integer part of the calculated value from the V limiter is AND gated with TEXVMASK, resulting in a repetition of texture in the y/v direction. 1: Texture clamp mode: Texture color at the border of the texture is taken, resulting in a repetition of texture border color in the y/v direction.	W
b16	TEXTURE FILTERX	Linear Filtering on Texture U Axis	0: No filtering on texture u axis 1: Linear filtering on texture u axis.	W
b17	TEXTURE FILTERY	Linear Filtering on Texture V Axis	0: No filtering on texture V axis 1: Linear filtering on texture V axis.	W
b19, b18	READ FORMAT[1:0]	Texture Format Descriptor	Pixel format of the texture buffer. b5 b4 b19 b18 0 0 0 0: 8 bpp a (8) 0 0 0 1: 16 bpp RGB (565) 0 0 1 0: 32 bpp aRGB (8888) 0 0 1 1: 16 bpp aRGB (4444) 0 1 0 0: 16 bpp aRGB (1555) 0 1 0 1: 8 bpp aCLUT (44), 4 bit alpha and 4 bit indexed color 1 0 0 1: 8 bpp CLUT (8)/I (8), 8 bit indexed color/luminance 1 0 1 0: 4 bpp CLUT (4)/I (4), 4 bit indexed color/luminance 1 0 1 1: 2 bpp CLUT (2)/I (2), 2 bit indexed color/luminance 1 1 0 0: 1 bpp CLUT (1)/I (1), 1 bit indexed color/luminance. Other settings are prohibited.	W
b21, b20	WRITE FORMAT[1:0]	Writeback Framebuffer Format	Pixel format of the framebuffer. b8 b21 b20 0 0 0: 8 bpp a (8) 0 0 1: 16 bpp RGB (565) 0 1 0: 32 bpp aRGB (8888) 0 1 1: 16 bpp aRGB (4444). Other settings are prohibited.	W
b23, b22	WRITE ALPHA[1:0]	Writeback Alpha Source for Framebuffer	• In non-alpha channel blending mode (USEACB = 0): Sets the alpha source for the framebuffer. b23 b22 0 0: Use alpha from color 2 0 1: Use source alpha (pixel coverage) 1 0: Use 0.0 as alpha 1 1: Use alpha from framebuffer. • In alpha channel blending mode (USEACB = 1): Blends alpha in color 2 instead of framebuffer alpha. b23 b22 00: BC2A = 1: Use alpha in color 2 as destination (DST_A). else: BC2A = 0: Use alpha from framebuffer as destination (DST_A)	W
b24	RLEENABLE	RLE Enable	0: Disable RLE 1: Enable RLE.	W
b25	CLUTENABLE	CLUT Enable	0: Disable CLUT 1: Enable CLUT. If CLUTENABLE = 0 (CLUT disabled), the index is directly put on the RGB channels.	W
b26	COLKEY ENABLE	Color Keying Enable	0: Disable color keying 1: Enable color keying.	W

Bit	Symbol	Bit name	Description	R/W
b27	CLUTFORMAT	CLUT Format	0: Format CLUT as aRGB (8888) 1: Format CLUT as RGB (565).	W
b28	BSIA	Blend Source Factor Inverted in Alpha Channel	• In alpha channel blending mode (USEACB = 1): 0: Use blend factor as specified through BSFA 1: Invert blend source factor (1-x).	W
b29	BDIA	Blend Destination Factor Inverted in Alpha Channel	• In alpha channel blending mode (USEACB = 1): 0: Use blend factor as specified through BDFA 1: Invert destination blend factor (1-x).	W
b31, b30	RLEPIXEL WIDTH[1:0]	Texel Width for RLE Unit	b31 b30 0 0: 1 byte per texel 0 1: 2 bytes per texel 1 0: 3 bytes per texel 1 1: 4 bytes per texel.	W

43.2.3 Interrupt Control Register (IRQCTL)

Address(es): DRW.IRQCTL E820 A0C0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	BUSIRQCLR	BUSIRQEN	DLISTIRQCLR	ENUMIRQCLR	DLISTIRQEN	ENUMIRQEN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	ENUMIRQEN	ENUMIRQ Interrupt Mask Enable	0: Disable (mask) ENUMIRQ enumeration interrupt 1: Enable (unmask) ENUMIRQ enumeration interrupt.	W
b1	DLISTIRQEN	DLISTIRQ Interrupt Mask Enable	0: Disable (mask) DLISTIRQ display list interrupt 1: Enable (unmask) DLISTIRQ display list interrupt.	W
b2	ENUMIRQCLR	Clear ENUMIRQ	0: Do not clear ENUMIRQ enumeration interrupt 1: Clear ENUMIRQ enumeration interrupt.	W
b3	DLISTIRQCLR	Clear DLISTIRQ	0: Do not clear DLISTIRQ display list interrupt 1: Clear DLISTIRQ display list interrupt.	W
b4	BUSIRQEN	BUSIRQ Interrupt Mask Enable	0: Disable (mask) BUSIRQ bus error interrupt 1: Enable (unmask) BUSIRQ bus error interrupt.	W
b5	BUSIRQCLR	Clear BUSIRQ	0: Do not clear BUSIRQ bus error interrupt 1: Clear BUSIRQ bus error interrupt.	W
b31 to b6	—	Reserved	The write value should be 0.	W

43.2.4 Cache Control Register (CACHECTL)

Address(es): DRW.CACHECTL E820 A0C4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	CFLUSHTX	CENABLETX	CFLUSHFX	CENABLEFX
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	CENABLEFX	Framebuffer Cache Enable	0: Disable the framebuffer cache 1: Enable the framebuffer cache.	W
b1	CFLUSHFX	Flush Framebuffer Cache	0: Do not flush the framebuffer cache 1: Flush the framebuffer cache.	W
b2	CENABLETX	Texture Cache Enable	0: Disable the texture cache 1: Enable the texture cache.	W
b3	CFLUSHTX	Flush Texture Cache	0: Do not flush the texture cache 1: Flush the texture cache.	W
b31 to b4	—	Reserved	The write value should be 0.	W

43.2.5 Status Control Register (STATUS)

Address(es): DRW.STATUS E820 A000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	BUSERRMDL	BUSERRMTXMRL	BUSERRMFB	—	BUSIRQ	DLISTIRQ	ENUMIRQ	DLISTACTIVE	CACHEDIRTY	BUSYWRITE	BUSYENUM
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b0	BUSYENUM	Enumeration Unit Status	0: Enumeration unit idle 1: Enumeration unit busy, new primitive cannot be started.	R
b1	BUSYWRITE	Framebuffer Writeback Status	0: Framebuffer writeback finished 1: Framebuffer writeback busy, framebuffer type cannot be changed.	R
b2	CACHEDIRTY	Framebuffer Cache Status	0: Framebuffer cache is not dirty 1: Framebuffer cache is dirty, and frame should not be flipped.	R
b3	DLISTACTIVE	Display List Reader Status	0: Display list reader is idle 1: Display list reader busy, and no direct write access to registers allowed.	R
b4	ENUMIRQ	Enumeration Interrupt Triggered	0: Enumeration not finished or interrupt disabled 1: Enumeration finished interrupt triggered.	R
b5	DLISTIRQ	Display List Interrupt Triggered	0: Display list not finished or interrupt disabled 1: Display list finished interrupt triggered.	R
b6	BUSIRQ	Bus Error Interrupt Triggered	0: No bus error occurred or interrupt disabled 1: Bus error interrupt triggered.	R
b7	—	Reserved	This bit is read as 0.	R
b8	BUSERRMFB	Framebuffer Bus Error Interrupt Triggered	0: No framebuffer bus error occurred or interrupt disabled 1: Framebuffer bus error interrupt triggered.	R
b9	BUSERRMTXMRL	Texture Bus Error Interrupt Triggered	0: No texture bus error occurred or interrupt disabled 1: Texture bus error interrupt triggered*1.	R
b10	BUSERRMDL	Display List Bus Error Interrupt Triggered	0: No display list bus error occurred or interrupt disabled 1: Display list bus error interrupt triggered.	R
b31 to b11	—	Reserved	These bits are read as 0.	R

Note 1. Because the RLE unit is also reading data through the texture bus, an error during RLE data access is also reflected in this bit.

43.2.6 Hardware Version and Feature Set ID Register (HWREVISION)

Address(es): DRW.HWREVISION E820 A004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	ACBLEND	—	COLORKEY	TEXCLUT256	RLEUNIT	—	TEXCLU	PERFCOUNT	TXCACHE	FBCACHE	DLR	—
Value after reset:	0	0	0	0	1	1	1	1	1	0	1	1	1	1	1	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	REV[11:0]											
Value after reset:	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1

Bit	Symbol	Bit name	Description	R/W
b11 to b0	REV[11:0]	Revision Number	Revision number.	R
b16 to b12	—	Reserved	These bits are read as 0.	R
b17	DLR	Display List Reader Available	Display list reader is available.	R
b18	FBCACHE	Framebuffer Cache Available	Framebuffer cache is available.	R
b19	TXCACHE	Texture Cache Available	Texture cache is available.	R
b20	PERFCOUNT	Two Performance Counter Available	Two performance counter is available.	R
b21	TECLUT	Texture CLUT with 16 or 256 Entries Available	Texture CLUT with 16 or 256 entries is available.	R
b22	—	Reserved	This bit is read as 0.	R
b23	RLEUNIT	RLE Unit Available	RLE unit is available.	R
b24	TEXCLUT256	Texture CLUT Available	Texture CLUT is available.	R
b25	COLORKEY	Color Key Available	Color key is available.	R
b26	—	Reserved	This bit is read as 1.	R
b27	ACBLEND	Alpha Channel Blending Available	Alpha channel blending is available.	R
b31 to b28	—	Reserved	These bits are read as 0.	R

43.2.7 Base Color Register (COLOR1)

Address(es): DRW.COLOR1 E820 A064h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	COLOR1A[7:0]								COLOR1R[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	COLOR1G[7:0]								COLOR1B[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b7 to b0	COLOR1B[7:0]	Blue Channel of Color 1	Specifies blue channel of color 1.	W
b15 to b8	COLOR1G[7:0]	Green Channel of Color 1	Specifies green channel of color 1.	W
b23 to b16	COLOR1R[7:0]	Red Channel of Color 1	Specifies red channel of color 1.	W
b31 to b24	COLOR1A[7:0]	Alpha Channel of Color 1	Specifies alpha channel of color 1. 00h: Transparent ... FFh: Opaque.	W

43.2.8 Secondary Color Register (COLOR2)

Address(es): DRW.COLOR2 E820 A068h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	COLOR2A[7:0]								COLOR2R[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	COLOR2G[7:0]								COLOR2B[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b7 to b0	COLOR2B[7:0]	Blue Channel of Color 2	Specifies blue channel of color 2.	W
b15 to b8	COLOR2G[7:0]	Green Channel of Color 2	Specifies green channel of color 2.	W
b23 to b16	COLOR2R[7:0]	Red Channel of Color 2	Specifies red channel of color 2.	W
b31 to b24	COLOR2A[7:0]	Alpha Channel of Color 2	Specifies alpha channel of color 2. 00h: Transparent ... FFh: Opaque.	W

43.2.9 Pattern Register (PATTERN)

Address(es): DRW.PATTERN E820 A074h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	PATTERN[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b7 to b0	PATTERN[7:0]	Bitmap of the Pattern	Specifies bitmap of the pattern.	W
b31 to b8	—	Reserved	The write value should be 0.	W

43.2.10 Limiter N Start Value Register (LnSTART)

Address(es): DRW.L1START E820 A010h, DRW.L2START E820 A014h, DRW.L3START E820 A018h, DRW.L4START E820 A01Ch, DRW.L5START E820 A020h, DRW.L6START E820 A024h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LSTART[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LSTART[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	LSTART[31:0]	Start Value of the nth Limiter	Specifies start value of the nth limiter.	W

43.2.11 Limiter N X-Axis Increment Register (LnXADD)

Address(es): DRW.L1XADD E820 A028h, DRW.L2XADD E820 A02Ch, DRW.L3XADD E820 A030h, DRW.L4XADD E820 A034h, DRW.L5XADD E820 A038h, DRW.L6XADD E820 A03Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LXADD[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LXADD[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	LXADD[31:0]	X-Axis Increment	Specifies x-axis increment.	W

43.2.12 Limiter N Y-Axis Increment Register (LnYADD)

Address(es): DRW.L1YADD E820 A040h, DRW.L2YADD E820 A044h, DRW.L3YADD E820 A048h, DRW.L4YADD E820 A04Ch, DRW.L5YADD E820 A050h, DRW.L6YADD E820 A054h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LYADD[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LYADD[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	LYADD[31:0]	Y-Axis Increment	Specifies y-axis increment.	W

43.2.13 Limiter M Bandwidth Parameter Register (LmBAND)

Address(es): DRW.L1BAND E820 A058h, DRW.L2BAND E820 A05Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LBAND[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LBAND[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	LBAND[31:0]	Limiter m Bandwidth Parameter	Specifies limiter m bandwidth parameter.	W

43.2.14 Texture Base Address Register (TEXORIGIN)

Address(es): DRW.TEXORIGIN E820 A0BCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	TEXORIGIN[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TEXORIGIN[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	TEXORIGIN[31:0]	Texture Base Address	Specifies texture base address.	W

43.2.15 Texels Per Texture Line Register (TEXPITCH)

Address(es): DRW.TEXPITCH E820 A0B4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TEXPITCH[10:0]										
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b10 to b0	TEXPITCH[10:0]	Texels Per Texture Line	Specifies texels per texture line.	W
b31 to b11	—	Reserved	The write value should be 0.	W

43.2.16 Texture Size or Texture Address Mask Register (TEXMASK)

Address(es): DRW.TEXMASK E820 A0B8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	TEXVMASK[20:5]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TEXVMASK[4:0]					TEXUMASK[10:0]										
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b10 to b0	TEXUMASK[10:0]	U Mask in Texture Mode	Specifies the U mask. Set to texture_width-1. • In texture wrapping mode (CONTROL2.TEXTURECLAMPX = 0): Texture_width must be a power of 2 • In texture clamping mode (CONTROL2.TEXTURECLAMPX = 1): All widths up to 2048 are allowed.	W
b31 to b11	TEXVMASK[20:0]	V Mask in Texture Mode	Specifies the V mask. Set to TEXPITCH × (texture_height - 1) • In texture wrapping mode (CONTROL2.TEXTURECLAMPY = 0): Texture_height must be a power of 2 • In texture clamping mode (CONTROL2.TEXTURECLAMPY = 1): All heights up to 1024 are allowed.	W

43.2.17 U Limiter Start Value Register (LUSTART)

Address(es): DRW.LUSTART E820 A090h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LUSTART[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LUSTART[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	LUSTART[31:0]	U Limiter Start Value	Specifies U limiter start value.	W

43.2.18 U Limiter X-Axis Increment Register (LUXADD)

Address(es): DRW.LUXADD E820 A094h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LUXADD[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LUXADD[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	LUXADD[31:0]	U Limiter X-Axis Increment	Specifies U limiter x-axis increment.	W

43.2.19 U Limiter Y-Axis Increment Register (LUYADD)

Address(es): DRW.LUYADD E820 A098h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LUYADD[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LUYADD[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	LUYADD[31:0]	U Limiter Y-Axis Increment	Specifies U limiter y-axis increment.	W

43.2.20 V Limiter Start Value Integer Part Register (LVSTARTI)

Address(es): DRW.LVSTARTI E820 A09Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LVSTARTI[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LVSTARTI[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	LVSTARTI[31:0]	V Limiter Start Value Integer Part	Specifies integer part of V limiter start value.	W

43.2.21 V Limiter Start Value Fractional Part Register (LVSTARTF)

Address(es): DRW.LVSTARTF E820 A0A0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LVSTARTF[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b15 to b0	LVSTARTF[15:0]	V Limiter Start Value Fractional Part	Specifies fractional part of V limiter start value.	W
b31 to b16	—	Reserved	The write value should be 0.	W

43.2.22 V Limiter X-Axis Increment Integer Part Register (LVXADDI)

Address(es): DRW.LVXADDI E820 A0A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LVXADDI[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LVXADDI[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	LVXADDI[31:0]	V Limiter X-Axis Increment Integer Part	Specifies integer part of V limiter x-axis increment.	W

43.2.23 V Limiter Y-Axis Increment Integer Part Register (LVYADDI)

Address(es): DRW.LVYADDI E820 A0A8h

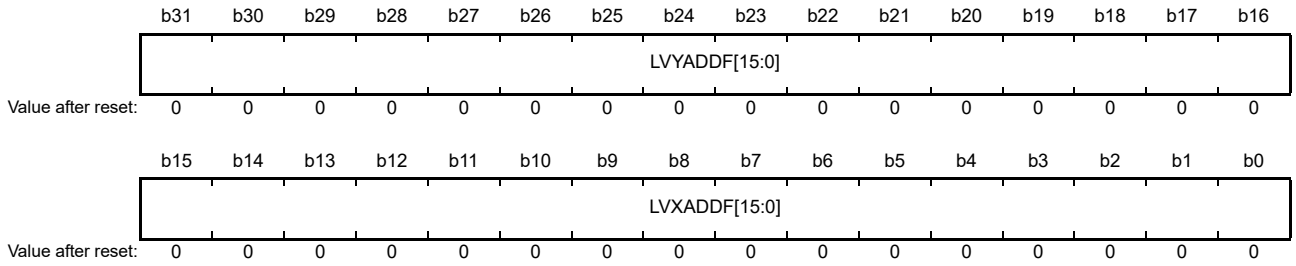
	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LVYADDI[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LVYADDI[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	LVYADDI[31:0]	V Limiter Y-Axis Increment Integer Part	Specifies integer part of V limiter y-axis increment.	W

43.2.24 V Limiter Increment Fractional Parts Register (LVYXADDF)

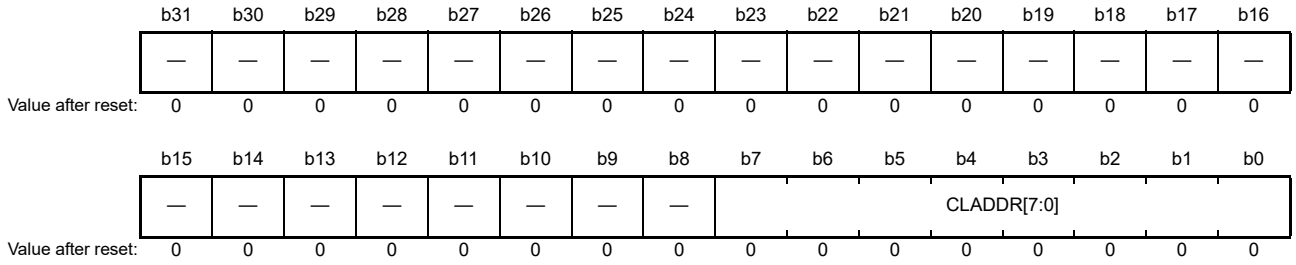
Address(es): DRW.LVYXADDF E820 A0ACh



Bit	Symbol	Bit name	Description	R/W
b15 to b0	LVXADDF[15:0]	V Limiter X-Axis Increment Fractional Part	Specifies fractional part of V limiter x-axis increment.	W
b31 to b16	LVYADDF[15:0]	V Limiter Y-Axis Increment Fractional Part	Specifies fractional part of V limiter y-axis increment.	W

43.2.25 CLUT Start Address Register (TEXCLADDR)

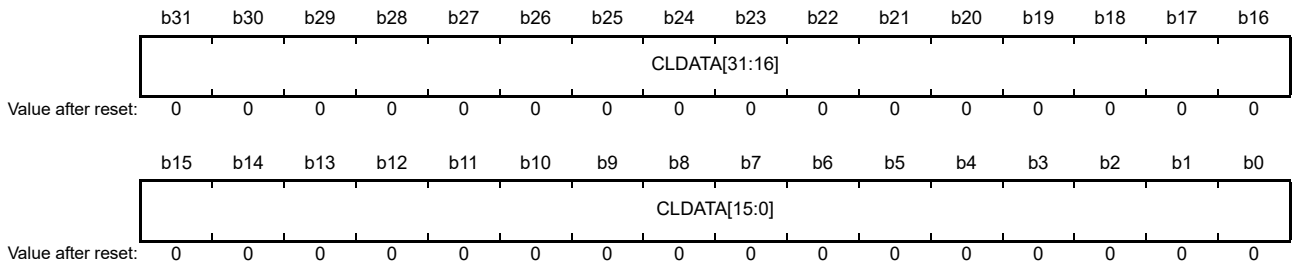
Address(es): DRW.TEXCLADDR E820 A0DCh



Bit	Symbol	Bit name	Description	R/W
b7 to b0	CLADDR[7:0]	Texture CLUT Start Address	Specifies texture CLUT start address.	W
b31 to b8	—	Reserved	The write value should be 0.	W

43.2.26 CLUT Data Register (TEXCLDATA)

Address(es): DRW.TEXCLDATA E820 A0E0h



Bit	Symbol	Bit name	Description	R/W
b31 to b0	CLDATA[31:0]	Texture CLUT Data	Specifies texture CLUT data.	W

43.2.27 CLUT Offset Register (TEXCLOFFSET)

Address(es): DRW.TEXCLOFFSET E820 A0E4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CLOFFSET[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b7 to b0	CLOFFSET[7:0]	Texture CLUT Offset	Specifies texture CLUT offset. CLOFFSET[7:0] is OR gated with the original index.	W
b31 to b8	—	Reserved	The write value should be 0.	W

43.2.28 Color Key Register (COLKEY)

Address(es): DRW.COLKEY E820 A0E8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	COLKEYR[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	COLKEYG[7:0]								COLKEYB[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b7 to b0	COLKEYB[7:0]	Blue Channel of Color Key	Specifies blue channel of color key.	W
b15 to b8	COLKEYG[7:0]	Green Channel of Color Key	Specifies green channel of color key.	W
b23 to b16	COLKEYR[7:0]	Red Channel of Color Key	Specifies red channel of color key.	W
b31 to b24	—	Reserved	The write value should be 0.	W

43.2.29 Bounding Box Dimension Register (SIZE)

Address(es): DRW.SIZE E820 A078h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SIZEY[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SIZEX[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b15 to b0	SIZEX[15:0]	Bounding Box Width	Specifies width of bounding box in pixels. Valid range: 0 to 1024.	W
b31 to b16	SIZEY[15:0]	Bounding Box Height	Specifies height of bounding box in pixels. Valid range: 0 to 1024.	W

43.2.30 Framebuffer Pitch And Spanstore Delay Register (PITCH)

Address(es): DRW.PITCH E820 A07Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SSD[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PITCH[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b15 to b0	PITCH[15:0]	Pitch of the Framebuffer	A negative width can be used to render bottom-up instead of top-down.	W
b31 to b16	SSD[15:0]	Spanstore Delay	Specifies number of scan lines to delay spanstore operations.	W

43.2.31 Framebuffer Base Address Register (ORIGIN)

Address(es): DRW.ORIGIN E820 A080h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ORIGIN[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ORIGIN[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	ORIGIN[31:0]	Address of the First Pixel in Framebuffer	Writing to ORIGIN triggers the start of rendering.	W

43.2.32 Display List Start Address Register (DLISTSTART)

Address(es): DRW.DLISTSTART E820 A0C8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DLISTSTART[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DLISTSTART[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	DLISTSTART[31:0]	Display List Start Address	Setting a new display list base address triggers execution of the new display list. Execution stops only when a new list is set or the current list terminates.	W

43.2.33 Performance Counters Control Register (PERFTRIGGER)

Address(es): DRW.PERFTRIGGER E820 A0D4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	PERFTRIGGER2[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	PERFTRIGGER1[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b4 to b0	PERFTRIGGER1[4:0]	Trigger of Performance Counter 1	Selects the internal event that increments the PERFCOUNT1 register. 0: Disable performance counter 1: Select 2D Drawing Engine active cycles 2: Select framebuffer read access 3: Select framebuffer write access 4: Select texture read access 5: Select invisible pixels (enumerated but selected with alpha 0%) 6: Select invisible pixels while internal FIFO is empty (lost cycles) 7: Select display list reader active cycles 8: Select framebuffer read hits 9: Select framebuffer read misses 10: Select framebuffer write hits 11: Select framebuffer write misses 12: Select texture read hits 13: Select texture read misses 31: Select every clock cycle (for use as timer).	W
b15 to b5	—	Reserved	The write value should be 0.	W
b20 to b16	PERFTRIGGER2[4:0]	Trigger of Performance Counter 2	Same as for PERFTRIGGER1, but for performance counter 2.	W
b31 to b21	—	Reserved	The write value should be 0.	W

43.2.34 Performance Counter k (PERFCOUNTk) (k = 1, 2)

Address(es): DRW.PERFCOUNT1 E820 A0CCh, DRW.PERFCOUNT2 E820 A0D0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PERFCOUNT[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PERFCOUNT[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Description	R/W
b31 to b0	PERFCOUNT[31:0]	Performance Counter k Value	Specifies counter k value. The counter is reset by writing PERFCOUNTk = 0000 0000h.	R/W

43.3 Drawing Features

43.3.1 Drawing Features Summary

43.3.1.1 Color Formats

Supported color formats are:

Framebuffer formats

- 8-bit: a (8)
- 16-bit: RGB (565), aRGB (4444)
- 32-bit: aRGB (8888).

Texture formats

- 1-bit: CLUT (1)/I (1)
- 2-bit: CLUT (2)/I (2)
- 4-bit: CLUT (4)/I (4)
- 8-bit: a (8), CLUT (8)/I (8), aCLUT (44)
- 16-bit: aRGB (4444), aRGB (1555), RGB (565)
- 24-bit: RGB (888) (run length encoded (RLE) unit)
- 32-bit: aRGB (8888).

CLUT formats use a 256-entry color lookup table.

43.3.1.2 BitBLT Features

The 2D Drawing Engine supports the BitBLT features using its vector drawing function to draw a rectangle and texture it based on the selected BitBLT function. This approach results in the following BitBLT features:

- Fill
- Copy
- Stretch BitBLT
- Rotate and scale
- Alpha blending
- Bilinear filtering
- Color conversion
- Subpixel exact placement.

43.3.1.3 Vector Drawing Features

The 2D Drawing Engine for vectors uses a half-plane rendering approach, which simplifies the implementation of the edge antialiasing and blurring features without much overhead.
When combining some of its functional units, this module can draw not only linear primitives such as lines or polygons but also quadratic equation-based primitives such as circles and ellipsoids.

The following primitives are supported:

- Lines
- Polygons
- Circles and ellipses
- Quadratic curves (software driver support)
- 2D texture mapping
- Bilinear filtering of the textures.

43.3.2 Vector Drawing

For a detailed explanation of the algorithms, see section 43.6, Rendering Pipeline. Supported vector drawing includes:

Lines

- Arbitrary width
- Round endpoints
- Truncated endpoints
- Alpha gradients
- Soft edges (blurring)
- Render attribute: color, pattern, or texture.

Polygons

- Triangles and quadrangles (complex polygons are tessellated by software)
- Alpha gradients
- Soft edges (blurring)
- Per edge controls for anti-aliasing
- Render attribute: color, pattern, or texture.

Circles and ellipses

- All conic sections
- Filled or with arbitrary width
- Arcs of 0° to 360°
- Soft edges
- Alpha gradients
- Render attribute: color, pattern, or texture.

Quadratic Bézier

- Approximated by circle arcs
- Arbitrary width
- Round or truncated endpoints
- Outlines, blurring
- Alpha gradients
- Render attribute: color.

Texture mapping

- 2D array of pixels that can be mapped implicitly or explicitly on all primitives provided by the 2D Drawing Engine
- Translation, rotation, and scaling/shearing
- Bilinear filtering of the textures
- 3D-like texturing accomplished with line-by-line mapping if either the x or y coordinate value is a constant.

43.3.3 BitBLT

A dedicated BitBLT unit is not required in the 2D Drawing Engine. The rendering pipeline described for vector drawing is used as the BitBLT unit and already provides a 1 pixel/cycle throughput. For details, see section 43.6, Rendering Pipeline.

43.3.3.1 Fill

Any rectangle in the framebuffer can be filled with any value. Possible color formats are any 8-, 16-, or 32-bpp format. The driver optimizes the fill to gain the full benefit of 32-bit parallel rasterization. If the color depth of the selected color format is less than 32 bpp, the driver corrects the alignment and fills 32 bits per clock cycle, resulting in two to four times faster fill performance in 8- and 16-bpp formats.

43.3.3.2 Copy

Any rectangle in the framebuffer can be filled with any rectangular data from the texture input. When the texture input points to the framebuffer, copying from framebuffer to framebuffer is possible. To avoid copy problems because of overlapping source and destination areas, the copy start point can be selected from top left to bottom right. Possible color formats are any 8-, 16-, or 32-bpp format.
The driver optimizes the copy to gain the full benefit of 32-bit parallel rasterization. If the color depth of the selected color format is less than 32 bpp, the driver corrects the alignment and copies 32 bits per clock cycle, resulting in two to four times faster copy performance in 8-and 16-bpp formats.

43.3.3.3 Stretch BitBLT

This is similar to the normal copy operation. Because the copy is done as a type of texture mapping, the full texture mapping feature set can be used. Any scaling ratios in the x and y directions is selectable, and filtering can be enabled independently for each axis.

43.3.3.4 Rotate and Scale

This is similar to the normal copy operation. Because the copy is done as a type of texture mapping, the full set of texture mapping features can be used. Any scaling ratio in the x and y directions and any rotation angle are selectable. The user can enable or disable filtering for the scalers independently for each of the x and y directions.

43.3.3.5 Alpha Blending

Alpha blending is a fundamental block in the rendering pipeline, so the full set of alpha blend features is available for any BitBLT operation. It is possible to copy an area and blend it over the destination by using any constant global alpha value (register value) or by using an alpha mask. The alpha mask is part of the texture data and can be either a per-pixel value together with a pixel color (aRGB formats) or an alpha-only format using a register color.
In addition to the color channels, the alpha channel can be blended. The formula for the alpha channel can be set independently from the formula for the color channels.

43.3.3.6 Bilinear Filtering

The texture unit can be used to scale, rotate, or shear images. The texturing result can be filtered in the x and y directions independently. When selecting both filters, the result is a bilinear filtered texture. Using the unit twice with two independent textures would generate trilinear filtered bitmaps, improving the visual impression for high dynamic scale ratios.

43.3.3.7 Color Conversion

Color conversion is required when using different texture formats than the framebuffer format. To save texture memory, formats having less bpp than that of the framebuffer are supported. The 2D Drawing Engine always operates internally with the 32-bpp aRGB (8888) format. All input data is converted into 32 bpp and is finally converted back into the framebuffer format.

43.4 Input and Output Data Formats

43.4.1 Source and Destination Data

There are two possible inputs, the framebuffer and the texture or pattern input. The output is always the framebuffer. Every drawing operation is internally rendered in 32 bpp aRGB (8888). If the input color does not provide an alpha channel, the blue channel is taken as the alpha channel. This alpha can be substituted with any alpha (for example, by an external constant) during the colorization step in the 2D Drawing Engine.

43.4.2 Framebuffer Color Formats

Table 43.1 shows the supported framebuffer color formats.

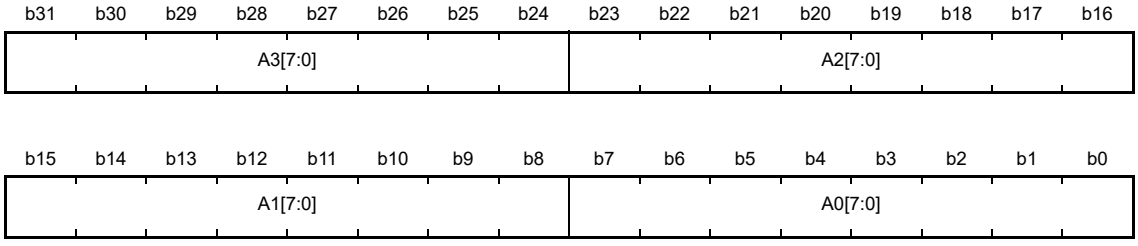
Table 43.1 Framebuffer color formats

Framebuffer memory occupation	Format	Remarks
8 bpp	a (8)	This color format uses 1 byte per pixel. The alpha channel is internally replicated on the red, blue, and green channels and can be substituted with any color during the color step in the 2D Drawing Engine.
16 bpp	RGB (565)	This color format uses 2 bytes per pixel with 5 bits for red and blue and 6 bits for green. The blue color is taken as the alpha channel during color conversion. The alpha can be substituted with any alpha during the colorization step in the 2D Drawing Engine.
	aRGB (4444)	This color format uses 2 bytes per pixel with 4 bits for each color and alpha channel.
32 bpp	aRGB (8888)	This color format uses 4 bytes per pixel with 8 bits for each color and alpha channel.

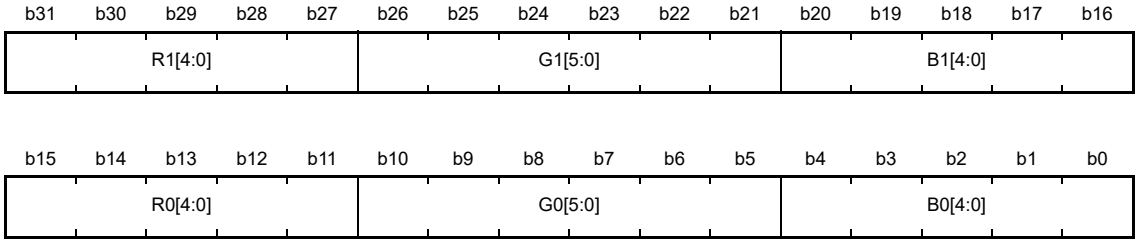
The framebuffer color format is selected in the Surface Control Register bits, CONTROL2.WRITEFORMAT[2:0].

The data allocation of the frame buffer format is as follows.

- a(8) (WRITEFORMAT[2:0]=0)

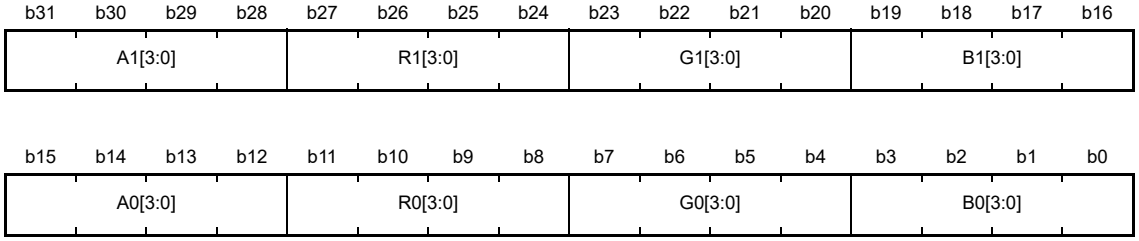


- RGB(565) (WRITEFORMAT[2:0]=1)



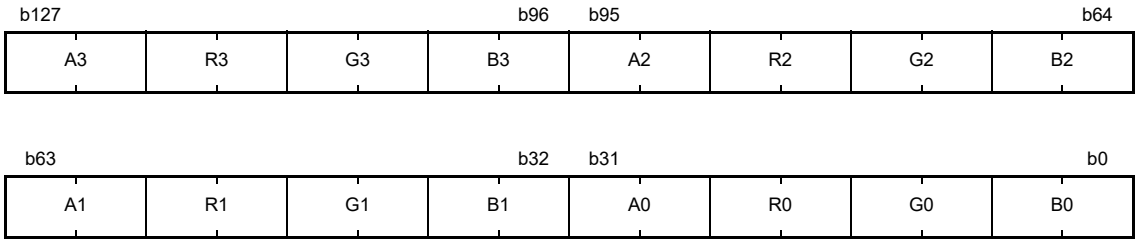
- aRGB(8888) (WRITEFORMAT[2:0]=2)

- aRGB(4444) (WRITEFORMAT[2:0]=3)



The data allocation on the 128-bit South main bus is little-endian as shown below.

- aRGB(8888)



Other formats are also little-endian.

43.4.3 Texture Color Formats

Table 43.2 shows the supported texture color formats.

Table 43.2 Texture color formats

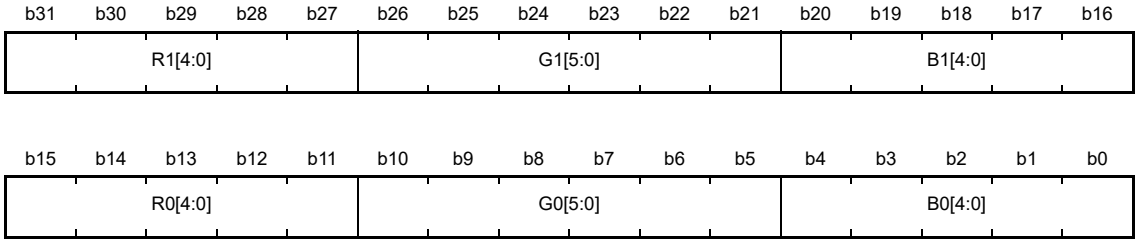
Texture memory occupation	Format	Remarks
1 bpp	CLUT (1)/I (1)	In this mode, a 1-bit index is used to address one of 256 predefined colors in the color lookup table. If the CLUT is not used, the index is taken as a luminance value.
2 bpp	CLUT (2)/I (2)	In this mode, a 2-bit index is used to address one of 256 predefined colors in the color lookup table. If the CLUT is not used, the index is taken as a luminance value.
4 bpp	CLUT (4)/I (4)	In this mode, a 4-bit index is used to address one of 256 predefined colors in the color lookup table. If the CLUT is not used, the index is taken as a luminance value.
8 bpp	CLUT (8)/I (8)	In this mode, an 8-bit index is used to address one of 256 predefined colors in the color lookup table. If the CLUT is not used, the index is taken as a luminance value.
	a (8)	This color format uses 1 byte per pixel. The alpha channel is internally replicated on the red, blue, and green channels and can be substituted with any color during the colorization step in the 2D Drawing Engine.
	aCLUT (44)	This color format uses 1 byte per pixel. 4 bits are used as an alpha value and 4 bits are used as an index to a color palette. This approach saves space if 16 colors are sufficient to describe the image, because the next bigger alpha format would be 2-byte aRGB (4444).
16 bpp	RGB(565)	This color format uses 2 bytes per pixel with 5 bits for red and blue and 6 bits for green. The blue color is taken as the alpha channel during color conversion. The alpha can be substituted with any alpha during the colorization step in the 2D Drawing Engine.
	aRGB (4444)	This color format uses 2 bytes per pixel with 4 bits for each color and alpha channel.
	aRGB (1555)	This color format uses 2 bytes per pixel. Every color channel has 5 bits and the topmost single bit is taken as an alpha value. This can be used to hold an image with a transparency mask.
24 bpp	RGB (888)	This color format uses 3 bytes per pixel with 8 bits for each color channel. This format is only available as run length encoded data (RLE compression).
32 bpp	aRGB (8888)	This color format uses 4 bytes per pixel with 8 bits for each color and alpha channel.

The texture color format is selected in the Surface Control Register bits, CONTROL2.READFORMAT[3:0] .

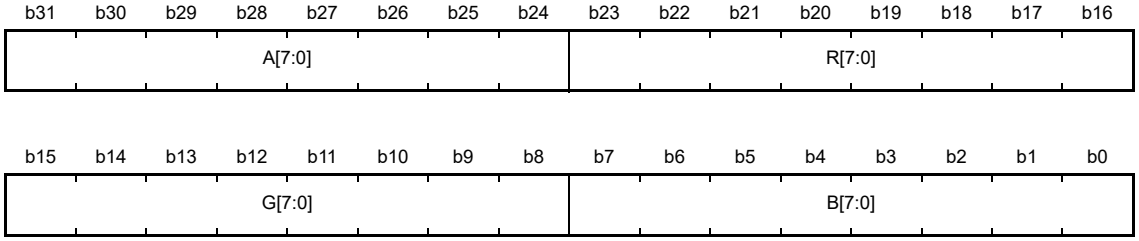
The data allocation of the texture format is as follows.

- a(8) (READFORMAT[3:0]=0h)

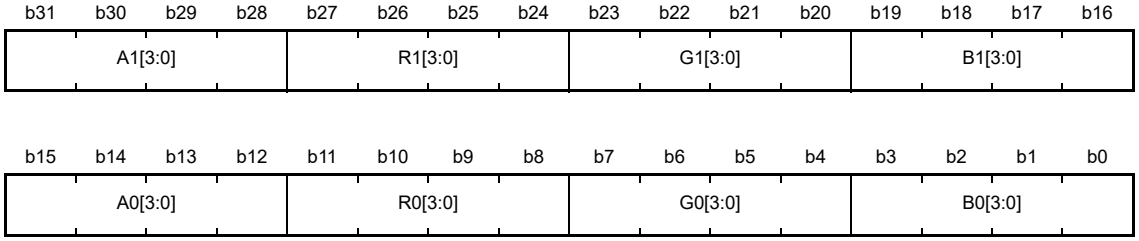
- RGB(565) (READFORMAT[3:0]=1h)



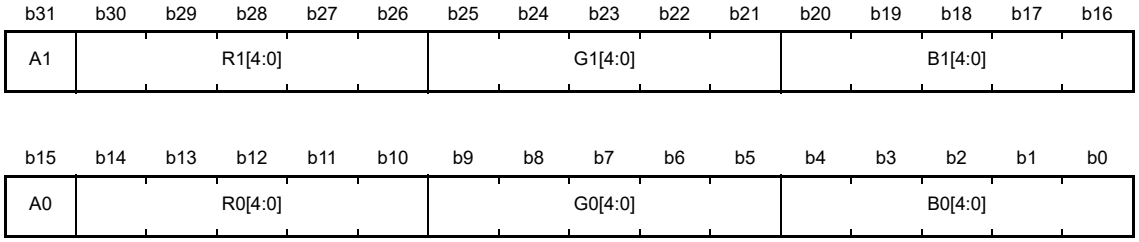
- aRGB(8888) (READFORMAT[3:0]=2h)



- aRGB(4444) (READFORMAT[3:0]=3h)

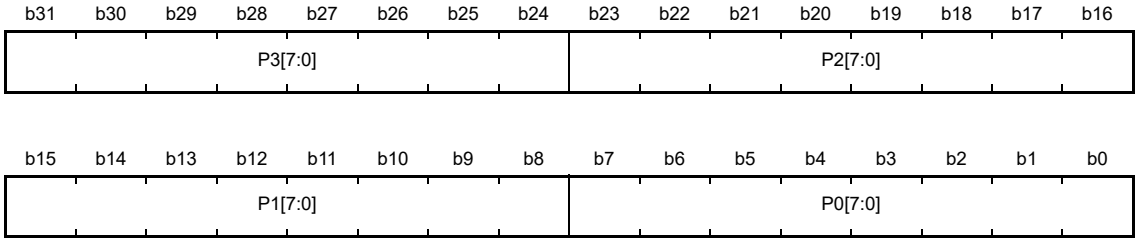


- aRGB(1555) (READFORMAT[3:0]=4h)

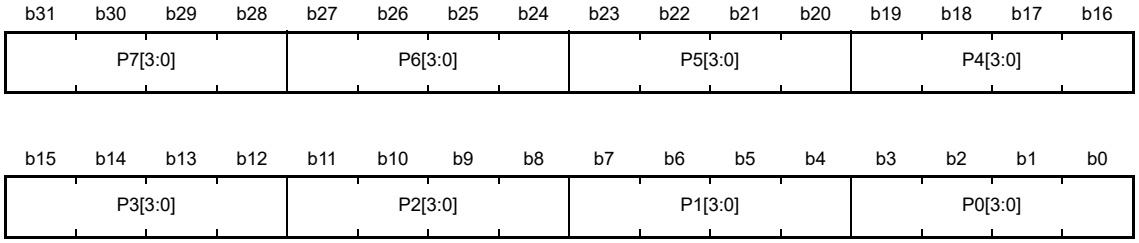


- aCLUT(44) (READFORMAT[3:0]=5h)

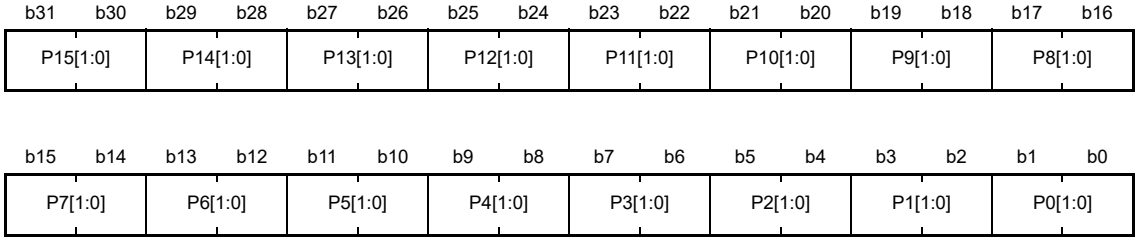
- CLUT(8) (READFORMAT[3:0]=9h)



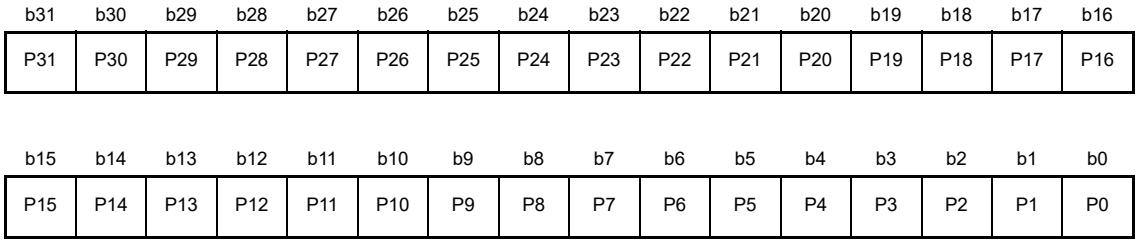
- CLUT(4) (READFORMAT[3:0]=Ah)



- CLUT(2) (READFORMAT[3:0]=Bh)



- CLUT(1) (READFORMAT[3:0]=Ch)



The data allocation on the 128-bit South main bus is the same little-endian as the frame buffer.

43.5 Texture Data Processing

Figure 43.4 shows the processing of texture data.

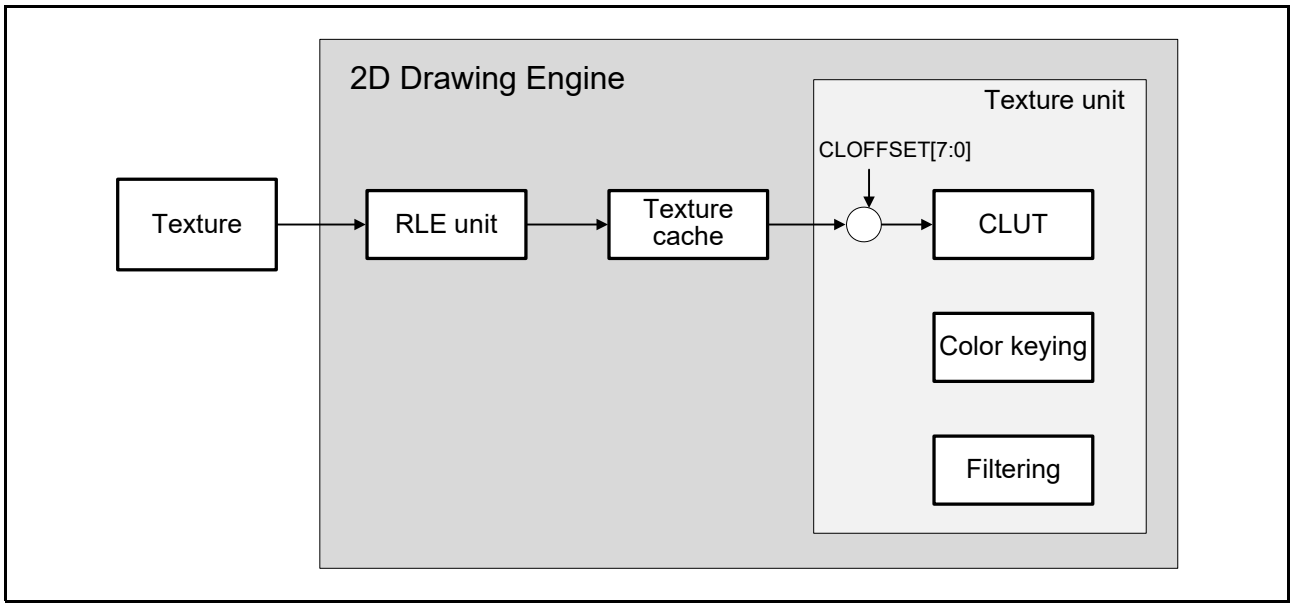


Figure 43.4 Texture data processing

43.5.1 Texture Color Format

Table 43.3 shows the supported texture data formats.

Table 43.3 Texture color formats

Texel bit width	Texel format
32 bits	aRGB (8888)
24 bits	RGB (888)
16 bits	aRGB (4444), aRGB (1555), RGB (565)
8 bits	CLUT (8)/I (8), a (8), aCLUT (44)
4 bits	CLUT (4)/I (4)
2 bits	CLUT (2)/I (2)
1 bit	CLUT (1)/I (1)

43.5.2 Run Length Encoded (RLE) Unit

The RLE unit decompresses Targa-like compressed textures and hands the decompressed texel data over to the texture unit. The key features are:

- Support for Targa format
- Avoids the additional Targa limitation to scan lines
- Support for clipping of compressed images, in which the 2D Drawing Engine is allowed to copy only a portion of a larger original texture
- Control of the RLE unit in display list operation
- Bypassing of the RLE unit logic if uncompressed textures are fetched.

Texture cache

- The RLE unit feeds the texture cache. The texture cache can be disabled by setting CACHECTL.CENABLETX = 0.

Note: A texture cache flush operation (CACHECTL.CFLUSHTX = 1) is necessary at the beginning and end of every new RLE texture.

The texture cache and the RLE unit can be bypassed by setting CONTROL2.RLEENABLE = 0.

43.5.2.1 RLE Texel Formats

Table 43.4 lists the data formats supported by the RLE unit.

Table 43.4 Texel formats supported by the RLE unit

Memory texel format	RLE parameters	RLE coded unit format (CONTROL2.RLEPIXELWIDTH[1:0])	Delivered format
32-bit aRGB (8888)	Included in Targa and RLE formats	32 bits (11b)	32 bits
24-bit RGB (888)		24 bits (10b)	32 bits *1
16-bit aRGB (4444), aRGB (1555), RGB (565)		16 bits (01b)	16 bits
8-bit CLUT (8)/I (8), a (8), aCLUT (44)		8 bits (00b)	8 bits
4-bit CLUT (4)/I (4)	Optional for RLE *2	8 = 2 x 4 bits	4 + 4 bits
2-bit CLUT (2)/I (2)	No RLE		
1-bit CLUT (1)/I (1)			

Note 1. 24-bit RGB (888) encoded texels are delivered as aRGB (8888) with Alpha set to 1.
Note 2. Encoding of textures with 4 bits per texel is not defined by the Targa specification but can be done by:
- Combining two 4-bit texels to one byte
- Padding with 4 zero bits at the end of the file, if the number of texels is odd
- Encoding as with 8-bit texels.

Texel addressing for RLE textures

The address of a texel is the byte address of the first byte of the texel. The origin of the texture is given by the register TEXORIGIN.

Note: The RLE code must begin at a word boundary of the memory.
When the FIFO is filled, there is no provision to inhibit read access beyond the end of the RLE code. To avoid memory access violations, the RLE code must be padded by 32 memory words, where every bit of each word is set to 1.

43.5.2.2 Targa RLE Format

Run-length encoded (RLE) images include two types of data elements:

- Run-length packets
- Raw packets.

The first field (1 byte) of each packet is called the "repetition count field". The second field is called the "pixel value field" (1, 2, 3, or 4 bytes per pixel).
For run-length packets, the pixel value field contains a single pixel value.
For raw packets, this field holds a variable number of pixel values.
The highest order bit of the repetition count indicates whether the packet is a raw packet or a run-length packet, as follows:

- If bit [7] of the repetition count is set to 1, the packet is a run-length packet
- If bit [7] of the repetition count is set to zero, the packet is a raw packet.

The lower 7 bits of the repetition count specify how many pixel values are represented by the packet. For a run-length packet, this count indicates how many successive pixels have the pixel value specified in the pixel value field. For raw packets, this count specifies how many pixel values are actually contained in the next field. This 7-bit value is actually encoded as 1 less than the number of pixels in the packet (a value of 0 implies 1 pixel while a value of 7Fh implies 128 pixels).

Run-length packet

Run-length packets are composed of two parts. The first is a repetition count and the second is the pixel value to repeat.

Table 43.5 Run-length packet

Field name	Repetition count field		Pixel value field
Contents	Packet type (must be 1 for run-length)	Pixel count (number of pixels encoded in this packet - 1)	Pixel data (the shared pixel value to be used)
Field size	1 bit	7 bits	Pixel depth (Table 43.4)

Raw packet

The raw packet always includes two fields. The first field is the repetition count and the second field is the pixel value field.

Table 43.6 Raw packet

Field name	Repetition count field		Pixel value field
Contents	Packet type (must be 0 for raw packet)	Pixel count (number of pixels encoded by this packet - 1)	Pixel data
Field size	1 bit	7 bits	Pixel depth × pixel count

43.5.3 Color Lookup Table (CLUT)

The color lookup table receives an index that addresses one out of the 256 predefined colors.
The predefined color format can be selected as:

- CONTROL2.CLUTFORMAT = 0: aRGB (8888)
- CONTROL2.CLUTFORMAT = 1: RGB (565).

The CLUT is filled by the use of two registers:

- TEXCLDATA
  The aRGB (8888) color definition is written to this register, while the CLUT address is taken from TEXCLADDR.
- TEXCLADDR.
  This is set to the first address of the CLUT to write to and is automatically incremented after each write to TEXCLDATA.

An offset for indexed formats (CLUT (1), CLUT (2), CLUT (4), and CLUT (8)) can be set up in the TEXCLOFFSET register to allow selecting an offset part of the CLUT. The CLUT index is calculated by CLUT (x) or TEXCLOFFSET.CLOFFSET[7:0].

43.5.3.1 CLUT/I Pixel Data Formats

The following tables explain in which order the pixels are stored within the byte. The left-most pixel is stored at the lowest bit of the memory byte.

CLUT (1)/I (1) format

The CLUT (1)/I (1) format expresses 1 pixel by using a total of 1 bit.

Memory byte	7 (MSB)	6	5	4	3	2	1	0 (LSB)
Pixel	P7	P6	P5	P4	P3	P2	P1	P0

The left-most pixel is stored at lowest bit of the memory byte.

CLUT (2)/I (2) format

The CLUT (2)/I (2) format expresses 1 pixel by using a total of 2 bits.

Memory byte	7 (MSB)	6	5	4	3	2	1	0 (LSB)
Pixel	P3		P2		P1		P0	

The left-most pixel is stored at lowest 2 bits of the memory byte.

CLUT (4)/I (4) format

The CLUT (4)/I (4) format expresses 1 pixel by using a total of 4 bits.

Memory byte	7 (MSB)	6	5	4	3	2	1	0 (LSB)
Pixel	P1				P0			

The left-most pixel is stored at lowest 4 bits of the memory byte.

43.5.4 Color Keying

The 2D Drawing Engine provides a color keying unit in front of the texture unit. It operates as follows:

1. When color keying is enabled (CONTROL2.COLKEYENABLE = 1), the incoming color is compared with the transparent color defined by COLKEY.
2. If the colors match, the alpha and color values are set to 0 to mark the incoming color as transparent, and the color is handled as if the alpha value were pre-multiplied.
3. If the value does not match, alpha is set to 1.
4. Additional operations such as alpha_in × alpha_const are still possible.

With this approach, an object such as a round icon can be cut out from a rectangular texture and can be faded by a constant alpha over the background.

43.6 Rendering Pipeline

43.6.1 Coordinate Transformation

Coordinate transformation such as rotation, translation, projection, and scaling must be done on the application side. This is not part of the 2D Drawing Engine hardware or driver. Because all coordinates fed into the 2D Drawing Engine are in fixed point format, these calculations can be made in fixed point format and do not require a floating point unit.

43.6.2 Rasterization

During rasterization, the vector data of the object must be converted to pixel data. To convert the data, the program sets up the edge interpolation hardware, called a limiter, for each edge of the object that calculates a decision value. The limiter determines which side of the edge the pixel is positioned on. The 2D Drawing Engine includes six internal hardware limiters. In principle, the limiter registers contain the distance between the pixel being processed and the edge. In the linear setup, a limiter describes a half plane. The intersection of all half planes is the object. If three half planes intersect, a triangle is created as shown in Figure 43.5.

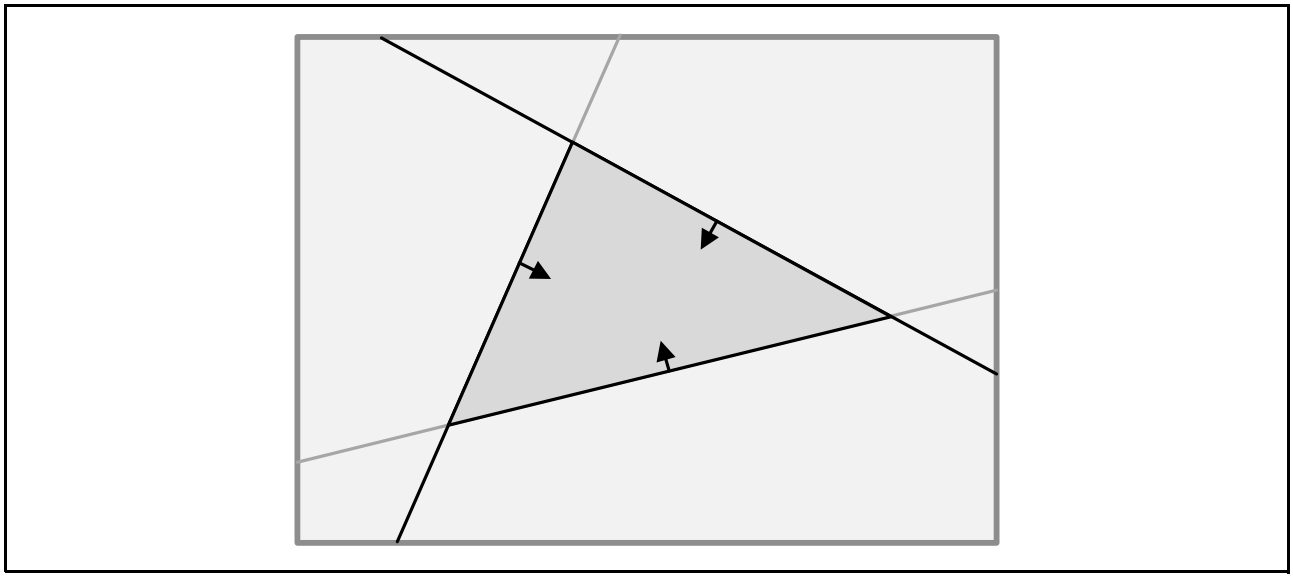


Figure 43.5 Intersection of half planes

The limiter output is clamped to an interval of [0:1]. In limiters 1 and 2 it is possible to apply a band filter before the clamping operation. In this case, the limiter is not describing a half plane but a small band. With this approach, a single limiter can describe a thick line of infinite length.
The output of the different limiters can be combined by the combiner units with a maximum or minimum operation. Maximum operation describes the union of both half planes, and minimum operation describes the intersection of both half planes. The final output is then used as an alpha value. Edge anti-aliasing can be done with no additional effort with this hardware.
To calculate the decision value for each possible pixel with a limiter, the bounding box of the object must be calculated. Then, the decision value for the top left corner of the bounding box must be calculated for each edge. Finally, the increments for a step in the x direction and a step in the y direction must be calculated. This is done by the CPU in the driver.
With this information, the 2D Drawing Engine scans the whole bounding box and calculates the decision value for every pixel incrementally. For a block diagram of the entire rasterization unit, see Figure 43.6.

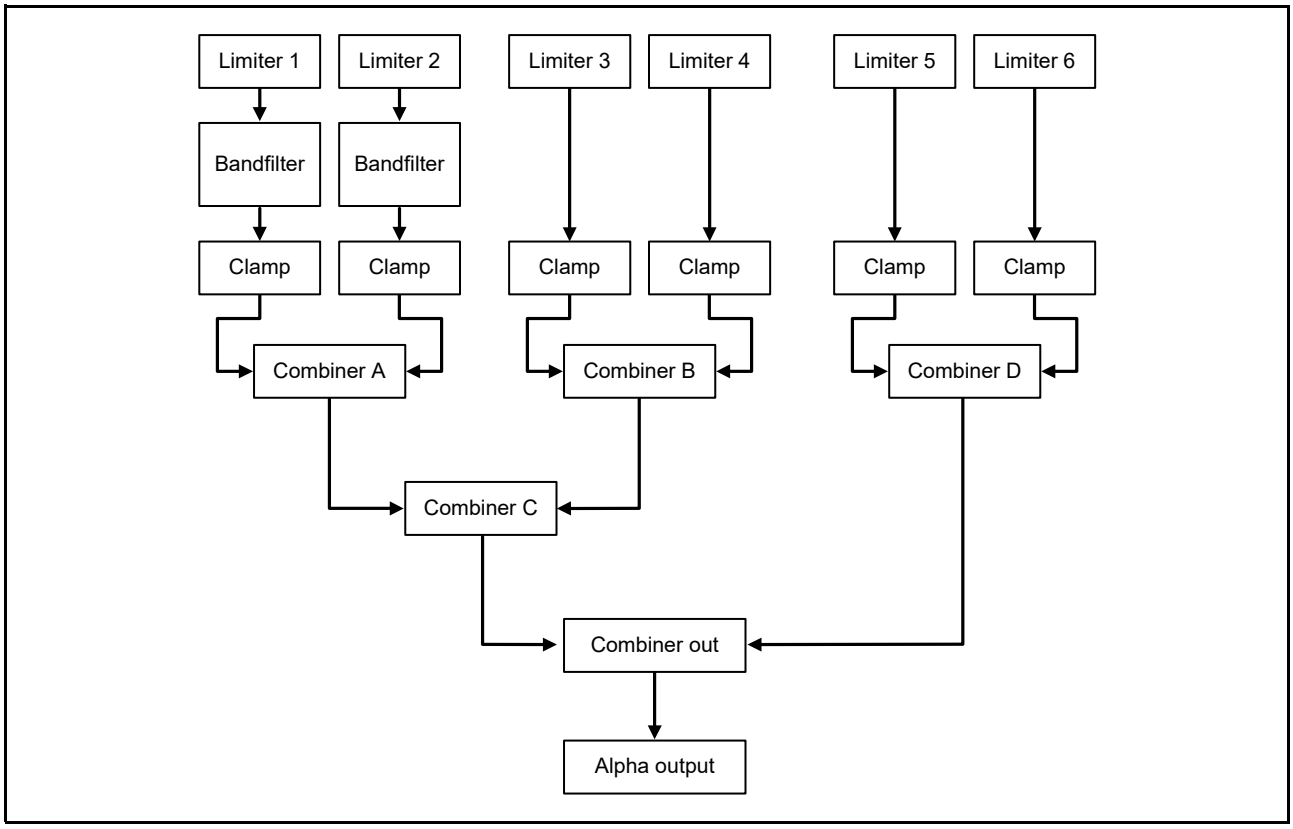


Figure 43.6 Block diagram of rasterization unit

The limiters calculate distances and the combiner units combine them to an alpha value. The combiner units define the conditions for whether or not a pixel is in the bounding box. The alpha value must be greater than 0.

Note: It is possible to have all limiters switched off.

43.6.2.1 Edge Setup Linear Case

(1) Mathematical background

To setup a linear edge, consider the line equation in the classical form:
This can be written as:

$$y = \tilde{a}x + \tilde{b}$$

This can be rewritten as:

$$0 = f(x, y) = \tilde{a}x - y + \tilde{b} = ax + by + c$$

with $a = \tilde{a}, b = -1, c = \tilde{b}$

This is a more general form. Consider a vector form of this equation:

$$\vec{p} = \begin{pmatrix} x \\ y \end{pmatrix}, \vec{n} = \begin{pmatrix} a \\ b \end{pmatrix} \Rightarrow f(x, y) = ax + by + c = \vec{p} \cdot \vec{n} + c$$

If a point $\vec{p}_0$ is on the line:

$$0 = f(x, y) = \vec{p}_0 \cdot \vec{n} + c \Rightarrow c = -\vec{p}_0 \cdot \vec{n}$$

Rewriting the constant, the equation becomes:

$$c = -\vec{p}_0 \cdot \vec{n} \Rightarrow f(x, y) = (\vec{p} - \vec{p}_0) \cdot \vec{n}$$

The vector $\vec{n}$ is called the normal vector and is perpendicular to the line. The setup can be seen in Figure 43.7.

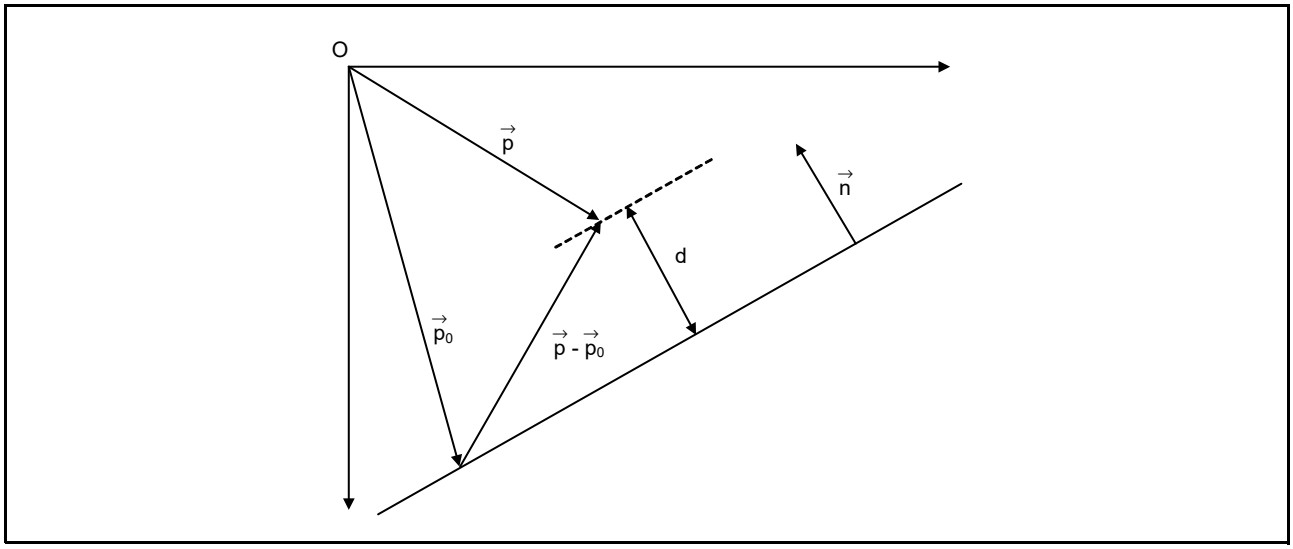


Figure 43.7 Distance of a point to a line

The projection of $\vec{p} - \vec{p}_0$ to $\vec{n}$ is the distance d of the point P to the line. In this case, f (x, y) describes the distance to the line of the pixel with coordinates (x, y).
To calculate the distance of every pixel of the bounding box incrementally, first the distance to the line at origin must be calculated:

$$f(0, 0) = -\vec{p}_0 \cdot \vec{n} = c$$

Next the increments for a step in the x direction and a step in the y direction must be calculated:

$$f(\vec{p} + \vec{e}_x) = (\vec{p} + \vec{e}_x - \vec{p}_0) \cdot \vec{n} = f(\vec{p}) + \vec{e}_x \cdot \vec{n} = f(\vec{p}) + a$$

$$f(\vec{p} + \vec{e}_y) = (\vec{p} + \vec{e}_y - \vec{p}_0) \cdot \vec{n} = f(\vec{p}) + \vec{e}_y \cdot \vec{n} = f(\vec{p}) + b$$

with $\vec{e}_x = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \vec{e}_y = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

Consequently, the new distance can be calculated from the old distance with the increments a and b. A step in the x direction changes the distance by a, and a step in the y direction changes the distance by b. The distance of the origin to

the line is c.
With this information, the entire bounding box can be scanned. If the bounding box top left corner is not at the origin, an offset must be added.

(2) Limiter operation

The 2D Drawing Engine contains six limiters. Each limiter contains three registers:

- LnSTART
- LnXADD
- LnYADD.

See Figure 43.8 for details. The limiters can be used in threshold mode, in which all values above 0.5 are set to 1 and all values below or equal to 0.5 are set to 0.
This feature is used when anti-aliasing is not wanted.

Note: In Figure 43.8, the following abbreviations are used:
start = LnSTART
xadd = LnXADD
yadd = LnYADD

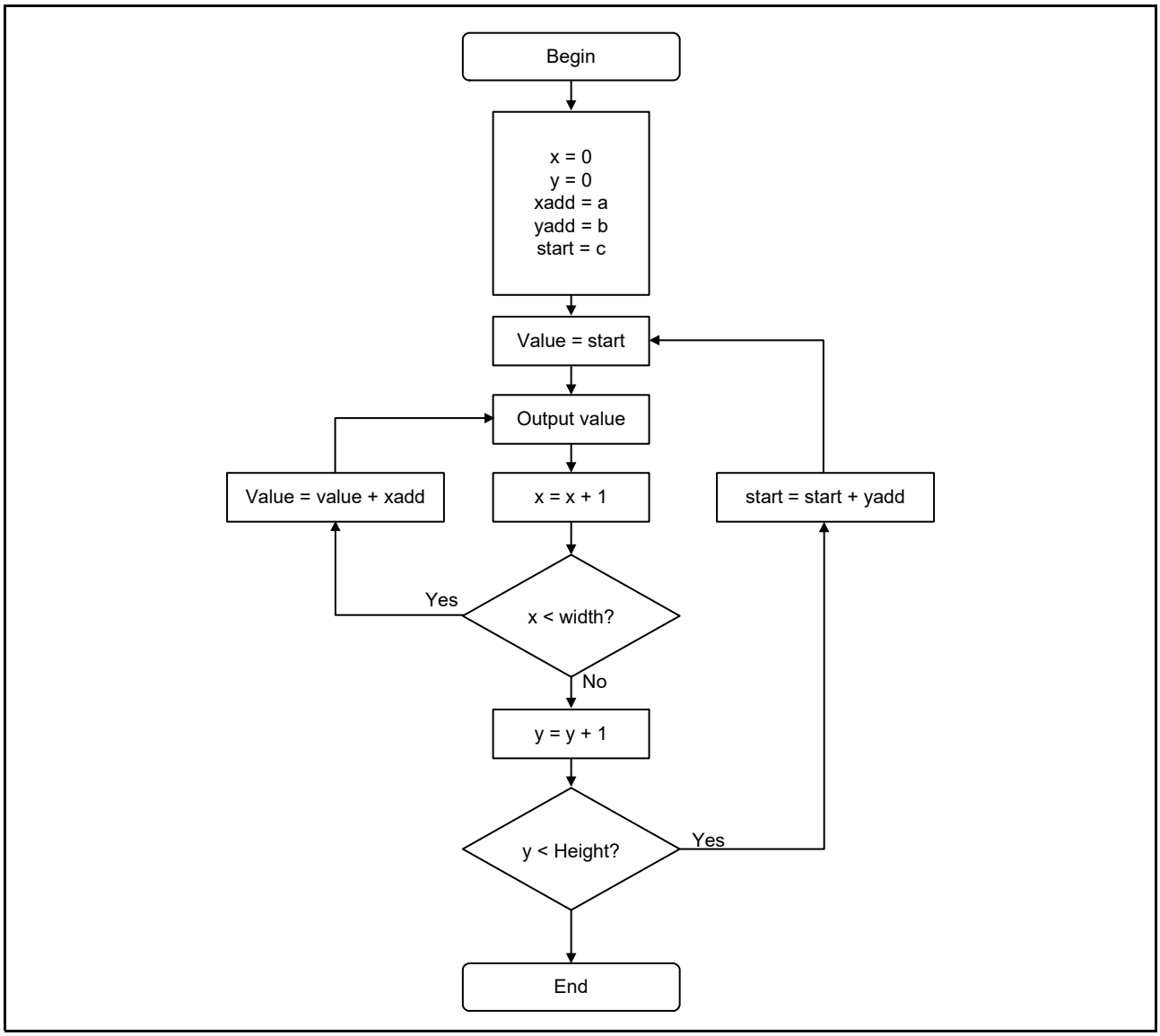


Figure 43.8 Operation flow of the linear limiter

(3) Example

If a straight line is given by the points P0 and P1, then the values are calculated as follows:

$$\Delta\vec{p} = \vec{p}_1 - \vec{p}_0 = \begin{pmatrix} x_1 - x_0 \\ y_1 - y_0 \end{pmatrix} = \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}$$

The normal vector (perpendicular but not unit size) is then:

$$\vec{n} = \begin{pmatrix} -\Delta y \\ \Delta x \end{pmatrix}$$

The not normalized distance between edge and origin is then:

$$\vec{p}_0 \cdot \vec{n} = -x_0\Delta y + y_0\Delta x$$

The limiter parameters would be:

$$\begin{aligned} start &= -x_0\Delta y + y_0\Delta x \\ xadd &= -\Delta y \\ yadd &= \Delta x \end{aligned}$$

In the normalized case, the normal vector is:

$$\vec{n} = \begin{pmatrix} -\Delta y \\ \Delta x \end{pmatrix} / (\sqrt{\Delta x^2 + \Delta y^2})$$

The distance between edge and origin is:

$$\vec{p}_0 \cdot \vec{n} = (-x_0 \cdot \Delta y + y_0 \cdot \Delta x) / (\sqrt{\Delta x^2 + \Delta y^2})$$

The limiter parameters would be:

$$start = (-x_0\Delta y + y_0\Delta x) / (\sqrt{\Delta x^2 + \Delta y^2})$$

$$xadd = -\Delta y / (\sqrt{\Delta x^2 + \Delta y^2})$$

$$yadd = \Delta x / (\sqrt{\Delta x^2 + \Delta y^2})$$

Normalization is only required if anti-aliasing is used. The driver contains an optimized inverse square root function to speed up the normalization process.

43.6.2.2 Edge Setup Quadratic Case

(1) Mathematical background

It is also possible to set up the limiters to incrementally calculate the following equation:

$$f(x, y) = ax^2 + by^2 + cx + dy + f$$

At the origin, the value is:

$$f(0, 0) = f$$

The step in the x direction is:

$$f(x + 1, y)$$
$$= a(x + 1)^2 + by^2 + c(x + 1) + dy + f$$
$$= ax^2 + 2ax + a + by^2 + cx + c + dy + f$$
$$= f(x, y) + 2ax + c + a$$
$$dx(x) = f(x + 1, y) - f(x, y) = 2ax + c + a$$

The step in the y direction is:

$$f(x, y + 1)$$
$$= ax^2 + b(y + 1)^2 + cx + d(y + 1) + f$$
$$= ax^2 + by^2 + 2by + b + cx + d(y + 1) + f$$
$$= f(x, y) + 2by + d + b$$
$$dy(y) = f(x, y + 1) - f(x, y) = 2by + d + b$$

In the quadratic case, the increments dx and dy depend on x and y and are not constant. They can be calculated incrementally:

$$d^2x = dx(x + 1) - dx(x) = 2a(x + 1) + c + 1 - (2ax + c + 1) = 2a$$
$$d^2y = dy(y + 1) - dy(y) = 2b(y + 1) + d + 1 - (2by + d + 1) = 2b$$

At the origin, the increments are:

$$dx(0) = c + 1 \text{ and } dy(0) = d + 1$$

By incrementing the value by dx and dy for every step in the x and y direction and incrementing dx, dy by d^2x, and d^2y for every step in the x and y direction, the quadratic equation can be easily calculated for the whole bounding box.

(2) Limiter operation

In the quadratic case, two linear limiters are combined to operate as one quadratic limiter, called limiter 1 and limiter 2. The registers are:

- L1START, L1XADD, L1YADD
- L2START, L2XADD, L2YADD.

See Figure 43.9 for details. The gray box is an addition that performs a different operation, as in the linear setup.
The limiters can be used in threshold mode, in which all values above 0.5 are set to 1 and all values below or equal to 0.5 are set to 0. This feature is used when anti-aliasing is not wanted.

Note: In Figure 43.9, the following abbreviations are used:

- start1 = L1START, start2 = L2START
- xadd1 = L1XADD, xadd2 = L2XADD
- yadd1 = L1YADD, yadd2 = L2YADD.

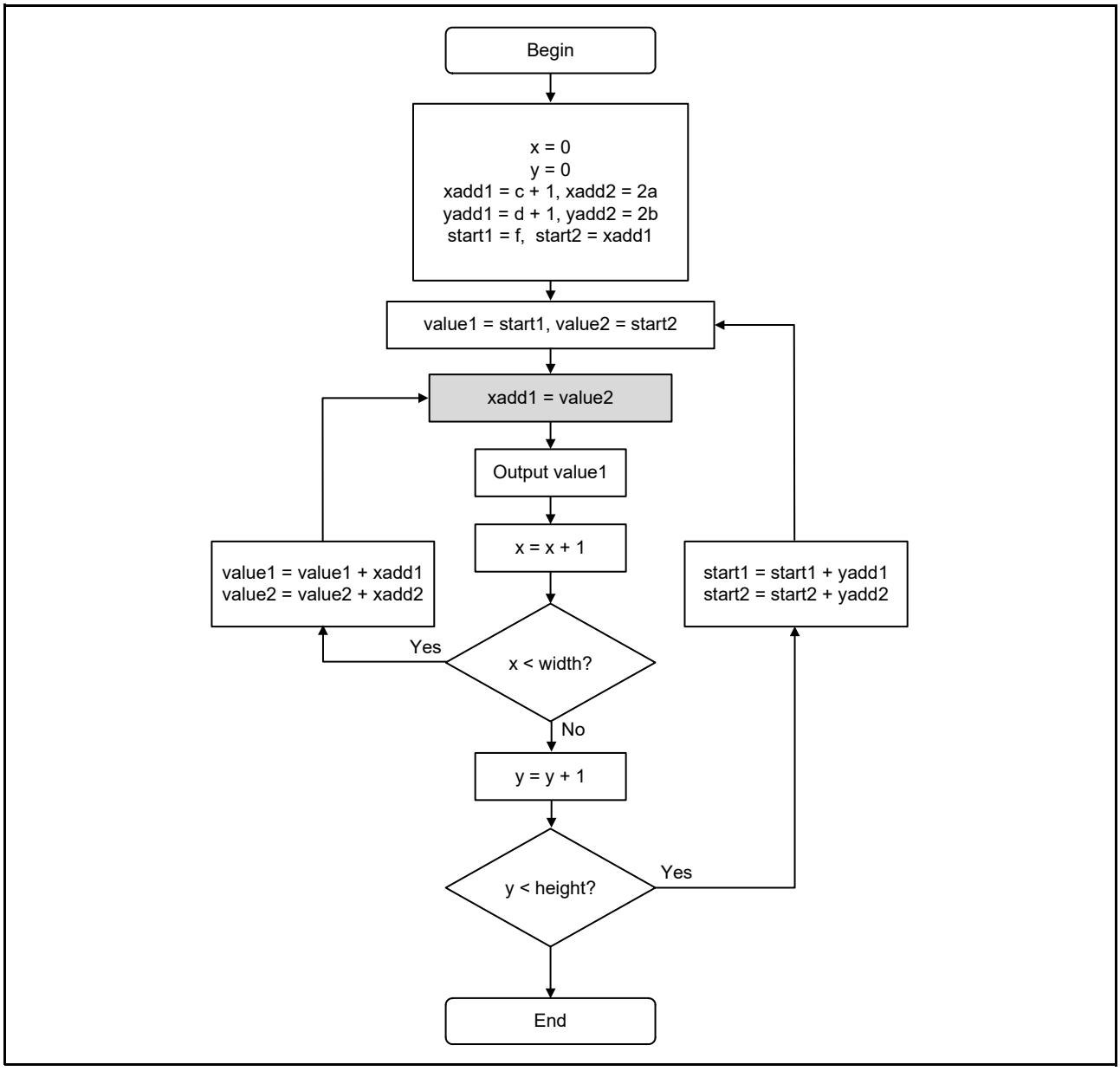


Figure 43.9 Operation flow of the quadratic limiter

(3) Example

Consider the equation for a circle with the center at $\vec{c} = \begin{pmatrix} s \\ t \end{pmatrix}$ and radius r:

$$0 = f(x, y) = (x - s)^2 + (y - t)^2 - r^2$$

This equation can be rewritten as:

$$f(x, y) = x^2 - 2xs + s^2 + y^2 - 2yt + t^2 - r^2$$

This can be sorted to fit to the original equation:

$$f(x, y) = x^2 + y^2 - 2sx - 2ty + (s^2 + t^2 - r^2)$$

With the following assignments, the circle equation can be calculated incrementally:

$$a = 1$$
$$b = 1$$
$$c = -2s$$
$$d = -2t$$
$$f = s^2 + t^2 - r^2$$

For the limiters with the results calculated in (1), Mathematical background, this would result in:

$$start1 = f = s^2 + t^2 - r^2$$
$$xadd1 = c + 1 = -2s + 1$$
$$yadd1 = d + 1 = -2t + 1$$
$$start2 = xadd1$$
$$xadd2 = 2a = 2$$
$$yadd2 = 2b = 2$$

43.6.2.3 Band Filter

The output of limiters 1 and 2 can be modified through a band filter. The band filter has a single filter parameter w.

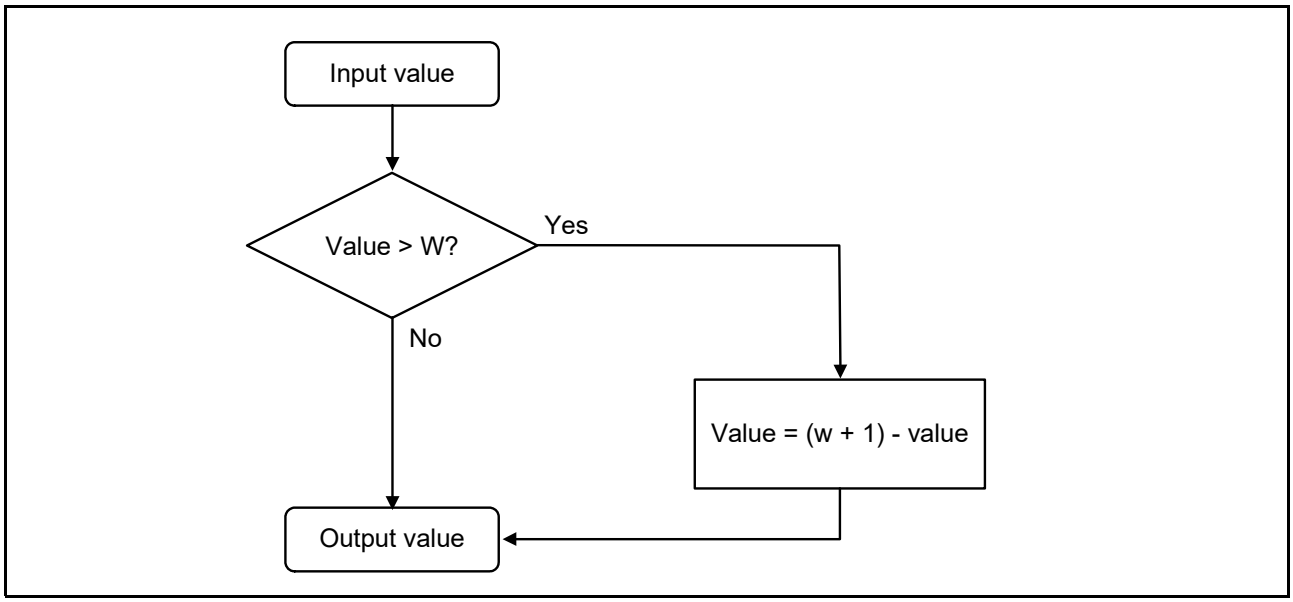


Figure 43.10 Band filter

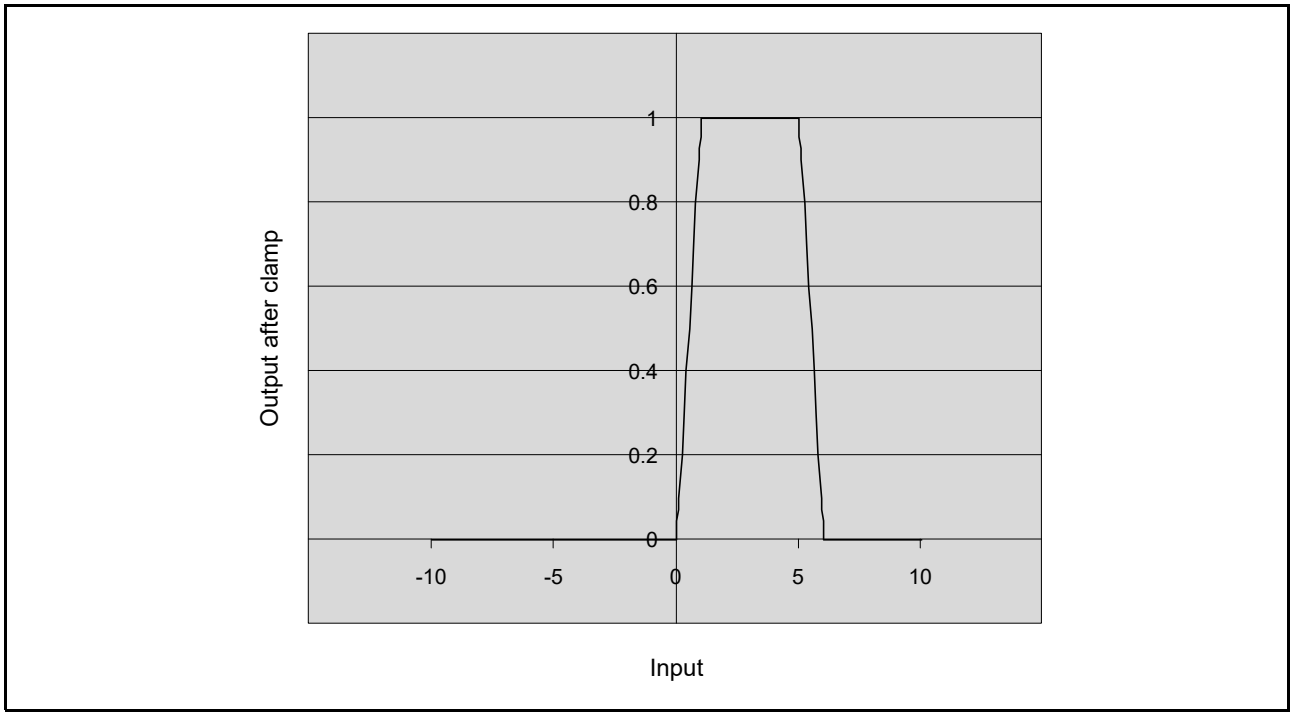


Figure 43.11 Band filter output after clamp with w = 7

43.6.2.4 Clamping Unit

The clamping unit cuts the limiter output to the interval [0:1].

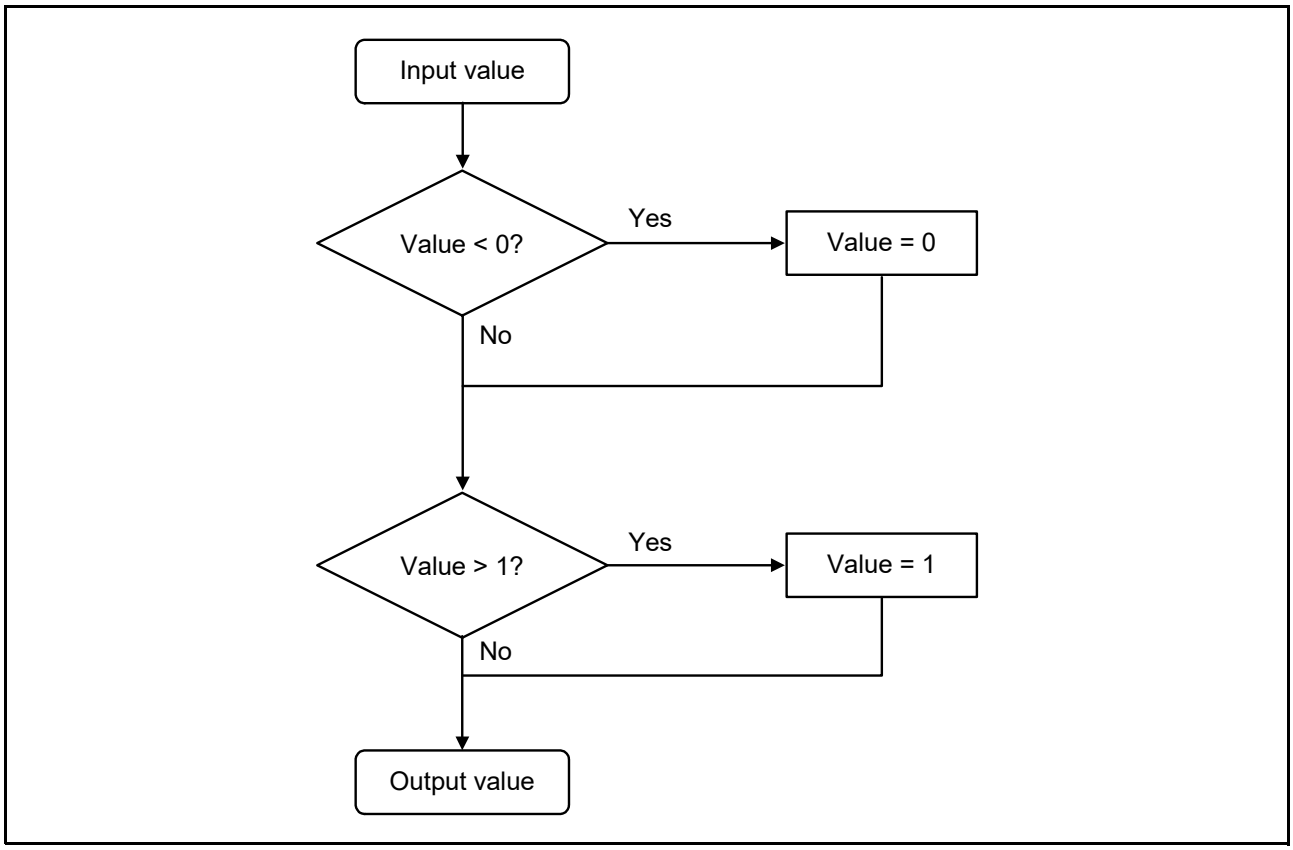


Figure 43.12 Clamping unit

The clamping unit can be put into threshold mode, in which all values greater than 0.5 are set to 1, and all values below or equal to 0.5 are set to 0. This feature is used when anti-aliasing is not wanted, such as for shared edges.

43.6.2.5 Combiner Unit

The combiner unit can be operated in minimum mode and in maximum mode. In minimum mode the smaller value is output, and in maximum mode the larger value is output. The minimum mode represents the intersection, and the maximum mode represents the union of the two regions.

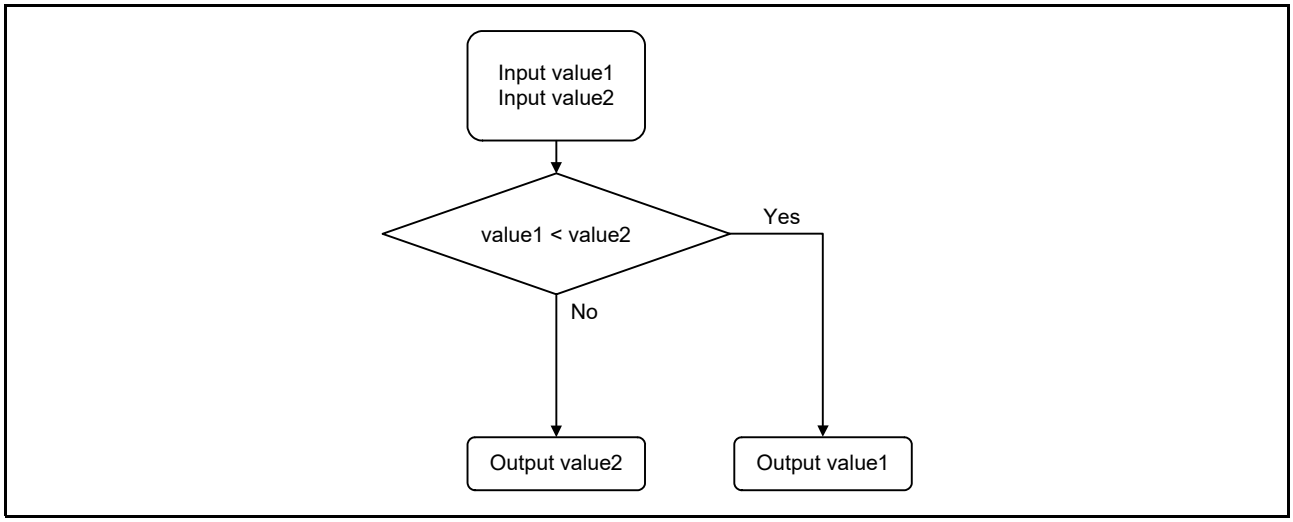


Figure 43.13 Combiner operated in minimum mode with intersection

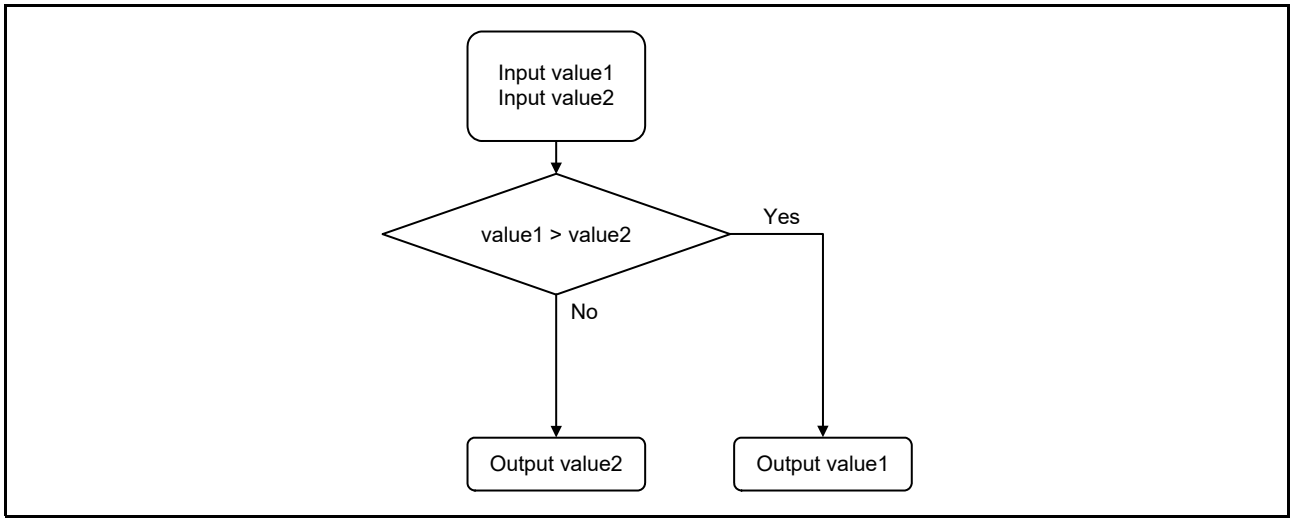


Figure 43.14 Combiner operated in maximum mode with union

43.6.2.6 Rasterization Optimization

During rasterization, it is necessary to step through the whole bounding box one pixel at a time. This requirement can lead to an unnecessary number of steps for pixels that are not drawn. The 2D Drawing Engine provides optimization methods designed to reduce the number of steps required during rasterization. One optimization relies on the fact that any convex primitive can have only one span per line (a span is a contiguous horizontal line). This form of optimization detects a span start and saves the information for the next line. Another optimization is to detect a span end and stop rasterization for the current line.

(1) Spanstore

Consider the case in Figure 43.15.

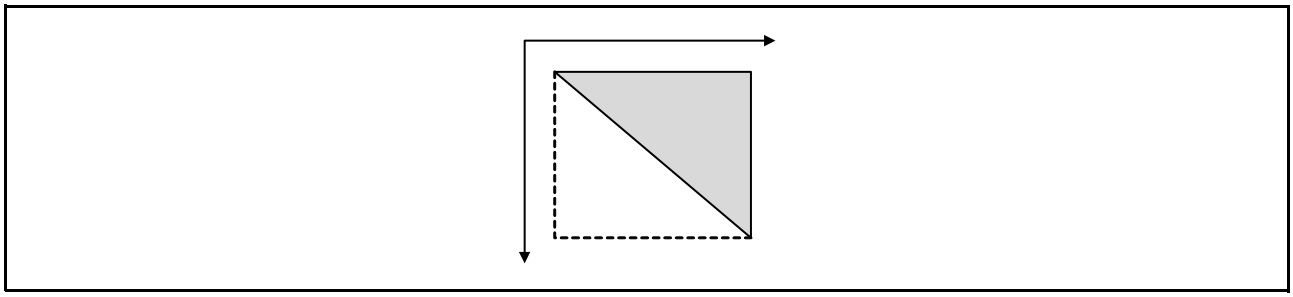


Figure 43.15 Triangle where the first edge is monotone growing

If the gray triangle must be rendered, half of the pixels processed by the rendering engine in the dotted bounding box would not be drawn. This situation can be optimized with the spanstore operation. When this operation is activated and a span start is detected, the x position of the start is detected.
In the next line the rendering starts with the stored x position. This only works if the edge is monotonically increasing ($y_1 > y_2 \Rightarrow x_1 \geq x_2$).
Consider the case in Figure 43.16.

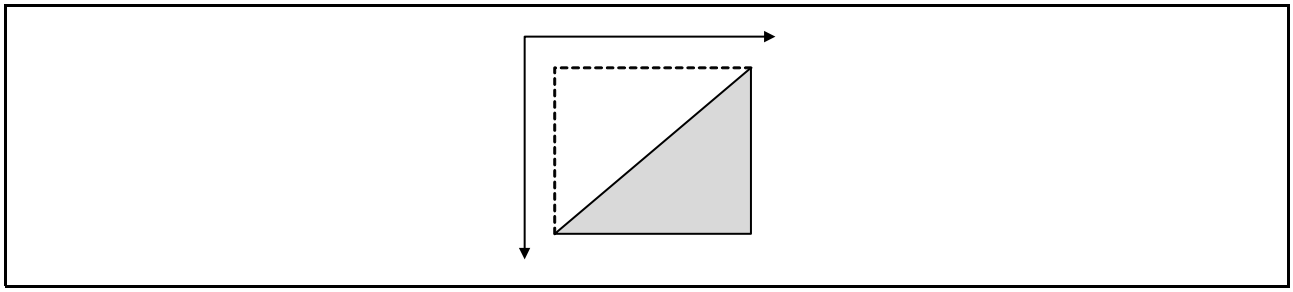


Figure 43.16 Triangle where the first edge is monotone falling

In this case, the normal spanstore operation cannot be performed. For this, the y direction of the rendering is reversed, and spanstore can operate again.

Consider the last case in Figure 43.17.

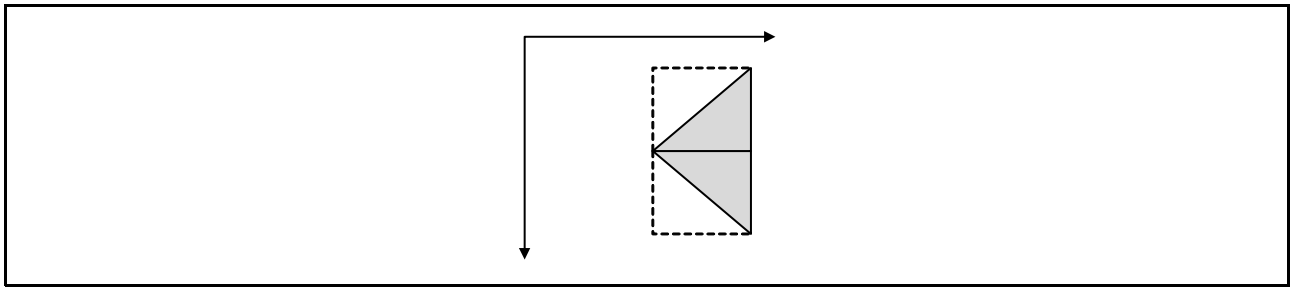


Figure 43.17 Triangle where the first edge is first monotone falling and then monotone growing

In this case, the triangle must be split and rendered as two parts for the spanstore optimization to work.
It is also possible to delay spanstore activation for a number of lines. This approach is used for rasterizing circles, as shown in Figure 43.18.

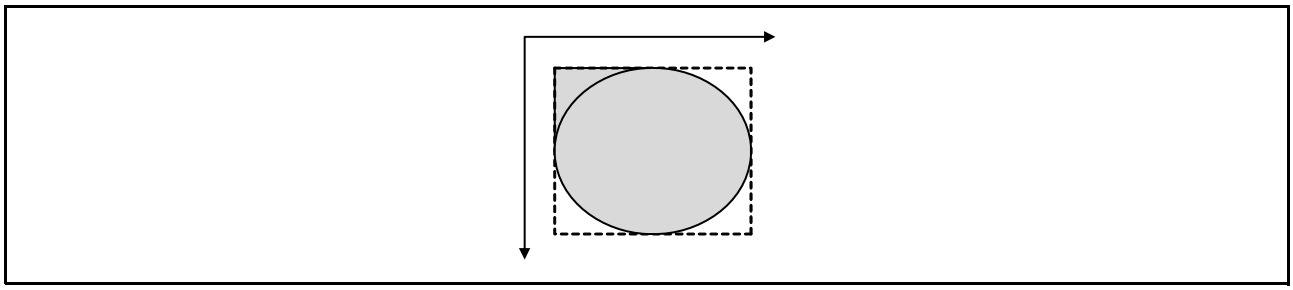


Figure 43.18 Full circle where the first edge is monotone falling for the first half and monotone growing for the second half

In this case, spanstore cannot be activated in the top left corner but can be activated in the bottom left corner. The empty corners in the top right and the bottom right cannot be rasterized because of the spanabort optimization.

(2) Spanabort

The second optimization assumes that the object that must be drawn is convex, which means there is only one span per scan line to be drawn. A non-convex object includes an object such as a triangle that is not filled and only consists of a thick border.
For a convex object, the rasterization can be stopped when the end of the first span is detected. No other constraints apply to this optimization for convex objects.

(3) Optimization efficiency

The efficiency of the optimizations can be seen for a typical case in Figure 43.19. In this case, a triangle is always rendered as a single piece and is not separated into multiple triangles for higher optimizations. For this, the spanstore delay is used.

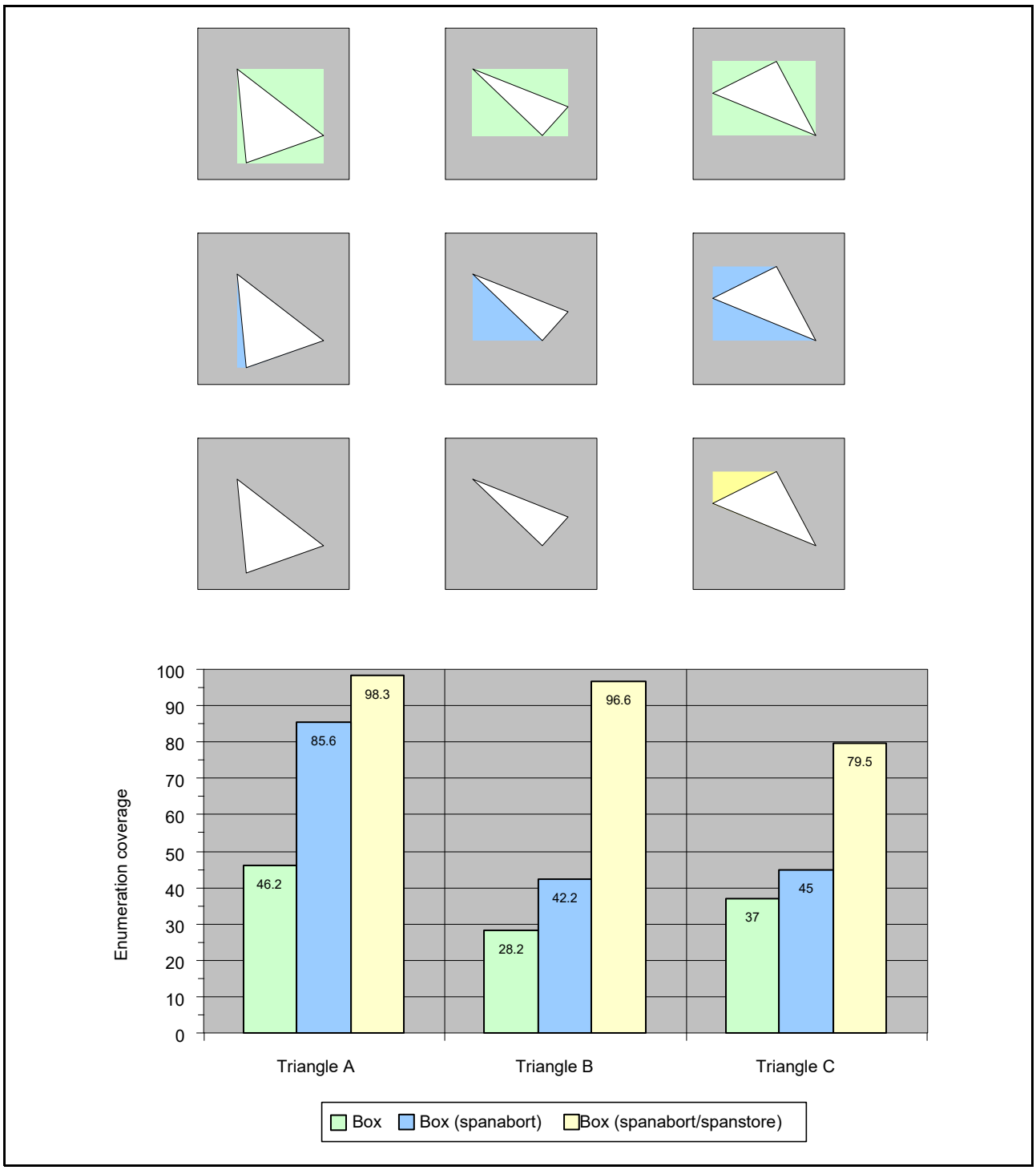


Figure 43.19 Efficiency of spanstore and spanabort optimizations with enumeration coverage equal to {pixels of primitive/pixels of bounding box}

43.6.3 Texturing

The texture unit can cover any primitive with a picture. The picture can be stretched, sheared, rotated, and translated in one step. To avoid aliasing, the result can be filtered bilinear in the u and v directions.

43.6.3.1 Mathematical Background

The arbitrary mapping problem is completely determined by a mapping from 3 points in the object space (x, y) to 3 points in the texture space (u, v).
Consider the following mapping:

$$\vec{p}_0 = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} \Rightarrow (\tilde{\vec{p}}_0) = \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\vec{p}_1 = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} \Rightarrow (\tilde{\vec{p}}_1) = \begin{pmatrix} u_1 \\ v_1 \end{pmatrix} = \begin{pmatrix} w \\ 0 \end{pmatrix}$$

$$\vec{p}_2 = \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} \Rightarrow (\tilde{\vec{p}}_2) = \begin{pmatrix} u_2 \\ v_2 \end{pmatrix} = \begin{pmatrix} 0 \\ h \end{pmatrix}$$

where w is the width of the texture and h is the height of the texture.
Examine Figure 43.20 in the object space. To simplify calculations the difference vectors are taken as calculations:

$$\vec{d}_1 = \vec{p}_1 - \vec{p}_0$$
$$\vec{d}_2 = \vec{p}_2 - \vec{p}_0$$

This is equivalent to transforming from coordinate system O to coordinate system O'.

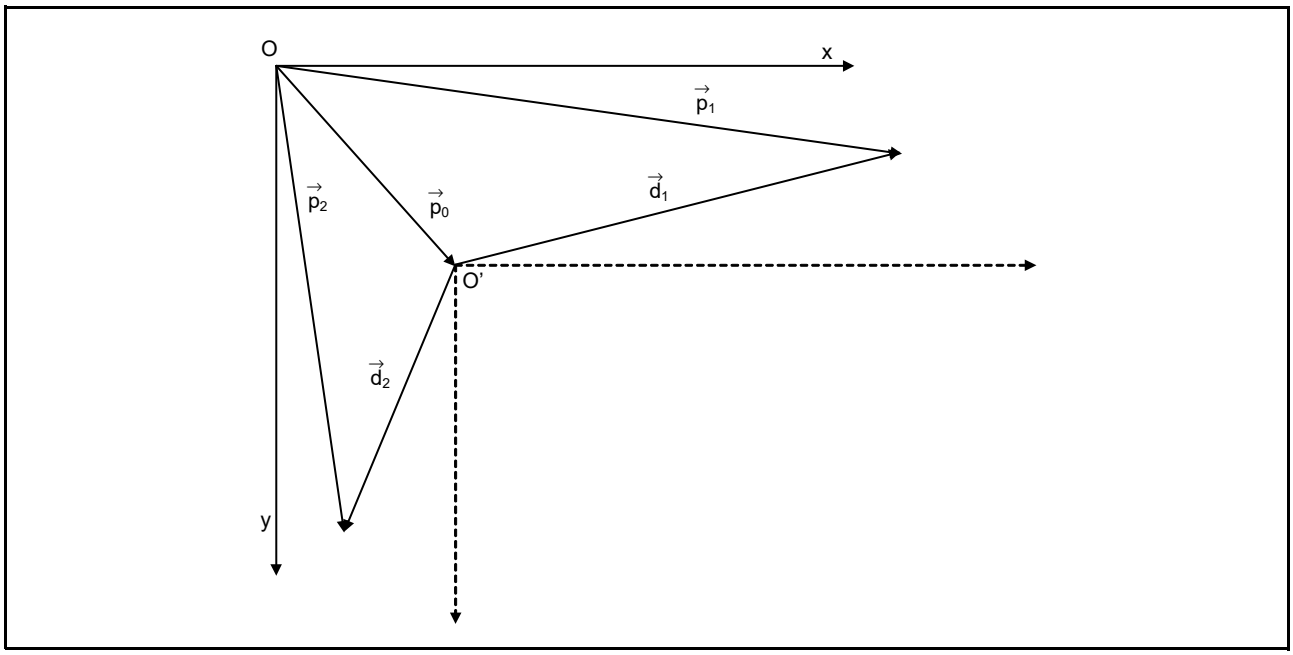


Figure 43.20 Texture mapping, object space, and transformation from coordinate system O to O' to simplify calculations

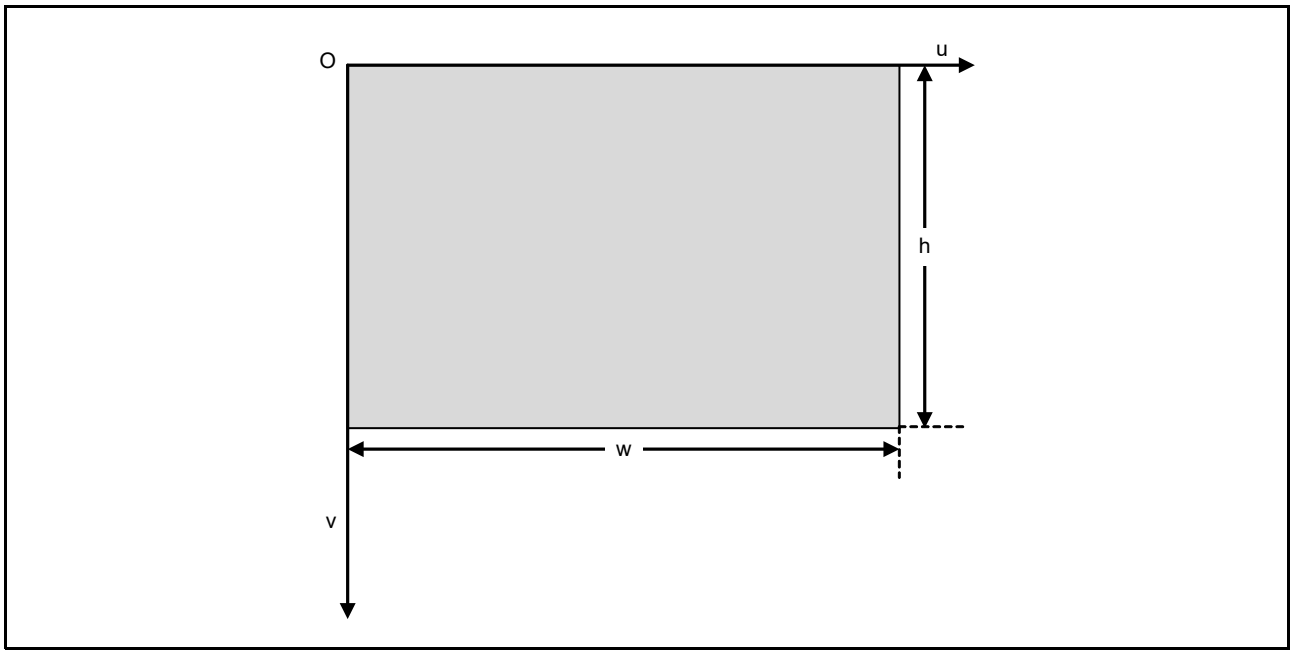


Figure 43.21 Texture mapping, texture space, and texture with width w and height h

In O', the mapping is:

$$\vec{p}'_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \Rightarrow (\tilde{\vec{p}}_0) = \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\vec{d}_1 = \begin{pmatrix} dx_1 \\ dy_1 \end{pmatrix} \Rightarrow (\tilde{\vec{p}}_1) = \begin{pmatrix} u_1 \\ v_1 \end{pmatrix} = \begin{pmatrix} w \\ 0 \end{pmatrix}$$

$$\vec{d}_2 = \begin{pmatrix} dx_2 \\ dy_2 \end{pmatrix} \Rightarrow (\tilde{\vec{p}}_2) = \begin{pmatrix} u_2 \\ v_2 \end{pmatrix} = \begin{pmatrix} 0 \\ h \end{pmatrix}$$

This is a linear mapping that can be described by a 2 x 2 matrix.

$$(\tilde{\vec{p}}) = M \cdot \vec{p}' = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \cdot \vec{p}'$$

$$\Rightarrow \begin{pmatrix} w \\ 0 \end{pmatrix} = M \cdot \vec{d}_1 \wedge \begin{pmatrix} 0 \\ h \end{pmatrix} = M \cdot \vec{d}_2$$

If the equations are expanded and sorted, the result is two equation systems each with two unknowns. These can be described more easily with a new matrix.

Let $A = \begin{bmatrix} dx_1 & dy_1 \\ dx_2 & dy_2 \end{bmatrix}$ then the equation system can be rewritten as:

$$\begin{pmatrix} w \\ 0 \end{pmatrix} = A \cdot \begin{pmatrix} m_{11} \\ m_{12} \end{pmatrix} \wedge \begin{pmatrix} 0 \\ h \end{pmatrix} = A \cdot \begin{pmatrix} m_{21} \\ m_{22} \end{pmatrix}$$

This can be easily solved with determinants.

$$\text{Let } c = \frac{1}{\det A} = \frac{1}{dx_1 \cdot dy_2 - dx_2 \cdot dy_1}$$

The resulting constants are:

$$m_{11} = c \cdot w \cdot dy_2 = \frac{du}{dx}$$
$$m_{12} = -c \cdot w \cdot dx_2 = \frac{du}{dy}$$
$$m_{21} = c \cdot h \cdot dx_1 = \frac{dv}{dx}$$
$$m_{22} = -c \cdot h \cdot dx_2 = \frac{dv}{dy}$$

To calculate the start values for u and v at the top of the bounding box, the transformation from O' to O must be reversed. Let us and vs be the start values, then:

$$\begin{pmatrix} u_s \\ v_s \end{pmatrix} = M \cdot (-\vec{p}_0) = c \cdot \begin{pmatrix} -w \cdot (x_0 \cdot dy_2 - y_0 \cdot dx_2) \\ h \cdot (x_0 \cdot dy_1 - y_0 \cdot dx_1) \end{pmatrix}$$

Examples

U and v are in the texture space. Enter the following into any case:

- Copy case:
  dx1 = 1, dx2 = 0, dy1 = 0, dy2 = 1
- Scaling case x scaling copy case:
  dx1 = f, dx2 = 0, dy1 = 0, dy2 = 1
  with f being the scaling factor in the x direction similar for the y direction
- Rotation case:
  dx1 = cosa, dx2 = -sina, dy1 = sina, dy2 = cosa
  with the angle between d1 and the x axis in the clockwise direction.

43.6.3.2 Limiter Operation

The texture limiters operate exactly in the same way as the linear limiters shown in Figure 43.8.
The register layout for the u limiter is the same:

- LUSTART = us
- LUXADD = du/dx
- LUYADD = du/dy.

The register layout for the v limiter is slightly different. TEXPITCH is multiplied to save one hardware multiplier.

- LVSTARTI = floor (vs) • TEXPITCH
  Contains the integer part of the start value.
- LVSTARTF = (vs-floor (vs)) • TEXPITCH
  Contains the fractional part of the start value.
- LVXADDI = floor (dv/dx) • TEXPITCH
  Contains the integer part of dv/dx.
- LVYADD = floor (dv/dy) • TEXPITCH
  Contains the integer part of dv/dy.
- LVYXADDF.LVXADDF = (dv/dx - floor (dv/dx)) • TEXPITCH
  Contains the fractional part of dv/dx
- LVYXADDF.LVYADDF = (dv/dy - floor (dv/dy)) • TEXPITCH
  Contains the fractional part of dv/dy
- TEXMASK.TEXUMASK, TEXMASK.TEXVMASK
  Contains a mask for u and v separately to wraparound values of u and v. This is useful for staying inside the limits of the texture or repeat a texture. Wrap around textures have to have a size multiple of 2.
- TEXPITCH
  Contains the width of the texture in pixels in framebuffer memory. This information is required to calculate the new address if stepping to a new line.
- TEXORIGIN.
  Contains the base address of the texture.

43.6.4 Colorization

After a pixel is found to be part of the geometry, its color is calculated. The 2D Drawing Engine supports a very general color calculation scheme, allowing support for several color modes. This color scheme uses an interpolation between two color registers, COLOR1 and COLOR2. See Figure 43.22 for details. COLOR1 and COLOR2 are marked as A, B in the figure for clarity.

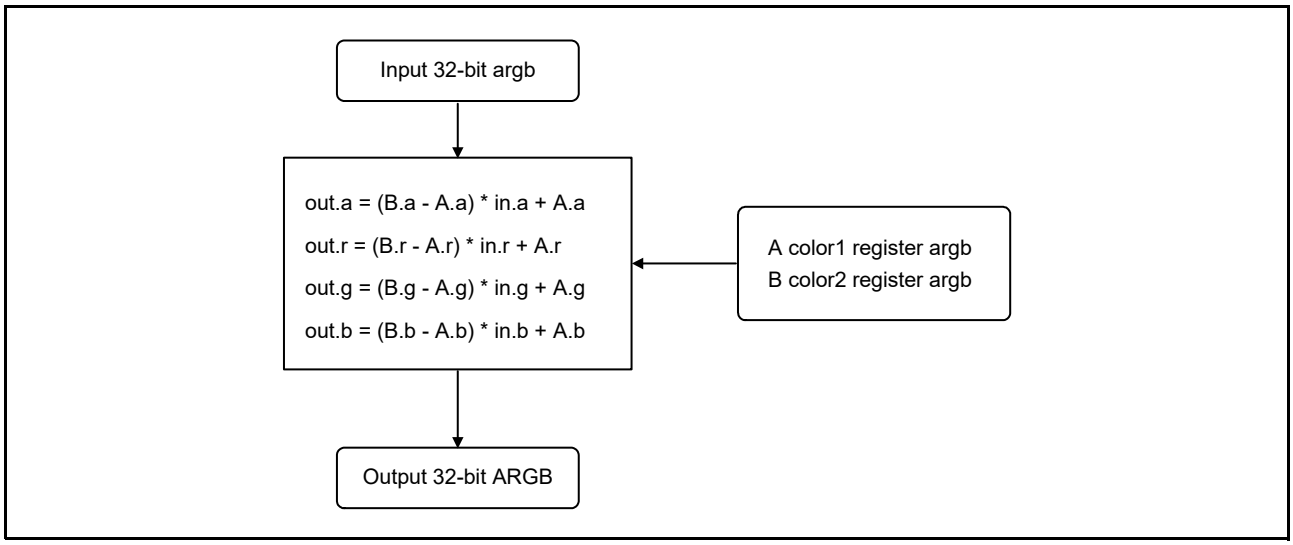


Figure 43.22 Colorization step and interpolation between the two color registers, A (COLOR1) and B (COLOR2)

This general approach can be used to support several different color modes that can be individually applied to any color or alpha channel of the input.

Table 43.7 Colorization operations

Operation	Settings for A and B *1
Copy	A = 0, B = ffh
Replace with a constant value v	A = v, B = v
Multiply by a constant value v	A = 0, B = v
Colorize an alpha texture with the RGB value v	A.a = 0, A.r = B.r, A.g = B.g, A.b = B.b, B.a = ffh, B.r = v.r, B.g = v.g, B.b = v.b
Invert a channel	A = ffh, B = 0
Invert multiply with v	A = v, B = 0
Interpolate between color v and color u	A = v, B = u

Note 1. A = COLOR1, B = COLOR2

43.6.5 Blending

43.6.5.1 Color Channel Blending

The last step before the pixel is written to the framebuffer is to blend the pixel with the data that is already written to the framebuffer. If blending is activated, the framebuffer, referred to as DST, must be read. SRC is the output from the colorization unit.

The following color blend modes are supported:

- SRC_ZERO
- SRC_ONE
- SRC_ALPHA
- SRC_ONE_MINUS_ALPHA
- DST_ZERO
- DST_ONE
- DST_ALPHA
- DST_ONE_MINUS_ALPHA.

The selection of the color channel blend modes is performed with the following flags:

- BSF: blend source factor is alpha
- BSI: blend source factor invert
- BDF: blend destination factor is alpha
- BDI: blend destination factor invert.

The formula for the blending is:
$dst = src \cdot f_S + dst \cdot f_D$
where:
BSF = 0, BSI = 0 $\Rightarrow f_S = 1$
BSF = 1, BSI = 0 $\Rightarrow f_S = \alpha$
BSF = 0, BSI = 1 $\Rightarrow f_S = 0$
BSF = 1, BSI = 1 $\Rightarrow f_S = 1 - \alpha$
BDF = 0, BDI = 0 $\Rightarrow f_D = 1$
BDF = 1, BDI = 0 $\Rightarrow f_D = \alpha$
BDF = 0, BDI = 1 $\Rightarrow f_D = 0$
BDF = 1, BDI = 1 $\Rightarrow f_D = 1 - \alpha$.

Table 43.8 lists all possible color channel blend modes.

Table 43.8 Color channel blend modes

Mode	BSF	BSI	BDF	BDI	Blend equation
SRC_ONE DST_ONE	0	0	0	0	SRC + DST
SRC_ONE	0	0	0	1	SRC
SRC_ONE DST_ALPHA	0	0	1	0	SRC + DST × ALPHA
SRC_ONE DST_ONE_MINUS_ALPHA	0	0	1	1	SRC + DST × (1 – ALPHA)
SRC_ZERO DST_ONE	0	1	0	0	DST
SRC_ZERO DST_ZERO	0	1	0	1	0
SRC_ZERO DST_ALPHA	0	1	1	0	DST × ALPHA
SRC_ZERO DST_ONE_MINUS_ALPHA	0	1	1	1	DST × (1 - ALPHA)
SRC_ALPHA DST_ONE	1	0	0	0	SRC × ALPHA + DST
SRC_ALPHA	1	0	0	1	SRC × ALPHA
SRC_ALPHA DST_ALPHA	1	0	1	0	SRC × ALPHA + DST × ALPHA
SRC_ALPHA DST_ONE_MINUS_ALPHA	1	0	1	1	SRC × ALPHA + DST × (1 - ALPHA)
SRC_ONE_MINUS_ALPHA DST_ONE	1	1	0	0	SRC × (1 – ALPHA) + DST
SRC_ONE_MINUS_ALPHA	1	1	0	1	SRC × (1 – ALPHA)
SRC_ONE_MINUS_ALPHA DST_ALPHA	1	1	1	0	SRC × (1 – ALPHA) + DST × ALPHA
SRC_ONE_MINUS_ALPHA DST_ONE_MINUS_ALPHA	1	1	1	1	SRC × (1 – ALPHA) + DST × (1 – ALPHA)

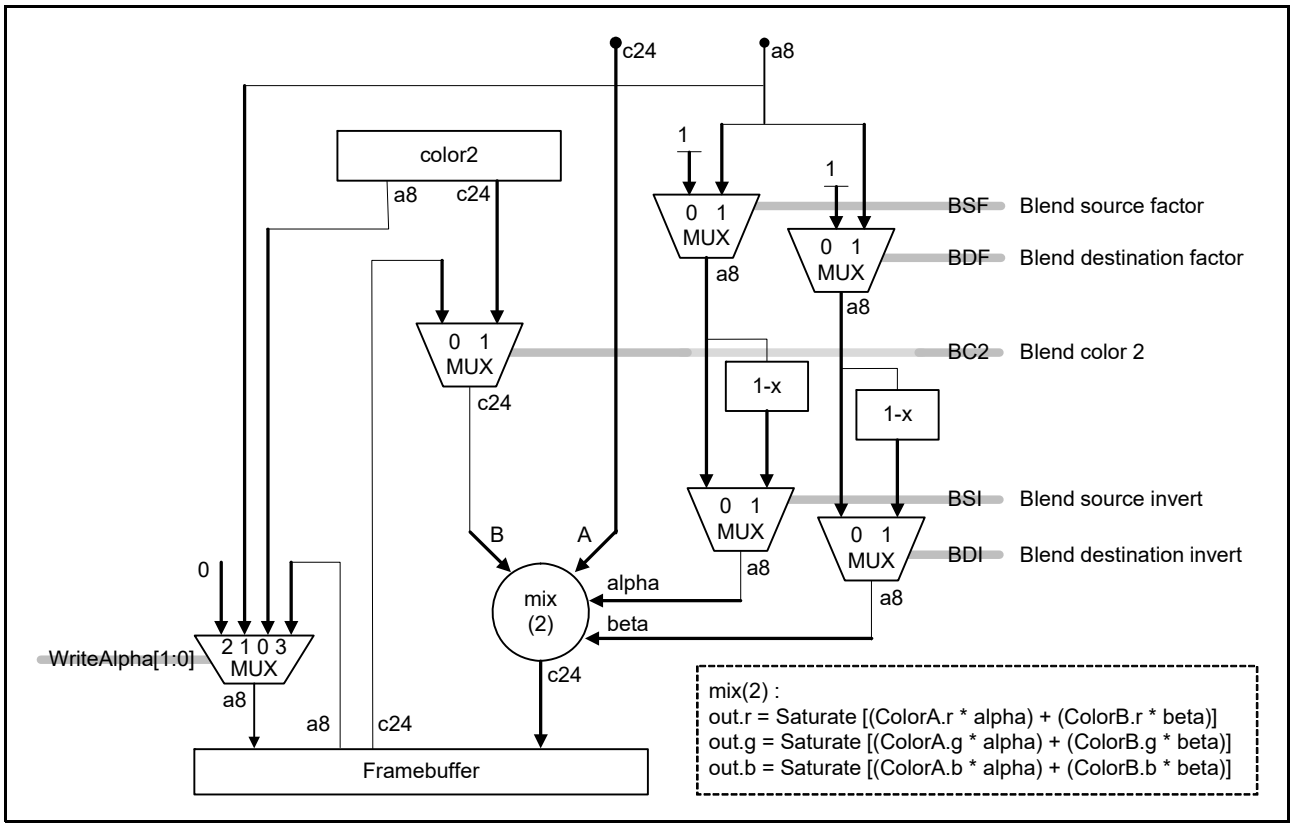


Figure 43.23 Color channel blend unit when CONTROL2.USEACB = 0

43.6.5.2 Alpha Channel Blending

The alpha channel can be blended in addition to the color channels. Alpha channel blending is enabled by setting CONTROL2.USEACB = 1. Alpha channel blending uses the same formulas and same blend modes as for the color channels. The alpha channel formulas can be set independently from the color channels.

The following alpha channel blend modes are supported:

- SRC_A_ZERO
- SRC_A_ONE
- SRC_A_SRC_A
- SRC_A_ONE_MINUS_SRC_A
- DST_A_ZERO
- DST_A_ONE
- DST_A_SRC_A
- DST_A_ONE_MINUS_SRC_A.

The alpha channel blend modes selected with the following flags:

- BSFA: blend source factor is SRC_A
- BSIA: blend source factor invert
- BDFA: blend destination factor is SRC_A
- BDIA: blend destination factor invert.

The formula for the blending is:
$dst\_alpha = src\_a \cdot f_{S\_a} + dst\_a \cdot f_{D\_a}$
where:
BSFA = 0, BSIA = 0 $\Rightarrow f_{S\_a} = 1$
BSFA = 1, BSIA = 0 $\Rightarrow f_{S\_a} = src\_a$
BSFA = 0, BSIA = 1 $\Rightarrow f_{S\_a} = 0$
BSFA = 1, BSIA = 1 $\Rightarrow f_{S\_a} = 1 - src\_a$
BDFA = 0, BDIA = 0 $\Rightarrow f_{D\_a} = 1$
BDFA = 1, BDIA = 0 $\Rightarrow f_{D\_a} = src\_a$
BDFA = 0, BDIA = 1 $\Rightarrow f_{D\_a} = 0$
BDFA = 1, BDIA = 1 $\Rightarrow f_{D\_a} = 1 - src\_a$.

Table 43.9 lists all possible alpha channel blend modes.

Table 43.9 Alpha channel blend modes

Mode	BSF	BSI	BDF	BDI	Blend equation
SRC_A_ONE DST_A_ONE	0	0	0	0	SRC_A + DST_A
SRC_A_ONE	0	0	0	1	SRC_A
SRC_A_ONE DST_A_SRC_A	0	0	1	0	SRC_A + DST_A × SRC_A
SRC_A_ONE DST_A_ONE_MINUS_SRC_A	0	0	1	1	SRC_A + DST_A × (1 – SRC_A)
SRC_A_ZERO DST_A_ONE	0	1	0	0	DST_A
SRC_A_ZERO DST_A_ZERO	0	1	0	1	0
SRC_A_ZERO DST_A_SRC_A	0	1	1	0	DST_A × SRC_A
SRC_A_ZERO DST_A_ONE_MINUS_SRC_A	0	1	1	1	DST_A × (1 - SRC_A)
SRC_A_SRC_A DST_A_ONE	1	0	0	0	SRC_A × SRC_A + DST_A
SRC_A_SRC_A	1	0	0	1	SRC_A × SRC_A
SRC_A_SRC_A DST_A_SRC_A	1	0	1	0	SRC_A × SRC_A + DST_A × SRC_A
SRC_A_SRC_A DST_A_ONE_MINUS_SRC_A	1	0	1	1	SRC_A × SRC_A + DST_A × (1 - SRC_A)
SRC_A_ONE_MINUS_SRC_A DST_A_ONE	1	1	0	0	SRC_A × (1 – SRC_A) + DST_A
SRC_A_ONE_MINUS_SRC_A	1	1	0	1	SRC_A × (1 – SRC_A)
SRC_A_ONE_MINUS_SRC_A DST_A_SRC_A	1	1	1	0	SRC_A × (1 – SRC_A) + DST_A × SRC_A
SRC_A_ONE_MINUS_SRC_A DST_A_ONE_MINUS_SRC_A	1	1	1	1	SRC_A × (1 – SRC_A) + DST_A × (1 – SRC_A)

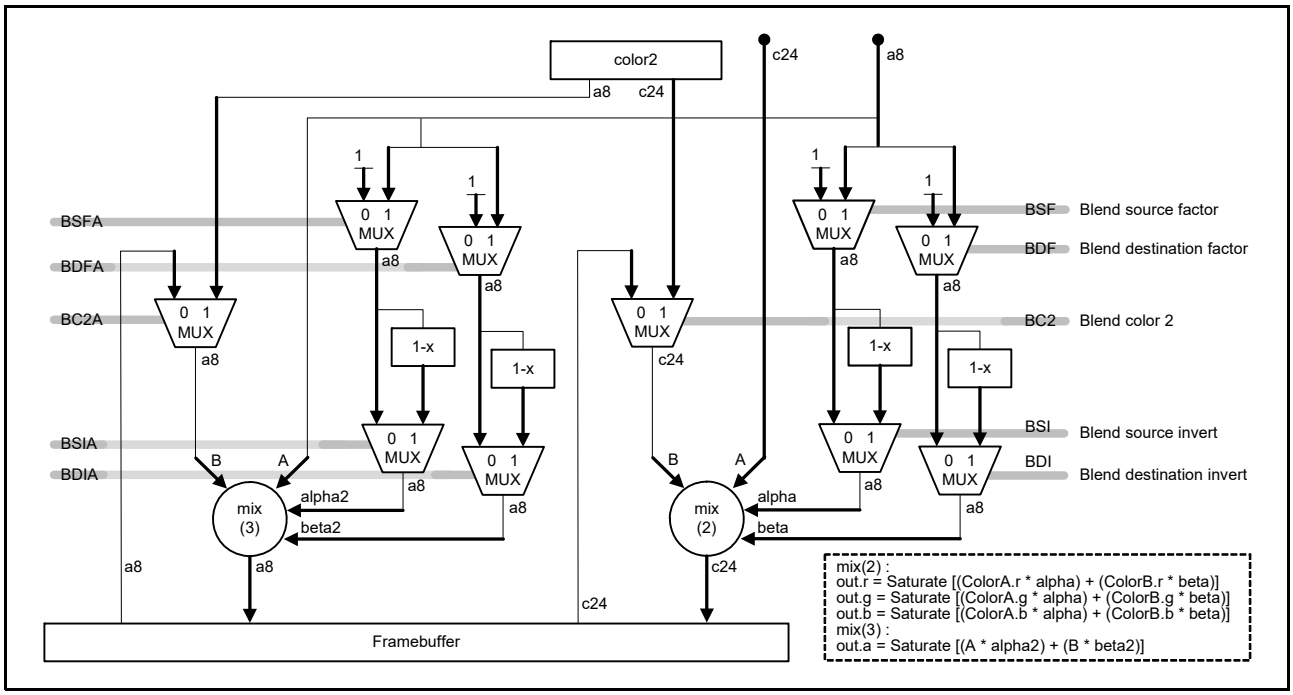


Figure 43.24 Alpha- and color-channel blend unit when CONTROL2.USEACB = 1

43.7 Rendering Modes

The rendering process can be performed in two different modes, register mode and display list mode.

43.7.1 Register Mode

In register mode, when operation is based on register settings, the host CPU configures and initiates each render process separately. To start a new render process, the host CPU must wait until the previous one is completed. In this mode, the host CPU is heavily engaged throughout the entire drawing procedure and is consequently to a large extent unavailable for other tasks.
The host CPU must set up all registers for performing a certain drawing operation before it can start the rendering process. A new register setup can only be started when the previous render process has completed. Before starting a new register setup, make sure that:

- STATUS.DLISTACTIVE = 0: Display list reader is idle
- STATUS.BUSYENUM = 0: Pixel selection unit is idle.

Lastly, write the framebuffer start address to the ORIGIN register. This write triggers the 2D Drawing Engine to start rendering.

43.7.2 Display List Mode

In display list mode, the host CPU creates a display list in memory prior to starting the 2D Drawing Engine. A display list contains a bundle of render operations. When started, the 2D Drawing Engine executes the display list autonomously, and the host CPU is not involved with drawing operations most of the time. Use of a display list allows for fully asynchronous operation of the host CPU and the 2D Drawing Engine and offers the best possible system performance. In this mode, the display list reader reads a memory block containing instructions on how to set the 2D Drawing Engine control registers and executes these control register writes accordingly.

Display list start

To start execution of a display list, which already resides in memory, the start address of the display list is written to the display list start address register DLISTSTART. Because rewriting DLISTSTART also stops any ongoing display list execution, make sure that the previous display list process is completed by either of the following two methods:

- Check that STATUS.DLISTACTIVE = 0, which indicates idle status of the display list reader
- Wait for the display list interrupt DRWDLISTIRQ, which indicates completion of the previous display list process.

Note: Direct writing to 2D Drawing Engine registers while the display list mode is active (STATUS.DLISTACTIVE = 1) might lead to a 2D Drawing Engine hang-up. To prevent this, always check that the display list reader is idle (STATUS.DLISTACTIVE = 0) before writing to any 2D Drawing Engine register.

Display list format

Display lists are stored using direct register-to-value mappings, which means that a display list contains a one-byte index that addresses a certain register and the value to be written to the register. The register index is derived from the address offset of the register address and can be calculated by dividing the address offset by 4. For the index of each register, see Table 43.11.
As the 2D Drawing Engine registers are always 32 bits wide, each data unit (called a data word) to be written to a register is of the same size. An address word that contains the indices of the register to be written is stored in a packed notation with up to four indices stored in one 32-bit address word.
A display list command always starts with an address word, followed by up to four data words, one for each register. The indices are read and interpreted from LSB to MSB, so the register of the low byte index is written first.

Example

In the following example:

- DWORD 201A 1930h // start of list address word
- DWORD 0000 0013h // data word 1 (for register 30h)
- DWORD FFFF FFAAh // data word 2 (for register 19h)
- DWORD 4033 6480h // data word 3 (for register 1Ah)
- DWORD 0001 0000h // data word 4 (for register 20h)
- DWORD... // next address word.

This stream of dwords updates the DRW registers as follows:

- Write 0000 0013h to register 30h = 48, which is IRQCTL
- Write FFFF FFAAh to register 19h = 25, which is COLOR1
- Write 4033 6480h to register 1Ah = 26, which is COLOR2
- Write 0001 0000h to register 20h = 32, which is ORIGIN.

Address word indices

Besides referencing a register, the indices of an address word can also have other meanings, depending on their value.

Table 43.10 Indices function

Index	Function	
00h to 7Fh	Register indices Two register indices trigger additional actions:	
	- 20h = 32:	A write to ORIGIN to set a new frame buffer address is delayed until the ongoing frame buffer write-back is complete, when STATUS.BUSYWRITE = 0.
	- 32h = 50:	A write to DLISTSTART sets a new display list start address stops the current display list and starts the new one.
80h	Gap index, which is used to fill unused bytes of an address word. For example, if fewer than four indices are required, the remaining bytes are filled with 80h. In this case, the number of the subsequent data words must be reduced accordingly.	
FFh	If the first index of an address word contains the special index FFh, the subsequent (second) index is interpreted as follows:	
	- Bit [0] set:	Display list end.
	- Bit [1] set:	Issue a full pipeline flush and wait (necessary before flip).
	- Bit [2] set:	Wait for writeback complete (necessary before framebuffer format change).
	- Bits [7:3]	Set all to 0s.
	Bit [1] and [2] settings are mutually exclusive. All indices after the special index FFh are ignored, and the next address word is read, if no display list end (bit [0] = 1) was set.	
	The remaining two indices must be set to 00h.	

Note: Gap indices 80h must not be placed between other indices, for example as “index1 - 80h - index3 - index4”. Always fill all indices after 80h with the gap index.
If any of the special indices 80h and FFh are used, no register index can follow after them in the same address word.

Table 43.11 2D Drawing Engine registers overview

Register function	Symbol	Index
Control registers:		
Geometry control	CONTROL	0
Surface control	CONTROL2	1
Interrupt control	IRQCTL	48
Cache control	CACHECTL	49
Status control	STATUS	n.a.*1
Hardware version and feature set ID	HWREVISION	n.a.*1
Color registers:		
Base color	COLOR1	25
Secondary color	COLOR2	26
Pattern	PATTERN	29
Limiter registers:		
Limiter 1 start value	L1START	4
Limiter 2 start value	L2START	5
Limiter 3 start value	L3START	6
Limiter 4 start value	L4START	7
Limiter 5 start value	L5START	8
Limiter 6 start value	L6START	9
Limiter 1 x-axis increment	L1XADD	10
Limiter 2 x-axis increment	L2XADD	11
Limiter 3 x-axis increment	L3XADD	12
Limiter 4 x-axis increment	L4XADD	13
Limiter 5 x-axis increment	L5XADD	14
Limiter 6 x-axis increment	L6XADD	15
Limiter 1 y-axis increment	L1YADD	16
Limiter 2 y-axis increment	L2YADD	17
Limiter 3 y-axis increment	L3YADD	18
Limiter 4 y-axis increment	L4YADD	19
Limiter 5 y-axis increment	L5YADD	20
Limiter 6 y-axis increment	L6YADD	21
Limiter 1 bandwidth parameter	L1BAND	22
Limiter 2 bandwidth parameter	L2BAND	23
Texture registers:		
Texture base address	TEXORIGIN	47
Texels per texture line	TEXPITCH	45
Texture size or texture address mask	TEXMASK	46
U limiter start value	LUSTART	36
U limiter x-axis increment	LUXADD	37
U limiter y-axis increment	LUYADD	38
V limiter start value integer part	LVSTARTI	39
V limiter start value fractional part	LVSTARTF	40
V limiter x-axis increment integer part	LVXADDI	41
V limiter y-axis increment integer part	LVYADDI	42
V limiter increment fractional parts	LVYXADDF	43

Table 43.11 2D Drawing Engine registers overview

Register function	Symbol	Index
Color lookup table start address	TEXCLADDR	55
Write Data to TEXCLADDR; after each data write, TEXCLADDR is incremented by 1.	TEXCLDATA	56
Offset to the index for the indexed texture formats i8, i4, i2, and i1	TEXCLOFFSET	57
Compare value for R,G, B components of internal texel color representation.	COLKEY	58
Miscellaneous registers:		
Bounding box dimension	SIZE	30
Framebuffer pitch and spanstore delay	PITCH	31
Address of the first pixel in framebuffer	ORIGIN	32
Display list start address	DLISTSTART	50
Performance counters control	PERFTRIGGER	53
Performance counter 1	PERFCOUNT1	51
Performance counter 2	PERFCOUNT2	52

Note 1. These registers are read-only and cannot be accessed in display list mode, and they therefore have no index.

43.7.3 Stopping the Render Process

Stopping an ongoing render process requires a specific procedure, which is described in section 43.10, Stopping the 2D Drawing Engine Render Process.

43.8 Interrupts

The 2D Drawing Engine generates three interrupts:

- DRWBUSIRQ
- DRWENUMIRQ
- DRWDLISTIRQ.

43.8.1 Interrupt sources

DRWBUSIRQ

This is the 2D Drawing Engine bus error interrupt. It occurs when the 2D Drawing Engine attempts to access an undefined address range through the following:

- Framebuffer Base Address Register (ORIGIN)
- Texture Base Address Register (TEXORIGIN)
- Display List Start Address Register (DLISTSTART)

The interrupt source can be determined by the BUSERRMFB, BUSERRMTXMRL, and BUSERRMDL bits of the STATUS register.

Note: After a DRWBUSIRQ occurrence, you must apply a software reset.
For a software reset, see section 52, Power-Down Modes.

DRWENUMIRQ

This is the current render process finished interrupt.

DRWDLISTIRQ

This is the display list interrupt. It is asserted on completion of a display list process. DRWDLISTIRQ is activated if one of the following is true:

- The entire display list is complete
- The display list processing stops because a new display list start is triggered by a write to the Display List Start Address Register (DLISTSTART).

43.8.2 Interrupt Control

The three 2D Drawing Engine interrupts are combined into a single shared interrupt, DRW_IRQ, to the CPU. Each interrupt can be masked (disabled), or unmasked (enabled) by setting its associated enable bit in the Interrupt Control Register (IRQCTL).
The occurrence of an enabled interrupt (one with its mask bit set to 1 in IRQCTL) is monitored in the Status Control Register (STATUS) with its associated interrupt status bit is set to 1. The shared 2D Drawing Engine interrupt DRW_IRQ is then generated.
To clear the interrupt, the host CPU must write 1 to the interrupt clear bit in IRQCTL. The interrupt clear bit returns to 0 automatically.

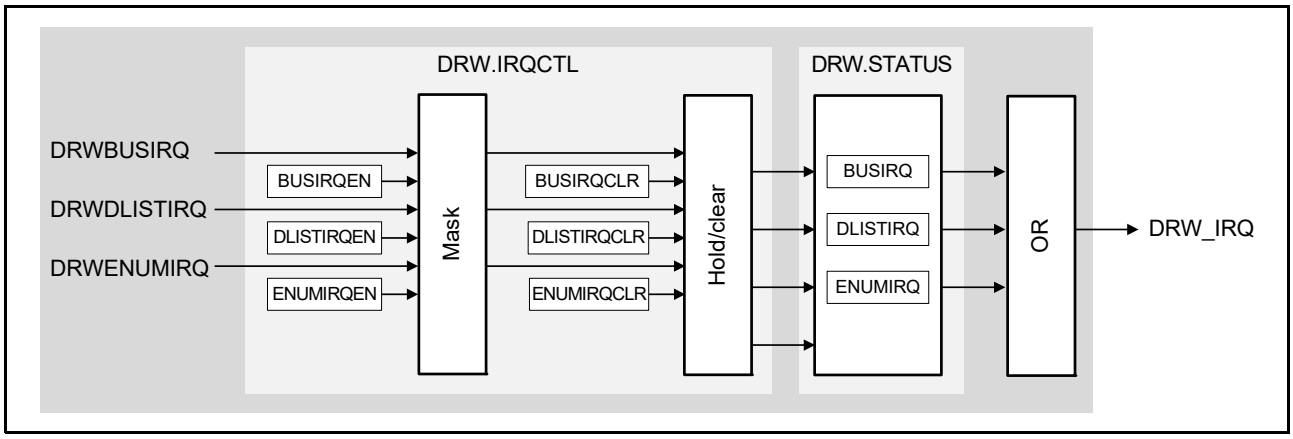


Figure 43.25 Interrupt Controller Unit (ICU)

43.9 Performance Counters

The 2D Drawing Engine features two independent 32-bit performance counter registers (PERFCOUNTk (k = 1, 2)) to count the number of occurrences of a certain event. The events to be counted can be set up independently for each performance counter register with the Performance Counter Control Registers, PERFTRIGGER.PERFTRIGGER2 for PERFCOUNT2 and PERFTRIGGER.PERFTRIGGER1 for PERFCOUNT1.
Table 43.12 lists the performance counter trigger events that can be selected.

Table 43.12 Performance counter trigger events

PERFTRIGGER.PERFTRIGGERk	Event
0	Disable performance counter
1	2D Drawing Engine active cycles
2	Framebuffer read access
3	Framebuffer write access
4	Texture read access
5	Invisible pixels (enumerated but selected with alpha 0%)
6	Invisible pixels while internal FIFO is empty (lost cycles)
7	Display list reader active cycles
8	Framebuffer read hits
9	Framebuffer read misses
10	Framebuffer write hits
11	Framebuffer write misses
12	Texture read hits
13	Texture read misses
31	Every clock cycle (for use as timer)

43.10 Stopping the 2D Drawing Engine Render Process

If a render process has started either in register or display list mode, the 2D Drawing Engine processes the data autonomously until the render process is finished. Depending on the rendering, this process might take several milliseconds.

If the 2D Drawing Engine is to be disabled because, for example, this LSI enters a low-power mode, proceed as follows to stop the ongoing rendering:

1. Set the following registers as follows:
   SIZE = 0001 0001h
   Set bounding box dimensions to 1 pixel x 1 line.
   CONTROL2 = 0000 0000h
   Color format a (8), no texture, CLUT.
   ORIGIN = UnmappedAddress
   The UnmappedAddress is an address that is not available for 2D Drawing Engine access.
   The recommended UnmappedAddress is given here under the key word “UnmappedAddress”.
   Alternatively do the same in display list mode:

```
DWORD 8020 011Eh          // start of list address word
DWORD 0001 0001h          // SIZE = 0001 0001h
DWORD 0000 0000h          // CONTROL2 = 0000 0000h
DWORD UnmappedAddress     // ORIGIN = UnmappedAddress
```

2. Wait for the Bus Error corresponding to the unmapped address violation, which indicates access to an unmapped address and the stop of the render process.
   UnmappedAddress = D000 0000h
3. Disable the 2D Drawing Engine, if wanted.

44. Sprite Engine (SPEA)

44.1 Overview of the Sprite and RLE Units

44.1.1 Units

This LSI has the following number of units of the Sprite Engine.

Table 44.1 Units

Sprite Engine (SPEA)	RZ/A2M
Units	1
Names	SPEA0

Unit index n

Throughout this section, the individual units of the Sprite Engines are identified by the index “n” (n = 0), for example SPEAnPHAi for the SPEAn physical address register of RLE Unit i.

44.1.2 Sprite and RLE Units indices

In this section following indices are used:

RLE Unit number index i

The one units of the RLE Units are identified by the index “i” (i = 0), for example SPEAnPHAi for the physical address register of RLE Unit i.

Sprite Unit number index k

The two units of the Sprite Units are identified by the index “k” (k = 0 to 1), for example SPEAnSkEN for the enable register of Sprite Unit k.

Sprite index m

The 16 sprites of a Sprite Unit k are identified by the index “m” (m = 0 to 15), for example SPEAnSkDAm for the destination address register of Sprite Unit k’s sprite m.

44.1.3 Register addresses

All Sprite Engines register addresses are given as address offsets from the individual base addresses <SPEAn_base>. The <SPEAn_base> addresses of each SPEAn are listed in the following table:

Table 44.2 Register base addresses <SPEAn_base>

SPEAn unit	<SPEAn_base> address
SPEA0	E803 E000H

44.2 Functional Overview

The video display controller 6 can select different ways to acquire their data from the memory for adding a layer to the image to be displayed:

- Direct access
  The color data to be processed by an The video display controller 6 is read directly in the correct format out of the memory.
  The graphics blocks (0), (2), and (3) of the video display controller 6 can directly access the memory.
- RLE (run-length encoded) layer
  RLE compressed color data in the memory is expanded and delivered to the video display controller 6 as separate pixel color data.
  The graphics block (0) of the video display controller 6, which determines the background layer, can uses an RLE layer.
- Sprite layer
  A layer with up to 16 rectangle areas, which can be placed anywhere on the layer area.
  Since only the color data for these rectangles - and not for the entire layer - need to be stored in memory, required memory occupation and bandwidth is minimized.
  The graphics blocks (2) and (3) of the video display controller 6 can use Sprite layers.

Features summary

- RLE Units
  - The RLE unit can cooperate with the graphics block (0) of the video display controller 6
  - RLE compressed data in Targa format
  - RLE compressed color data formats: 24 bpp
- Sprite Units
  - The two Sprite units can cooperate with graphics blocks (2) and (3) of the video display controller 6.
  - up to 16 separate sprites processed by each Sprite Unit

Indices

Throughout this section following indices are used:

- i = 0: RLE Unit number
- k = 0 to 1: Sprite Unit number
- m = 0 to 15: sprite number for each Sprite Unit

44.3 RLE Units Functional Description

The following figure shows the block diagram of the RLE Units.

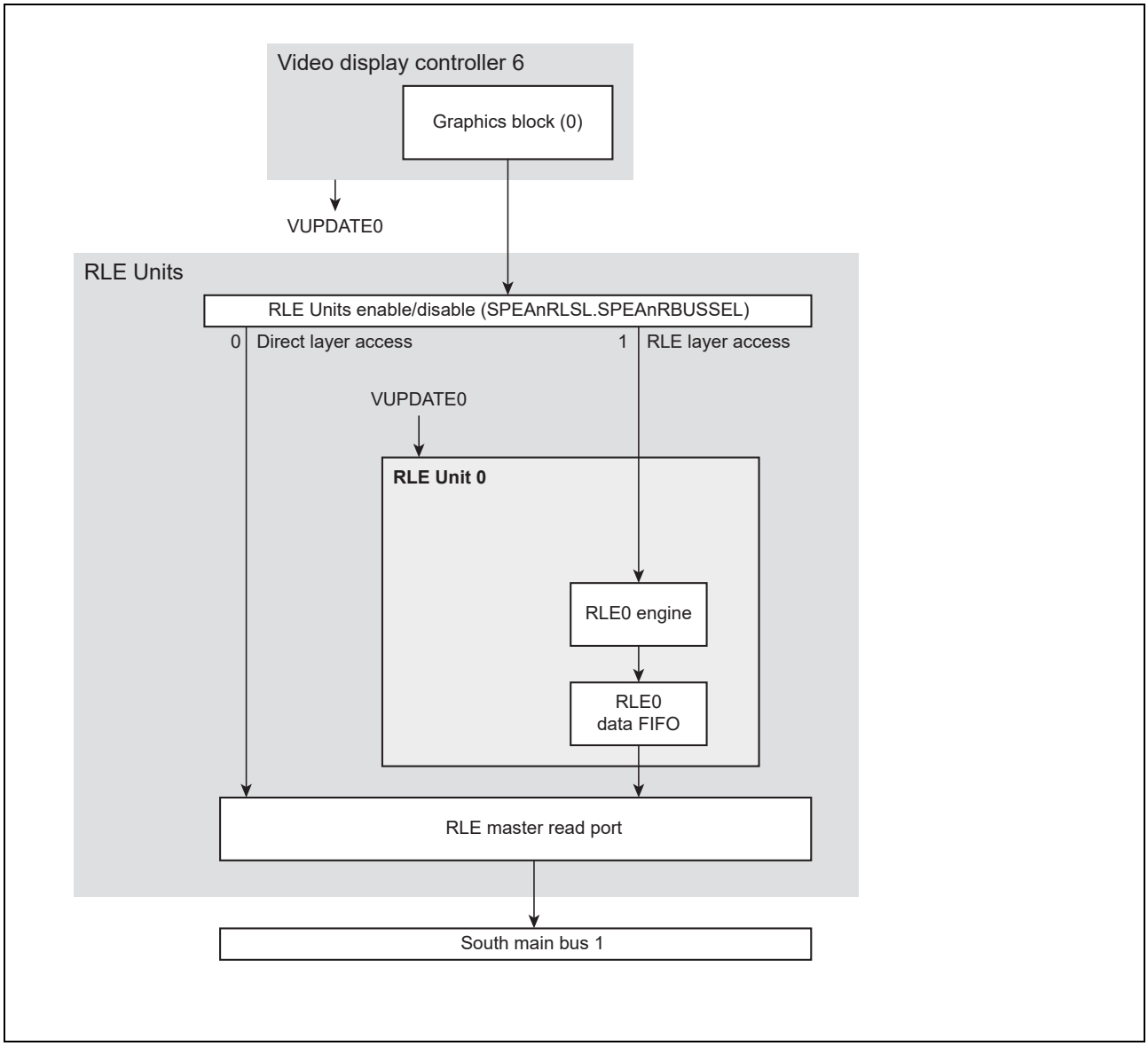


Figure 44.1 RLE Engines block diagram

The graphics block (0) of the video display controller 6 can read RLE compressed color data from the memory via an RLE Unit.
If the graphics block (0) of the video display controller 6 does not read RLE data, it accesses the physical memory for reading frame buffer data directly.

RLE Units enable

The RLE Units can be enabled and disabled:
RLE or direct layer selection is done by a bypass switch:

- SPEAnRLSL.SPEAnRBUSSEL = 0: RLE Units disabled
  The graphics block (0) of the video display controller 6 performs direct layer accesses to the memory.
- SPEAnRLSL.SPEAnRBUSSEL = 1: RLE Units enabled
  The graphics block (0) of the video display controller 6 performs RLE layer accesses.

RLE Units data FIFO

The data read from the memory by an RLE Unit is stored in a RLE data FIFO, that allows prefetching of memory data. The prefetching of data can be influenced by the FIFO fill stage and the length of the red bursts, the RLE issues read accesses towards the memory via the south main bus.
The size of RLE data FIFO is 1024 bit, organized as 16 x 64 bit.
Two values in the RLE prefetch configuration register SPEAnRCFG allow to configure the FIFO fill threshold when a new read burst is issued and the length of the burst:

- SPEAnRDTH[2:0] set the FIFO fill threshold for the next data prefetch
- SPEAnRLEN[2:0] set the burst length for the next data prefetch

Note: The burst size SPEAnRLEN[2:0] and the prefetch timing SPEAnRDTH[2:0] must be set to values, that prevent the RLE data FIFO from overflow.
The RLE data FIFO size is 1024 bit.
When operating the RLE unit, always set the GR0_BST_MD bit of the video display controller 6 to 1.

44.3.1 RLE layer definition

The RLE Unit i identifies a RLE layer read access of the video display controller 6 by the graphics block (0)'s read address of the first layer pixel.
The valid address is defined in the start address register SPEAnSTAi, that marks the virtual frame start address.

If the graphics block (0) of the video display controller 6 read access matches address, the RLE Unit reads RLE data from the memory address, defined by the physical address register SPEAnPHAi.
Since the entire graphics block (0) of the video display controller 6 layer data is read via the RLE Unit, the RLE Unit reads data from the memory consecutively, until the graphics block (0) of the video display controller 6 issues the start address SPEAnSTAi again. This causes the RLE Unit to restart data read from address SPEAnPHAi.

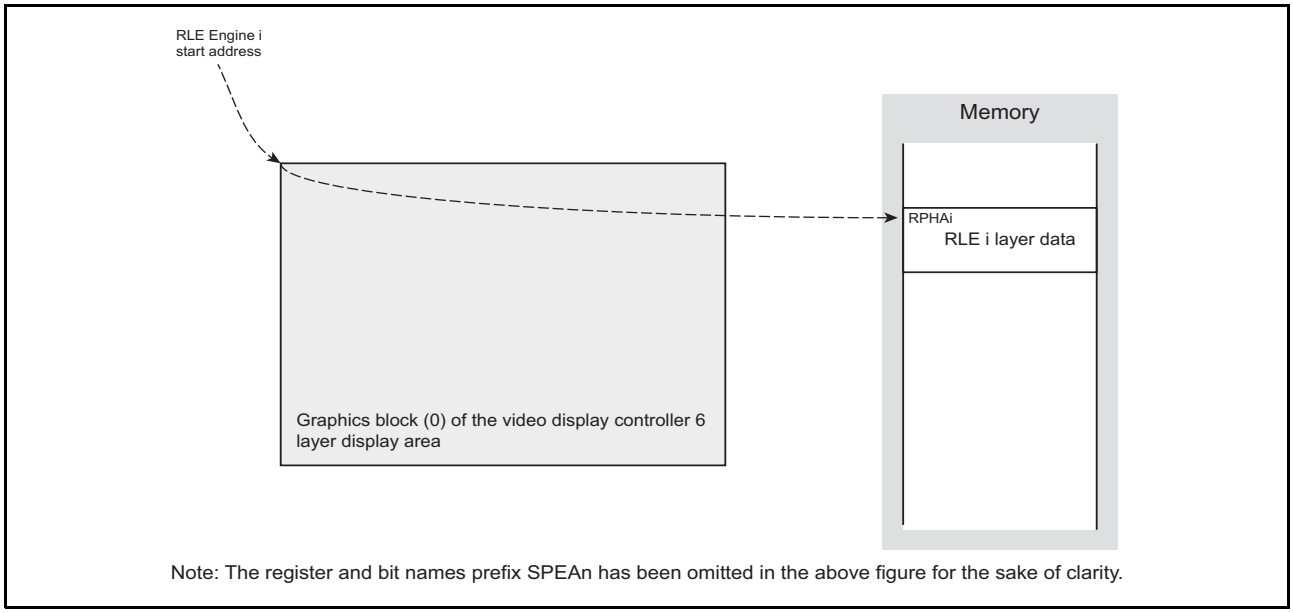


Figure 44.2 RLE definitions

As the RLE unit is reading the memory consecutively, the following graphics block (0) of the video display controller 6 restrictions need to be considered:

Note 1. Always set '1' to the GR0_BST_MD bit of the GR0_FLM1 register of the video display controller 6. If an image data size in the horizontal direction does not fit to 128 bytes alignment, some dummy data in each line needs to be added.
2. After completion of the reading of each line, the RLE unit reads additional 128 bytes. For this reason, 128 bytes of additional dummy data must be inserted in each line.

Example: Dummy data insertion required for each line in the horizontal 240 pixel image of 32bpp:
64- and 128-byte of dummy data are required to satisfy the restrictions of the Note 1 and Note 2, respectively. Dummy data insertion of 192 bytes in total is required. The number of horizontal pixels after dummy data insertion is 288 pixels.

44.3.2 Color modes

The following table lists the possible combinations of the

- color data formats selectable in the graphics block (0) of the video display controller 6
- RLE color mode selection and the data format delivered to the graphics block (0) of the video display controller 6
- color data format in the memory

when RLE data is used for the background layer.

Table 44.3 RLE Units supported color formats

	RLE Unit configuration		
Graphics block (0) of the video display controller 6 color format selection	Color mode selection	Graphics block (0) of the video display controller 6 data	Color data in the memory
32 bpp αRGB8888	SPEAnRCMi = 2: 24 bpp	RGB888 with α = 1	RGB888
24 bpp RGB888	SPEAnRCMi = 2: 24 bpp	RGB888	RGB888
Other	not supported	–	–

Color expansion

Depending on the color format the graphics block (0) of the video display controller 6 is expecting, the color data read from the memory is expanded to fit a 32-bit word.

The following diagrams show how memory data in different formats are passed to the graphics block (0) of the video display controller 6.

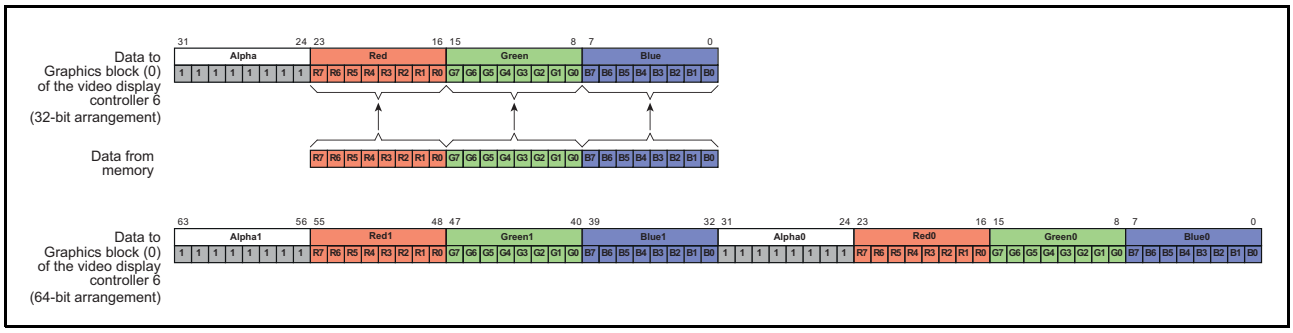


Figure 44.3 24 bpp color expansion to 32 bpp

44.3.3 RLE data packets in the memory

The RLE Units support Targa RLE data packet formats.
The Targa RLE packets comprise two types of data elements:

- run-length packets
- raw packets

The packet consists of two fields:

- Control byte: defines the type of data and the number of pixels covered by the packet
- Data field: defines the pixels color data

Table 44.4 RLE Targa packets

Packet type	Control byte		Data field
Run-length	• Packet type = 1 for run-length packet • Size: 1 bit	• Pixel count = (number of pixel repetitions with color of data field) - 1 • Size: 7 bit	• Pixel color • Size: pixel bpp
Raw	• Packet type = 0 for raw packet • Size: 1 bit	• Pixel count = (number of pixels in following data field) - 1 • Size: 7 bit	• Pixels color • Size: pixel bpp x pixel count

44.3.4 Packet arrangement in the memory

Padding bits

A data packet must always start at a byte boundary. Thus it may be necessary to add padding bits to the previous packet. The following diagrams show the arrangement of run-length and raw packets with different color format in the memory.

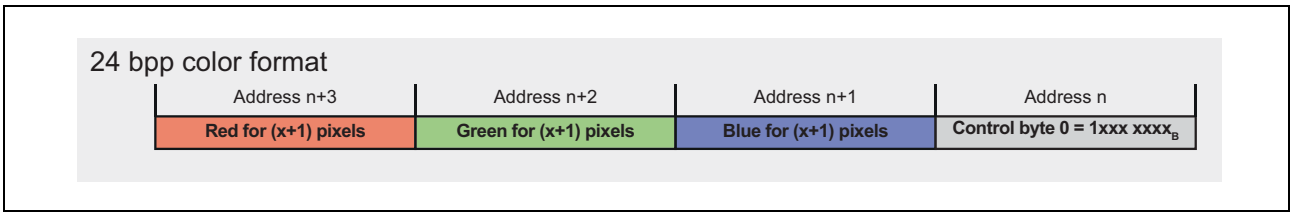


Figure 44.4 Run-length data packets in the memory

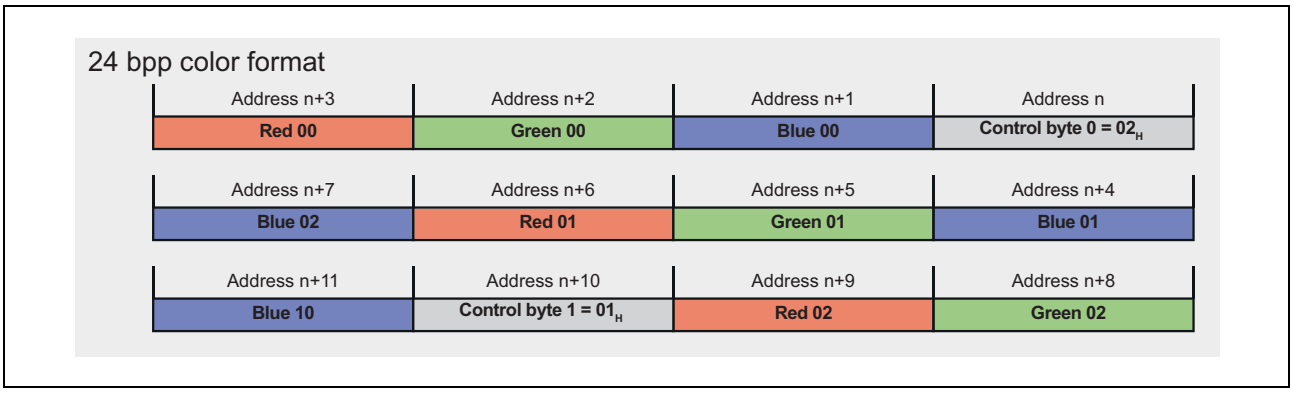


Figure 44.5 Raw data packets in the memory

44.3.5 RLE definition registers modification

The RLE definition registers are buffered.
While the RLE definition registers are in operation, the buffer registers can be modified without disturbing the video output.

The buffered RLE Unit i definition registers are:

- Start address register SPEAnSTAi
- physical address register SPEAnPHAi
- color mode register SPEAnRMCi

When setup of new RLE definitions is completed, the update of all definition registers of RLE Unit i must be requested via the SPEAnRUP register:

- SPEAnRUP.SPEAnRUP0 = 1: initiates update of all RLE Unit 0 definition registers with the next VUPDATE0

Caution: Caution: ?????LE definition buffer registers must not be modified while SPEAnRUPi = 1.

44.4 Sprite Units functional description

The graphics blocks (2) and (3) of the video display controller 6 read accesses to the memory are performed via the Sprite Units of the Sprite Engine.
The assignment of the graphics blocks (2) and (3) of the video display controller 6 to the two Sprite Units is determined, as shown in the figure below.
The graphics blocks (2) and (3) of the video display controller 6 can use a sprite layer, through access to a virtual frame buffer, the sprite virtual frame.
If the graphics blocks (2) and (3) of the video display controller 6 does not implement a sprite layer, it accesses the physical memory for reading frame buffer data directly.

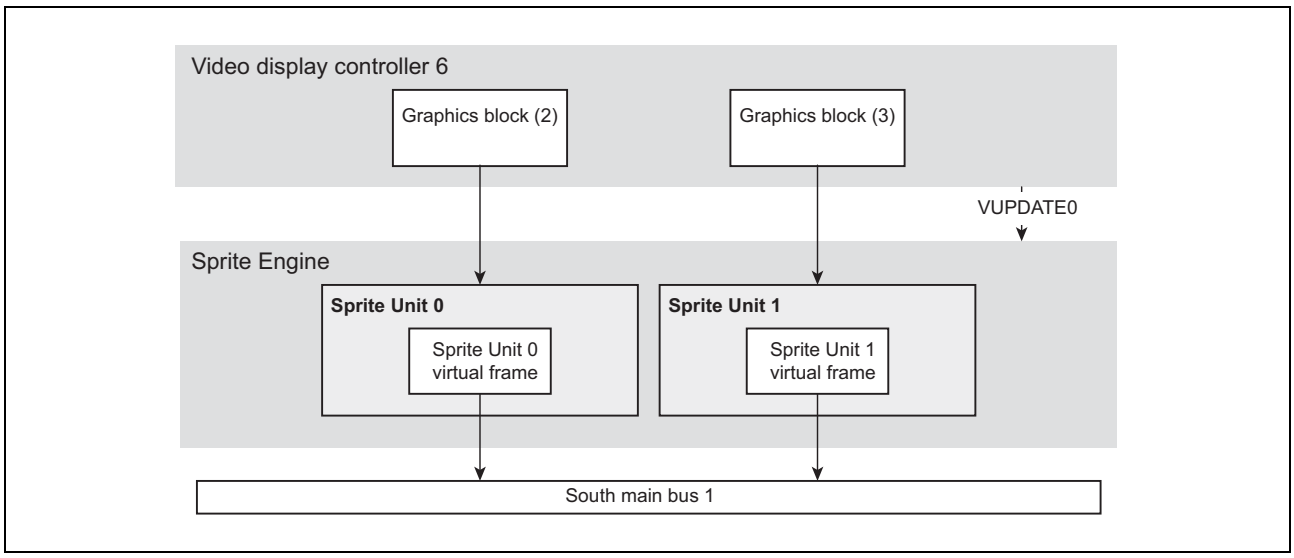


Figure 44.6 Sprite Units block diagram

44.4.1 Sprite virtual frame

The size of the sprite virtual frame is fixed to 8192 x 8192 byte.
Its address is mapped outside of any physical address space, that is used to access the physical memory directly.
The address of the first pixel of the virtual frame is the virtual frame base address

Virtual frame base address = $3000\ 0000_H$

44.4.2 Sprite activation

Each sprite m of a Sprite Unit k can be enabled or disabled separately:

- Sprite enable/disable registers
  - SPEAnSkEN.SPEAnSkENm = 1 enables sprite k
  - SPEAnSkDS.SPEAnSkDSm = 1 disables sprite k

The enable/disable status of each sprite can be read via the SPEAnSkEN register.

44.4.3 Color modes

The Sprite unit supports the following color formats.

Table 44.5 Sprite unit supported color format

Read formats of the graphics blocks (2) and (3) of the video display controller 6	Sprite unit support
αRGB8888	○
RGBα8888	○
Other	–

44.4.4 Sprite definition

A sprite m of Sprite Unit k is defined by several register settings:

Sprite X/Y position

Sprite X/Y position register SPEAnSkPSm: X/Y position on the sprite virtual frame, with respect to its virtual frame base address.

(a) SPEAnSkPSm.SPEAnSkPSXm: X position
The X position is defined as multiple of 64 bit and thus depends on the sprite color format.
The corresponding pixel position depends on the color format and the amount of pixels fitting into 64 bits.

- 32 bpp color formats: 64 bit / 32 = 2 pixel units
  - possible X positions at a multiple of 2 = 0, 2, 4, 6, 8, ...
  - SPEAnSkPSX[9:0] = X position in pixel / 2

(b) SPEAnSkPSm.SPEAnSkPSYm: Y position
The Y position is defined in pixels.

Sprite width and height

Sprite size register SPEAnSkLYm: width SPEAnSkLYWm and height SPEAnSkLYHm

(a) SPEAnSkLYm.SPEAnSkLYWm: width
The width is defined as multiple of 64 bit.

- 32 bpp color formats: 64 bit / 32 = 2 pixel units
  - possible X widths: multiple of 2 = 0, 2, 4, 6, 8, ...
  - SPEAnSkLYW[9:0] = width in pixel / 2

(b) SPEAnSkLYm.SPEAnSkLYHm: height
The height is defined in pixels.

Destination address

Destination address register SPEAnSkDAm: address of 1st pixel color data of sprite m in the memory
The destination address must be aligned to 64 bit.

Graphics blocks (2) and (3) of the video display controller 6 line offset

Since the Sprite Unit calculates the address of each new line with respect to the fixed virtual frame width of 8192 Byte, the graphics blocks (2) and (3) of the video display controller 6 must define the same line offset, when reading layer data via a Sprite Unit.
Note that the maximum number of pixels per line in the virtual frame depends on the color format:

- 32 bpp: max. 2048 pixels

The figure below shows an example with two enabled sprites.

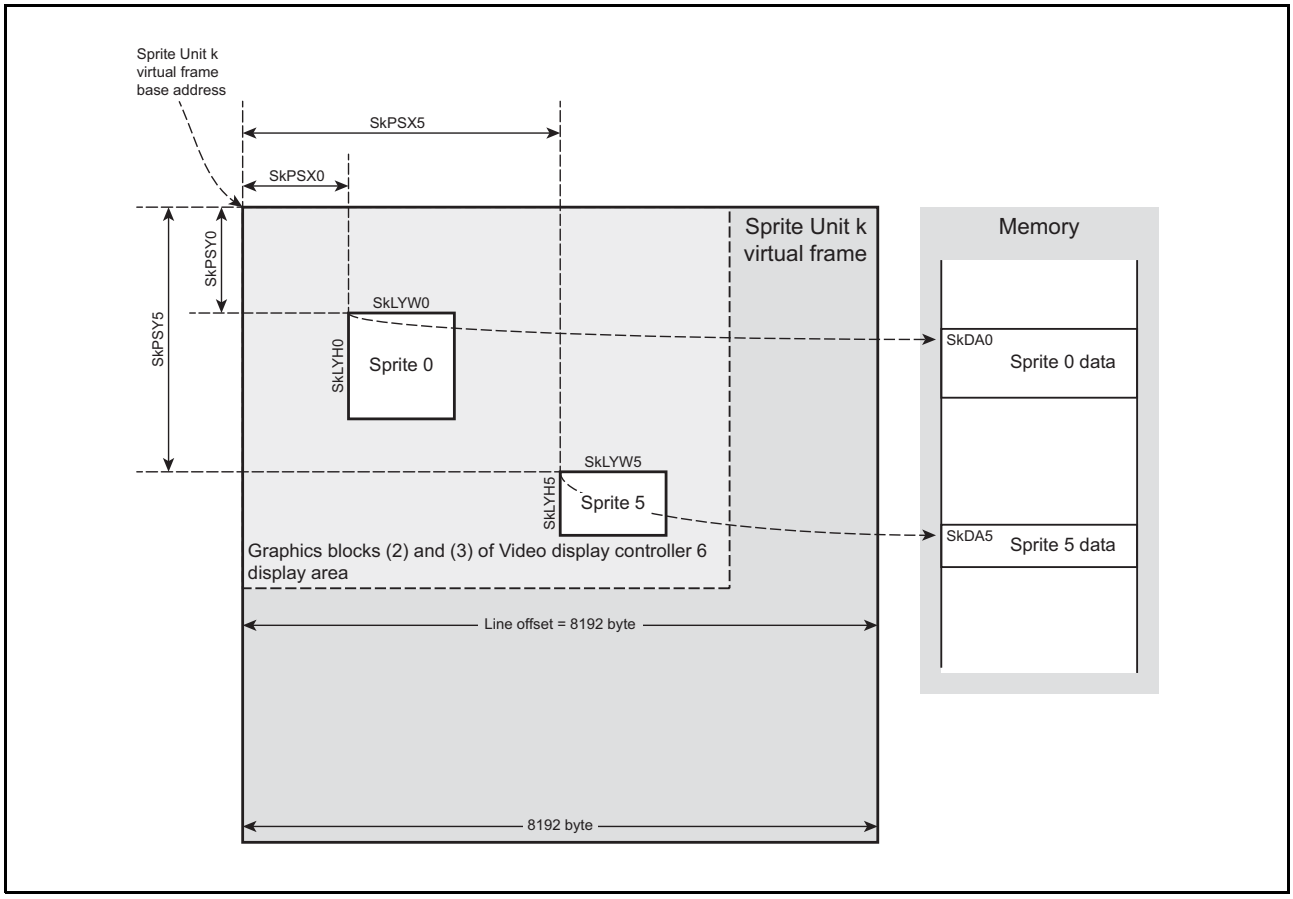


Figure 44.7 Sprite definitions

44.4.5 Overlapping sprites

Sprites can be also placed overlapping.
In such case a fixed priority order rules which sprite data is forwarded to the graphics blocks (2) and (3) of the video display controller 6:

- Sprite m = 0: highest priority
- ...
- Sprite m = 15: lowest priority

The color data of the sprite with the highest priority is forwarded to the graphics blocks (2) and (3) of the video display controller 6.

44.4.6 Sprite layer areas without active sprites

Areas of the sprite layer without active sprites return color data with value 0.

44.4.7 Sprite definition registers modification

The sprite definition registers are buffered.
While the sprite definition registers are in operation, the buffer registers can be modified without disturbing the video output.
The buffered Sprite Unit k sprite m definition registers are:

- Sprite Unit k enable registers SPEAnSkEN
- Sprite Unit k disable registers SPEAnSkDS
- Destination address registers SPEAnSkDAm
- Width/height registers SPEAnSkLYm
- X/Y position registers SPEAnSkPSm

All sprite definitions of the VUPDATE0 update group are updated simultaneously with the respective VUPDATE0 signal.
When setup of new sprite definitions is completed, the update of all sprite definition registers of Sprite Unit k must be requested via the SPEAnSkUP register:

- SPEAnSkUP.SPEAnSkUP0 = 1: initiates update of all Sprite Unit k sprite definition registers of update group VUPDATE0 with the next VUPDATE0

Caution: All sprite definition buffer registers must not be modified while SPEAnSkUP0 = 1.

44.4.8 Color Data Arrangement

The following diagram shows how color data are passed to the graphics blocks (2) and (3) of the video display controller 6 when a sprite is used.

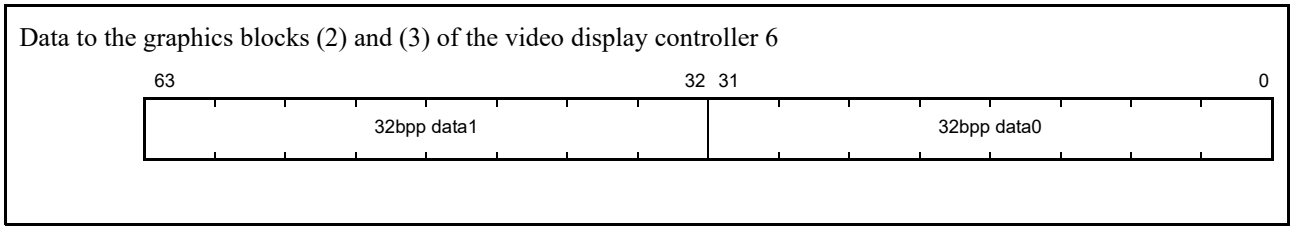


Figure 44.8 Sprite data arrangement

44.5 Sprite and RLE Units Registers

This section contains a description of all registers of the Sprite and RLE Units.
The Sprite and RLE Units are controlled and operated by the following registers:

Table 44.6 Sprite and RLE Units registers overview

Register name	Shortcut	Address
RLE Units registers		
RLE enable control register	SPEAnRLSL	<SPEAn_base> + 10_H
RLE Unit i start address register	SPEAnSTAi	<SPEAn_base> + 20_H + i x 10_H
RLE Unit i physical address register	SPEAnPHAi	<SPEAn_base> + 24_H + i x 10_H
RLE Unit i color mode register	SPEAnRCMi	<SPEAn_base> + $2C_H$ + i x 10_H
RLE register update request register	SPEAnRUP	<SPEAn_base> + 40_H
RLE configuration register	SPEAnRCFG	<SPEAn_base> + 48_H
Sprite Unit k registers		
Sprite Unit k enable register	SPEAnSkEN	<SPEAn_base> + 100_H + k x 10_H
Sprite Unit k disable register	SPEAnSkDS	<SPEAn_base> + 104_H + k x 10_H
Sprite Unit k update request register	SPEAnSkUP	<SPEAn_base> + 108_H + k x 10_H
Sprite Unit k sprite m destination address register	SPEAnSkDAm	<SPEAn_base> + 400_H + k x 400_H + m x 20_H
Sprite Unit k sprite m width/height register	SPEAnSkLYm	<SPEAn_base> + 408_H + k x 400_H + m x 20_H
Sprite Unit k sprite m X/Y position register	SPEAnSkPSm	<SPEAn_base> + $40C_H$ + k x 400_H + m x 20_H

<SPEAn_base>

The base addresses <SPEAn_base> of the SPEAn is defined in section 44.1.3, Register addresses.

44.5.1 RLE Units registers

44.5.1.1 SPEAnRLSL - RLE Units enable control register

The RLE Units can be enabled and disabled via this register.

Access: This register can be accessed in 32-bit units.
Address: <SPEAn_base> + 10_H
Initial value: 0000 0000H

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	SPEAn RBUS SEL
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 44.7 SPEAnRLSL register contents

Bit position	Bit name	Function
31 to 1	–	Reserved bits The write value should always be 0.
0	SPEAn RBUSSEL	RLE Units enable control 0: RLE Units are disabled (direct layer access) 1: RLE Units are enabled (RLE layer access)

44.5.1.2 SPEAnSTAi - RLE Unit i start address register

This register defines the virtual frame start address when the graphics block (0) of the video display controller 6 reads via the REL unit i.

Access: This register can be accessed in 32-bit units.
Address: <SPEAn_base> + 20_H + i x 10_H
Initial value: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPEAnRSTAi[31:16]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPEAnRSTAi[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 44.8 SPEAnSTAi register contents

Bit position	Bit name	Function
31 to 0	SPEAn RSTAi[31:0]	RLE Unit i start address The address must be 128-byte aligned, so the lower 7 bits SPEAnRSTAi[6:0] of the address are always 000 0000_B.

44.5.1.3 SPEAnPHAi - RLE Unit i physical address register

This register defines the address of the RLE Unit i first data in the memory.

Access: This register can be accessed in 32-bit units.
Address: <SPEAn_base> + 24_H + i x 10_H
Initial value: 0000 0000_H

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPEAnRPHAi[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPEAnRPHAi[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 44.9 SPEAnPHAi register contents

Bit position	Bit name	Function
31 to 0	SPEAn RPHAi[31:0]	RLE Unit i data memory address The address must be 64-bit aligned, so the lower 3 bits SPEAnRPHAi[2:0] of the address are always 000_B.

44.5.1.4 SPEAnRCMi - RLE Unit i color mode selection register

This register selects the color format of data in memory for RLE Unit i.

Access: This register can be accessed in 32-bit units.
Address: <SPEAn_base> + $2C_H$ + i x 10_H
Initial value: 0000 0000_H

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	SPEAnRCMi [1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 44.10 SPEAnRCMi register contents

Bit position	Bit name	Function
31 to 2	–	Reserved bits The write value should always be 0.
1, 0	SPEAn RCMi[1:0]	RLE Unit i color mode selection 00_B: setting prohibited 01_B: setting prohibited 10_B: 24 bpp 11_B: setting prohibited

44.5.1.5 SPEAnRUP - RLE registers update request register

This register controls the update of the RLE Unit's registers.

Access: This register can be accessed in 32-bit units.
Address: <SPEAn_base> + 40_H
Initial value: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	SPEAn RUP0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 44.11 SPEAnRUP register contents

Bit position	Bit name	Function
31 to 1	–	Reserved bits The write value should always be 0.
0	SPEAn RUP0	RLE Unit 0 register update request 0: no function 1: update RLE Unit 0 register with the next VUPDATE0

Note 1. The update request bit SPEAnRUPi remains set, after it has been set to 1, until the update is completed. Afterwards it returns to 0 automatically.
Note 2. All RLE definition buffer registers must not be modified while SPEAnRUPi = 1.
Refer to section 44.3.5, RLE definition registers modification for details.

44.5.1.6 SPEAnRCFG - RLE prefetch configuration register

This register controls the size and timing for the RLE data stream prefetching

Access: This register can be accessed in 32-bit units.
Address: <SPEAn_base> + 48_H
Initial value: 0000 0044_H

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	–	–	–	SPEAnRLEN[2:0]			–	SPEAnRDTH[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 44.12 SPEAnRCFG register contents

Bit position	Bit name	Function
31 to 7	–	Reserved bits The write value should always be 0.
6 to 4	SPEAn RLEN[2:0]	Burst size of the RLE unit next data prefetching. RLE unit prefetches next data every burst length x 64 bit. 0: Burst length = 1 1: Burst length = 2 2: Burst length = 4 3: Burst length = 6 4: Burst length = 8 (default) 5: Burst length = 10 6: Burst length = 12 7: Burst length = 14
3	–	Reserved bit The write value should always be 0.
2 to 0	SPEAn RDTH[2:0]	Prefetch timing. RLE unit prefetches next data when remaining amount of the data in the RLE FIFO is less than threshold x 64 bit. 0: setting prohibited 1: threshold = 2 2: threshold = 4 3: Burst length = 6 4: threshold = 8 (default) 5: threshold = 10 6: threshold = 12 7: threshold = 14

Note: The burst size SPEAnRLEN[2:0] and the prefetch timing SPEAnRDTH[2:0] must be set to values, that prevent the RLE data FIFO from overflow.
The RLE data FIFO size is 1024 bit.

44.5.2 Sprite Units registers

44.5.2.1 SPEAnSkEN - Sprite Unit k enable register

Each sprite of Sprite Unit k can be enabled by this register.
The enabled/disabled status of the sprites on Sprite Unit k can be checked by reading this register.
Disabling a sprite is done via the Sprite Unit k disable register SPEAnSkDS.

Access: This register can be accessed in 32-bit units.
Address: <SPEAn_base> + 100_H + k x 10_H
Initial value: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPEAn SkEN15	SPEAn SkEN14	SPEAn SkEN13	SPEAn SkEN12	SPEAn SkEN11	SPEAn SkEN10	SPEAn SkEN9	SPEAn SkEN8	SPEAn SkEN7	SPEAn SkEN6	SPEAn SkEN5	SPEAn SkEN4	SPEAn SkEN3	SPEAn SkEN2	SPEAn SkEN1	SPEAn SkEN0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 44.13 SPEAnSkEN register contents

Bit position	Bit name	Function
31 to 16	–	Reserved bits The write value should always be 0.
15 to 0	SPEAn SkENm	Register write: Sprite m enable 0: at write: no function 1: at write: enable sprite m Register read: Sprite m status 0: at read: sprite m is disabled 1: at read: sprite m is enabled

44.5.2.2 SPEAnSkDS - Sprite Unit k disable register

Each sprite of Sprite Unit k can be disabled by this register.
Enabling a sprite is done via the Sprite Unit k enable register SPEAnSkEN.

Access: This register can be accessed in 32-bit units.
Address: <SPEAn_base> + 104_H + k x 10_H
Initial value: Reading this register returns an undefined value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPEAn SkDS15	SPEAn SkDS14	SPEAn SkDS13	SPEAn SkDS12	SPEAn SkDS11	SPEAn SkDS10	SPEAn SkDS9	SPEAn SkDS8	SPEAn SkDS7	SPEAn SkDS6	SPEAn SkDS5	SPEAn SkDS4	SPEAn SkDS3	SPEAn SkDS2	SPEAn SkDS1	SPEAn SkDS0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 44.14 SPEAnSkDS register contents

Bit position	Bit name	Function
31 to 16	–	Reserved bits The write value should always be 0.
15 to 0	SPEAn SkDSm	Sprite m disable 0: no function 1: disable sprite m Note that reading of this register returns an undefined value. The enable/disable status of each sprite can be evaluated by reading the Sprite Unit k enable register SPEAnSkEN.

44.5.2.3 SPEAnSkUP - Sprite Unit k update request register

This register controls the update of the Sprite Unit k registers.

Access: This register can be accessed in 32-bit units.
Address: <SPEAn_base> + 108_H + k x 10_H
Initial value: 0000 0000_H

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	SPEAn SkUP0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 44.15 SPEAnSkUP register contents

Bit position	Bit name	Function
31 to 1	–	Reserved bits The write value should always be 0.
0	SPEAn SkUP0	Update request of all sprite definition registers of Sprite Unit k, which are assigned to VUPDATE0 0: no function 1: update VUPDATE0 assigned registers with next VUPDATE0

Note 1. The update request bit SPEAnSkUP0 remains set, after it has been set to 1, until the update is completed. Afterwards it returns to 0 automatically.
Note 2. All sprite definition buffer registers must not be modified while SPEAnSkUP0 = 1.
Refer to section 44.4.7, Sprite definition registers modification for details.

44.5.2.4 SPEAnSkDAm - Sprite Unit k sprite m destination address register

This register defines the address of the 1st pixel's color data in the memory of the sprite m of the Sprite Unit k.

Access: This register can be accessed in 32-bit units.
Address: <SPEAn_base> + 400_H + k x 400_H + m x 20_H
Initial value: 0000 0000_H

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPEAnSkDAm[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPEAnSkDAm[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 44.16 SPEAnSkDAm register contents

Bit position	Bit name	Function
31 to 0	SPEAn SkDAm[31:0]	Sprite Unit k sprite m color data address The address must be 64-bit aligned, so the lower 3 bits SPEAnSkDAm[2:0] of the address are always 000_B.

44.5.2.5 SPEAnSkLYm - Sprite Unit k sprite m width/height register

This register defines the width and height of the Sprite Unit k sprite m.

Access: This register can be accessed in 32-bit units.
Address: <SPEAn_base> + 408_H + k x 400_H + m x 20_H
Initial value: 0000 0000_H

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	SPEAn SkLYWm[9:0]										–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	–	SPEAn SkLYHm[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 44.17 SPEAnSkLYm register contents

Bit position	Bit name	Function
31 to 27	–	Reserved bits The write value should always be 0.
26 to 17	SPEAn SkLYWm[9:0]	Sprite Unit k sprite m width The width is defined as multiple of 64 bit and thus depends on the sprite pixel color format. • 32 bpp color formats: 64 bit / 32 = 2 pixel units - possible X widths: multiple of 2 = 0, 2, 4, 6, 8, ... - SPEAnSkLYW[9:0] = width in pixel / 2
16 to 11	–	Reserved bits The write value should always be 0.
10 to 0	SPEAn SkLYHm[10:0]	Sprite Unit k sprite m layer height in pixels

44.5.2.6 SPEAnSkPSm - Sprite Unit k sprite m X/Y position register

This register defines the position of the Sprite Unit k sprite m.

Access: This register can be accessed in 32-bit units.
Address: <SPEAn_base> + $40C_H$ + k x 400_H + m x 20_H
Initial value: $0000\ 0000_H$

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	SPEAn SkPSXm[9:0]										–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	SPEAn SkPSYm[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 44.18 SPEAnSkPSm register contents

Bit position	Bit name	Function
31 to 27	–	Reserved bits The write value should always be 0.
26 to 17	SPEAn SkPSXm[9:0]	Sprite Unit k sprite m X position The X position is defined as multiple of 64 bit and thus depends on the sprite color format. • 32 bpp color formats: 64 bit / 32 = 2 pixel units - possible X positions at a multiple of 2 = 0, 2, 4, 6, 8, ... - SPEAnSkPSX[9:0] = X position in pixel / 2
16 to 13	–	Reserved bits The write value should always be 0.
12 to 0	SPEAn SkPSYm[12:0]	Sprite Unit k sprite m Y position in pixels

45. JPEG Codec Unit

The JPEG codec unit (JCU) incorporates a JPEG codec conforming to the JPEG baseline compression and decompression standard to provide high-speed compression of image data and high-speed decoding of JPEG data.

45.1 Features

The JPEG codec unit has the following features:

- Conforms to the JPEG baseline standard within the range described in this document.
  This module does not support the following basic features:
  Scanning with two elements
  Non-interleave scanning with multiple elements
- Operational precision: Conforming to JPEG Part 2, ISO-IEC10918-2
- Image input/output system: Block interleave method
  Pixel format:
  Compression: YCbCr422 (H = 2:1:1, V = 1:1:1)
  Decompression: YCbCr444 (H = 1:1:1, V = 1:1:1), YCbCr422 (H = 2:1:1, V = 1:1:1),
  YCbCr411 (H = 4:1:1, V = 1:1:1), YCbCr420 (H = 2:1:1, V = 2:1:1)
  Output pixel format to the buffer: YCbCr422, ARGB8888, RGB565
- Four quantization tables provided
- Four Huffman tables provided (two tables for AC coefficients and two tables for DC coefficients)
- Markers supported: SOI (start of image), SOF0 (start of frame type 0), SOS (start of scan), DQT (define quantization tables), DHT (define Huffman tables), DRI (define restart interval), RSTm (restart marks), and EOI (end of image)
- Image data rate: Max. 132 Mbytes/s (at 66-MHz operation)
- The buffer size can be reduced by using the mode in which data transfer is temporarily stopped each time the specified number of lines or the specified amount of data is transferred during image data or coded data input/output.
- Processing unit: 8-byte address boundary units can be set
- Image sizes that can be processed: Sizes divisible by the minimum coded unit (MCU): 8 lines by 8 pixels in YCbCr444; 8 lines by 16 pixels in YCbCr422; 8 lines by 32 pixels in YCbCr411; 16 lines by 16 pixels in YCbCr420

Note: Compression and decompression processing of images in unsupported pixel formats or unsupported image sizes should be avoided.

Figure 45.1 shows a block diagram.

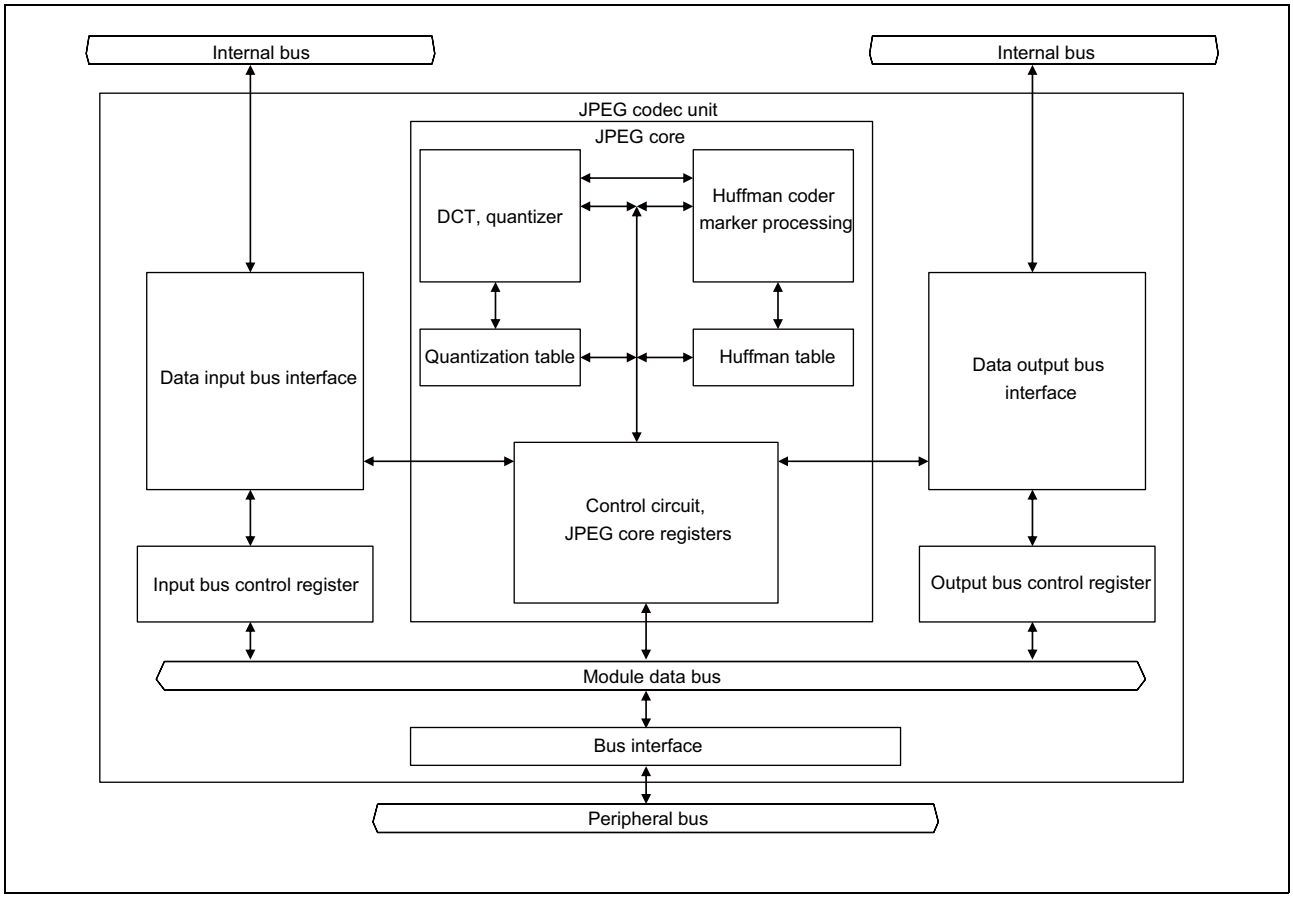


Figure 45.1 Block Diagram

45.2 Register Descriptions

Table 45.1shows the JCU registers.

Table 45.1 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
JPEG code mode register	JCMOD	R/W	H'E801 7000	8
JPEG code command register	JCCMD	R/W	H'E801 7001	8
JPEG code quantization table number register	JCQTN	R/W	H'E801 7003	8
JPEG code Huffman table number register	JCHTN	R/W	H'E801 7004	8
JPEG code DRI upper register	JCDRIU	R/W	H'E801 7005	8
JPEG code DRI lower register	JCDRID	R/W	H'E801 7006	8
JPEG code vertical size upper register	JCVSZU	R/W	H'E801 7007	8
JPEG code vertical size lower register	JCVSZD	R/W	H'E801 7008	8
JPEG code horizontal size upper register	JCHSZU	R/W	H'E801 7009	8
JPEG code horizontal size lower register	JCHSZD	R/W	H'E801 700A	8
JPEG code data count upper register	JCDTCU	R	H'E801 700B	8
JPEG code data count middle register	JCDTCM	R	H'E801 700C	8
JPEG code data count lower register	JCDTCD	R	H'E801 700D	8
JPEG interrupt enable register 0	JINTE0	R/W	H'E801 700E	8
JPEG interrupt status register 0	JINTS0	R/W	H'E801 700F	8
JPEG code decode error register	JCDERR	R/W	H'E801 7010	8
JPEG code reset register	JCRST	R	H'E801 7011	8
JPEG interface compression control register	JIFECNT	R/W	H'E801 7040	32
JPEG interface compression source address register	JIFESA	R/W	H'E801 7044	32
JPEG interface compression line offset register	JIFESOFST	R/W	H'E801 7048	32
JPEG interface compression destination address register	JIFEDA	R/W	H'E801 704C	32
JPEG interface compression source line count register	JIFESLC	R/W	H'E801 7050	32
JPEG interface compression destination register	JIFEDDC	R/W	H'E801 7054	32
JPEG interface decompression control register	JIFDCNT	R/W	H'E801 7058	32
JPEG interface decompression source address register	JIFDSA	R/W	H'E801 705C	32
JPEG interface decompression destination offset register	JIFDDOFST	R/W	H'E801 7060	32
JPEG interface decompression destination address register	JIFDDA	R/W	H'E801 7064	32
JPEG interface decompression source count register	JIFDSDC	R/W	H'E801 7068	32
JPEG interface decompression destination line count register	JIFDDLC	R/W	H'E801 706C	32
JPEG interface decompression α setting register	JIFDADT	R/W	H'E801 7070	32
JPEG interrupt enable register 1	JINTE1	R/W	H'E801 708C	32
JPEG interrupt status register 1	JINTS1	R/W	H'E801 7090	32
JPEG input image data CbCr range setting register	JIFESVSZ	R/W	H'E801 7094	32
JPEG output image data CbCr range setting register	JIFESHSZ	R/W	H'E801 7098	32
JPEG code quantization table 0 register	JCQTBL0	R/W	H'E801 7100 to H'E801 713F	8
JPEG code quantization table 1 register	JCQTBL1	R/W	H'E801 7140 to H'E801 717F	8
JPEG code quantization table 2 register	JCQTBL2	R/W	H'E801 7180 to H'E801 71BF	8
JPEG code quantization table 3 register	JCQTBL3	R/W	H'E801 71C0 to H'E801 71FF	8

Table 45.1 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
JPEG code Huffman table DC0 register	JCHTBD0	W	H'E801 7200 to H'E801 721B	8
JPEG code Huffman table AC0 register	JCHTBA0	W	H'E801 7220 to H'E801 72D1	8
JPEG code Huffman table DC1 register	JCHTBD1	W	H'E801 7300 to H'E801 731B	8
JPEG code Huffman table AC1 register	JCHTBA1	W	H'E801 7320 to H'E801 73D1	8

Note: For the settings of the JPEG code quantization table and JPEG code Huffman table, see section 45.3.1 (4), Table Setting.

45.2.1 JPEG Code Mode Register (JCMOD)

JCMOD sets the operating mode before the JCU starts operation.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	DSP	REDU[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R/W	R/W	R/W	R/W
R/W(De-compression):	R	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 4	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
3	DSP	0	R/W		Compression/Decompression Set 0: Compression process 1: Decompression process When changing between processing for compression and for decompression, be sure to reset this module in advance by setting the SRST11 bit in the software reset control register 1 (SWRSTCR1) of the power-down modes.
2 to 0	REDU[2:0]	000	R/W	R	Pixel Format [Compression] 001: YCbCr422 Other than above: Setting prohibited. [Decompression] 000: YCbCr444 001: YCbCr422 110: YCbCr411 010: YCbCr420 Other than above: Error (JCU cannot process normally.)

45.2.2 JPEG Code Command Register (JCCMD)

JCCMD sets commands. Bits of this register need not be cleared to 0 after setting a command.
Multiple commands must not be set simultaneously.

Bit:	7	6	5	4	3	2	1	0
	BRST	—	—	—	—	JEND	JRST	JSRT
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R*/W	R	R	R	R	R*/W	Invalid	R*/W
R/W(De-compression):	R*/W	R	R	R	R	R*/W	R*/W	R*/W

Note: *Values read from these bits are undefined.

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7	BRST	0	R*/W		Bus Reset Setting this bit to 1 resets the internal circuits. While the JCU is in operation (from setting the JPEG core process start command to writing the last output coded/image data), do not set this bit to 1. For the bus reset processing, see section 45.5, Bus Reset Processing.
6 to 3	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
2	JEND	0	R*/W		Interrupt Request Clear Command This bit is valid only for the interrupt sources corresponding to bits INS6, INS5, and INS3 in JINTS0. To clear an interrupt request, set this bit to 1.
1	JRST	0	Invalid	R*/W	JPEG Core Process Stop Clear Command To clear the process-stopped state caused by requests to read the image size and pixel format (enabled by the INT3 bit in JINTE0), set this bit to 1.
0	JSRT	0	R*/W		JPEG Core Process Start Command To start JPEG core processing, set this bit to 1. Do not write this bit to 1 again while the JCU is in operation.

Note: * Values read from these bits are undefined.

45.2.3 JPEG Code Quantization Table Number Register (JCQTN)

JCQTN sets the quantization table number before compression process is started.

- To use quantization table No. 0 (JCQTBL0) as the first color component, set QT1 to B'00
- To use quantization table No. 1 (JCQTBL1) as the first color component, set QT1 to B'01
- To use quantization table No. 2 (JCQTBL2) as the first color component, set QT1 to B'10
- To use quantization table No. 3 (JCQTBL3) as the first color component, set QT1 to B'11

Bit:	7	6	5	4	3	2	1	0
	—	—	QT3[1:0]		QT2[1:0]		QT1[1:0]	
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R/W	R/W	R/W	R/W	R/W	R/W
R/W(De-compression):	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7, 6	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
5, 4	QT3[1:0]	00	R/W	R	Quantization table number for the third color component
3, 2	QT2[1:0]	00	R/W	R	Quantization table number for the second color component
1, 0	QT1[1:0]	00	R/W	R	Quantization table number for the first color component

45.2.4 JPEG Code Huffman Table Number Register (JCHTN)

JCHTN sets the Huffman table number (AC/DC) before compression process is started.

- To use DC/AC Huffman table No. 0 (JCHTBD0 and JCHTBA0) as the first color component, set bits HTA1 and HTD1 to B'0
- To use DC/AC Huffman table No. 1 (JCHTBD1 and JCHTBA1) as the first color component, set bits HTA1 and HTD1 to B'1

Bit:	7	6	5	4	3	2	1	0
	—	—	HTA3	HTD3	HTA2	HTD2	HTA1	HTD1
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R/W	R/W	R/W	R/W	R/W	R/W
R/W(De-compression):	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7, 6	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
5	HTA3	0	R/W	R	Huffman table number (AC) for the third color component
4	HTD3	0	R/W	R	Huffman table number (DC) for the third color component
3	HTA2	0	R/W	R	Huffman table number (AC) for the second color component
2	HTD2	0	R/W	R	Huffman table number (DC) for the second color component
1	HTA1	0	R/W	R	Huffman table number (AC) for the first color component
0	HTD1	0	R/W	R	Huffman table number (DC) for the first color component

45.2.5 JPEG Code DRI Upper Register (JCDRIU)

JCDRIU sets the upper bytes of the minimum coded units (MCUs) preceding an RST marker.

Bit:	7	6	5	4	3	2	1	0
	DRIU[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(De-compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	DRIU[7:0]	H'00	R/W	Invalid	Upper Bytes of MCUs Preceding RST Marker When both upper and lower bytes are set to H'00, neither a DRI nor an RST marker is placed.

45.2.6 JPEG Code DRI Lower Register (JCDRID)

JCDRID sets the lower bytes of MCUs preceding an RST marker.

Bit:	7	6	5	4	3	2	1	0
	DRID[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(De-compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	DRID[7:0]	H'00	R/W	Invalid	Lower Bytes of MCUs Preceding RST Marker When both upper and lower bytes are set to H'00, neither a DRI nor an RST marker is placed.

45.2.7 JPEG Code Vertical Size Upper Register (JCVSZU)

JCVSZU sets the upper bytes of the vertical image size.

Bit:	7	6	5	4	3	2	1	0
	VSZU[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(De-compression):	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	VSZU[7:0]	H'00	R/W	R	Upper Bytes of Vertical Image Size In decompression process, a downloaded value from the JPEG coded data is set.

45.2.8 JPEG Code Vertical Size Lower Register (JCVSZD)

JCVSZD sets the lower bytes of the vertical image size.

Bit:	7	6	5	4	3	2	1	0
	VSZD[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(De-compression):	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	VSZD[7:0]	H'00	R/W	R	Lower Bytes of Vertical Image Size In decompression process, a downloaded value from the JPEG coded data is set.

45.2.9 JPEG Code Horizontal Size Upper Register (JCHSZU)

JCHSZU sets the upper bytes of the horizontal image size.

Bit:	7	6	5	4	3	2	1	0
	HSZU[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(De-compression):	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	HSZU[7:0]	H'00	R/W	R	Upper Bytes of Horizontal Image Size In decompression process, a downloaded value from the JPEG coded data is set.

45.2.10 JPEG Coded Horizontal Size Lower Register (JCHSZD)

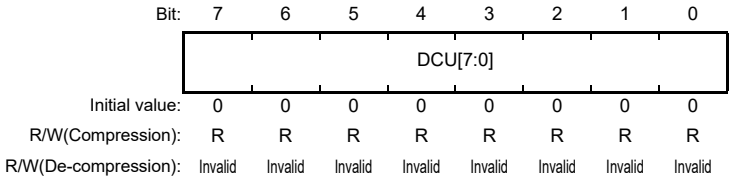
JCHSZD sets the lower bytes of the horizontal image size.

Bit:	7	6	5	4	3	2	1	0
	HSZD[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(De-compression):	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	HSZD[7:0]	H'00	R/W	R	Lower Bytes of Horizontal Image Size In decompression process, a downloaded value from the JPEG coded data is set.

45.2.11 JPEG Code Data Count Upper Register (JCDTCU)

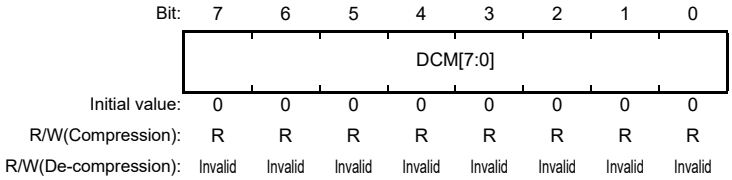
The upper bytes for the counted amount of data to be compressed are set to JCDTCU. The values of this register are reset before compression starts.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	DCU[7:0]	H'00	R	Invalid	Upper bytes of the counted amount of data to be compressed

45.2.12 JPEG Code Data Count Middle Register (JCDTCM)

The middle bytes for the counted amount of data to be compressed are set to JCDTCM. The values of this register are reset before compression starts.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	DeCompression	
7 to 0	DCM[7:0]	H'00	R	Invalid	Middle bytes of the counted amount of data to be compressed

45.2.13 JPEG Code Data Count Lower Register (JCDTCD)

The lower bytes for the counted amount of data to be compressed are set to JCDTCD. The values of this register are reset before compression starts.

Bit:	7	6	5	4	3	2	1	0
	DCD[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R
R/W(De-compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	DCD[7:0]	H'00	R	Invalid	Lower bytes of the counted amount of data to be compressed

45.2.14 JPEG Interrupt Enable Register 0 (JINTE0)

JINTE0 enables interrupts.

When any of bits INT7 to INT5 is set to B'1, the INS5 bit in JINTS0 indicates B'1 as the error status upon occurrence of the compression data error, and the ERR bit in JCDERR indicates the particular error code.

Bit:	7	6	5	4	3	2	1	0
	INT7	INT6	INT5	—	INT3	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	Invalid	Invalid	Invalid	R	Invalid	R	R	R
R/W(De-compression):	R/W	R/W	R/W	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7	INT7	0	Invalid	R/W	This bit enables an interrupt to be generated when the number of data in the restart interval of the Huffman-coding segment is not correct in decompression. When this bit is not set to enable interrupt generation, an error code is not returned.
6	INT6	0	Invalid	R/W	This bit enables an interrupt to be generated when the total number of data in the Huffman-coding segment is not correct in decompression. When this bit is not set to enable interrupt generation, an error code is not returned.
5	INT5	0	Invalid	R/W	This bit enables an interrupt to be generated when the final number of MCU data in the Huffman-coding segment is not correct in decompression. When this bit is not set to enable interrupt generation, an error code is not returned.
4	—	0	R		Reserved This bit is always read as 0. The write value should always be 0.
3	INT3	0	Invalid	R/W	This bit enables an interrupt to be generated when it has been determined that the image size and the subsampling setting of the compressed data can be read through analyzing the data.
2 to 0	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.

45.2.15 JPEG Interrupt Status Register 0 (JINTS0)

JINTS0 identifies the interrupt sources.
The interrupt sources of this register should be cleared by clearing the corresponding interrupt status bits to 0 and setting the relevant bit in JCCMD appropriately.

Bit:	7	6	5	4	3	2	1	0
	—	INS6	INS5	—	INS3	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R	R/W*	Invalid	R	Invalid	R	R	R
R/W(De-compression):	R	R/W*	R/W*	R	R/W*	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7	—	0	R		Reserved This bit is always read as 0. The write value should always be 0.
6	INS6	0	R/W∗		This bit is set to 1 when the JCU completes compression process normally.
5	INS5	0	Invalid	R/W∗	This bit is set to 1 when a compressed data error occurs.
4	—	0	R		Reserved This bit is always read as 0. The write value should always be 0.
3	INS3	0	Invalid	R/W∗	This bit is set to 1 when the image size and pixel format can be read. When an interrupt occurs, this module stops processing and the state is indicated by the JCRST register. To make the JCU resume processing, set the JPEG core process stop clear command bit (JRST) in JCCMD.
2 to 0	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.

Note: ∗ Clear this bit by writing 0 to it. Do not write 1 to this bit.

45.2.16 JPEG Code Decode Error Register (JCDERR)

JCDERR indicates the error code to identify the type of the error which has occurred in the compressed data analysis for decompression. The values of this register are reset before the JCU starts decompression.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	ERR[3:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	Invalid	Invalid	Invalid	Invalid
R/W(De-compression):	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 4	—	All 0	R		Reserved These bits are always read as 0.
3 to 0	ERR[3:0]	1010	Invalid	R/W	Error Code (See Table 45.3 and Table 45.4)

45.2.17 JPEG Code Reset Register (JCRST)

JCRST indicates a processing-stopped state caused by requests to read the image size and pixel format (enabled by the INT3 bit in JINTE0). To resume processing, set the JPEG core process stop clear command bit (JRST) in JCCMD.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	RST
Initial value:	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	Invalid
R/W(De-compression):	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 1	—	All 0	R		Reserved These bits are always read as 0.
0	RST	0	Invalid	R/W	Operating State 0: State other than below 1: Suspended state caused by interrupt sources of JINTE0

45.2.18 JPEG Interface Compression Control Register (JIFECNT)

JIFECNT controls the compression process.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	JOUTRINI	JOUTRCMD	JOUTC	—	JOUTSWAP[2:0]			—	DINRINI	DINRCMD	DINLC	—	DINSWAP[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W
R/W(De-compression):	R	Invalid	Invalid	Invalid	R	Invalid	Invalid	Invalid	R	Invalid	Invalid	Invalid	R	Invalid	Invalid	Invalid

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 15	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
14	JOUTRINI	0	R/W	Invalid	Address Initialization when Output Coded Data is Resumed This bit is only valid when the count mode for stopping the output of coded data is on. Set this bit before writing 1 to the data resume command bit. 0: The transfer address is not initialized when the output of coded data is restarted. 1: The transfer address is initialized when the output of coded data is restarted.
13	JOUTRCMD	0	R/W	Invalid	Output Coded Data Resume Command This bit is only valid when the count mode for stopping the output of coded data is on. Setting this bit to 1 resumes writing output coded data. This bit is always read as 0.
12	JOUTC	0	R/W	Invalid	Count Mode Setting for Stopping Output Coded Data 0: Count mode for stopping the output of coded data is off. 1: Count mode for stopping the output of coded data is on.
11	—	0	R		Reserved This bit is always read as 0. The write value should always be 0.
10 to 8	JOUTSWAP[2:0]	000	R/W	Invalid	Byte/Word/Longword Swap Output coded data in compression is swapped. 000: (1) (2) (3) (4) (5) (6) (7) (8) 001: (2) (1) (4) (3) (6) (5) (8) (7) [Byte swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [Word swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [Word - byte swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [Longword swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [Longword - byte swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [Longword - word swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [Longword - word - byte swap]
7	—	0	R		Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
6	DINRINI	0	R/W	Invalid	Address Initialization when Resuming Input of Image Data Lines This bit is only valid when the count mode for stopping the input of image data lines is on. Set this bit before writing 1 to the data-line resume command bit. 0: The transfer address is not initialized when the input of image data lines is restarted. 1: The transfer address is initialized when the input of image data lines is restarted.
5	DINRCMD	0	R/W	Invalid	Input Image Data Lines Resume Command This bit is valid only when the count mode for stopping the input of image data lines is on. Setting this bit to 1 resumes reading input image data. This bit is always read as 0.
4	DINLC	0	R/W	Invalid	Count Mode Setting for Stopping Input Image Data Lines 0: Count mode for stopping the input of image data lines is off. 1: Count mode for stopping the input of image data lines is on.
3	—	0	R		Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	DINSWAP [2:0]	000	R/W	Invalid	Byte/Word Swap Input image data in compression is swapped. 000: (1) (2) (3) (4) (5) (6) (7) (8) 001: (2) (1) (4) (3) (6) (5) (8) (7) [Byte swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [Word swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [Word - byte swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [Longword swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [Longword - byte swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [Longword - word swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [Longword - word - byte swap]

45.2.19 JPEG Interface Compression Source Address Register (JIFESA)

JIFESA sets the source address of the input image data. This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ESA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(De-compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ESA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(De-compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 3	ESA[31:3]	H'0000	R/W	Invalid	Input Image Data Source Address (in 8-byte units)
2 to 0	ESA[2:0]	0000	R		The lower three bits should be set to 0.

45.2.20 JPEG Interface Compression Line Offset Register (JIFESOFST)

JIFESOFST sets the line offset of the input image data (refer to section 45.3.4, Storing Image Data). This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ESMW[14:0]														
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(De-compression):	R	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 15	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
14 to 3	ESMW[14:3]	H'0000	R/W	Invalid	Input Image Data Lines Offset (in 8-byte units)
2 to 0	ESMW[2:0]		R		The lower three bits should be set to 0.

45.2.21 JPEG Interface Compression Destination Address Register (JIFEDA)

JIFEDA sets the destination address of the output coded data. This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EDA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(De-compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EDA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(De-compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 3	EDA[31:3]	H'0000	R/W	Invalid	Output Coded Data Destination Address (in 8-byte units)
2 to 0	EDA[2:0]	0000	R		The lower three bits should be set to 0.

45.2.22 JPEG Interface Compression Source Line Count Register (JIFESLC)

JIFESLC sets the number of input image data lines when the count mode for stopping the input of image data lines is on (the DINLC bit in JIFECNT is set to 1). This register should be set in 8-line units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LINES[15:0]															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(Compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(De-compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 16	—	H'FFF8	R		Reserved Values read from these bits are undefined. The write value should always be 0.
15 to 3	LINES[15:3]	H'FFF8	R/W	Invalid	Number of Input Image Data Lines to be Read (in 8-line units)
2 to 0	LINES[2:0]		R		The lower three bits should be set to 0.

45.2.23 JPEG Interface Compression Destination Count Register (JIFEDDC)

JIFEDDC sets the amount of output coded data when the count mode for stopping the output of coded data is on (the JOUTC bit in JIFECNT is set to 1). This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	JDATAS[15:0]															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(Compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(De-compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 16	—	H'FFF8	R	Invalid	Reserved Values read from these bits are undefined. The write value should always be 0.
15 to 3	JDATAS[15:3]	H'FFF8	R/W	Invalid	Amount of Output Coded Data to be Written (in 8-byte units)
2 to 0	JDATAS[2:0]		R		The lower three bits should be set to 0.

45.2.24 JPEG Interface Decompression Control Register (JIFDCNT)

JIFDCNT controls the decompression process.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	VINTER[1:0]		HINTER[1:0]		OPF[1:0]		—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	JINR INI	JINR CMD	JINC	—	JINSWAP[2:0]			—	DOUT RINI	DOUT RCMD	DOUT LC	—	DOUTSWAP[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	Invalid	Invalid	Invalid	R	Invalid	Invalid	Invalid	R	Invalid	Invalid	Invalid	R	Invalid	Invalid	Invalid
R/W(De-compression):	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31, 30	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
29, 28	VINTER[1:0]	00	Invalid	R/W	Vertical Subsampling Subsamples vertical output image data. 00: No subsampling 01: Subsamples output data into 1/2. 10: Subsamples output data into 1/4. 11: Subsamples output data into 1/8.
27, 26	HINTER[1:0]	00	Invalid	R/W	Horizontal Subsampling Subsamples horizontal output image data. 00: No subsampling 01: Subsamples output data into 1/2. 10: Subsamples output data into 1/4. 11: Subsamples output data into 1/8.
25, 24	OPF[1:0]	00	Invalid	R/W	Specifies output image data pixel format. 00: YCbCr422 01: ARGB8888 10: RGB565 11: Setting prohibited
23 to 15	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
14	JINRINI	0	Invalid	R/W	Address Initialization when Input Coded Data is Resumed This bit is only valid when the count mode for stopping the input of coded data is on. Set this bit before writing 1 to the data resume command bit. 0: The transfer address is not initialized when the input of coded data is restarted. 1: The transfer address is initialized when the input of coded data is restarted.
13	JINRCMD	0	Invalid	R/W	Input Coded Data Resume Command This bit is valid only when the count mode for stopping the input of coded data is on. Setting this bit to 1 resumes reading input coded data. This bit is always read as 0.
12	JINC	0	Invalid	R/W	Count Mode Setting for Stopping Input Coded Data 0: Count mode for stopping the input of coded data is off. 1: Count mode for stopping the input of coded data is on.
11	—	0	R		Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
10 to 8	JINSWAP [2:0]	000	Invalid	R/W	Byte/Word/Longword Swap Input coded data in decompression is swapped. 000: (1) (2) (3) (4) (5) (6) (7) (8) 001: (2) (1) (4) (3) (6) (5) (8) (7) [Byte swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [Word swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [Word - byte swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [Longword swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [Longword - byte swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [Longword - word swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [Longword - word - byte swap]
7	—	0	R		Reserved This bit is always read as 0. The write value should always be 0.
6	DOUTRINI	0	Invalid	R/W	Address Initialization when Resuming Output of Image Data Lines This bit is only valid when the count mode for stopping the output of image data lines is on. Set this bit before writing 1 to the data-line resume command bit. 0: The transfer address is not initialized when the output of lines of image data is restarted. 1: The transfer address is initialized when the output of lines of image data is restarted.
5	DOUTRCMD	0	Invalid	R/W	Output Image Data Lines Resume Command This bit is valid only when the count mode for stopping the output of image data lines is on. Setting this bit to 1 resumes writing image data. This bit is always read as 0.
4	DOUTLC	0	Invalid	R/W	Count Mode for Stopping Output Image Data Lines 0: Count mode for stopping the output of image data lines is off. 1: Count mode for stopping the output of image data lines is on.
3	—	0	R		Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	DOUTSWAP [2:0]	000	Invalid	R/W	Byte/Word Swap Output image data in decompression is swapped. 000: (1) (2) (3) (4) (5) (6) (7) (8) 001: (2) (1) (4) (3) (6) (5) (8) (7) [Byte swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [Word swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [Word - byte swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [Longword swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [Longword - byte swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [Longword - word swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [Longword - word - byte swap]

45.2.25 JPEG Interface Decompression Source Address Register (JIFDSA)

JIFDSA sets the source address of the input coded data. This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
R/W(De-compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
R/W(De-compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W: Compression	R/W: De-compression	Description
31 to 3	DSA[31:3]	H'0000	Invalid	R/W	Input Coded Data Source Address (in 8-byte units)
2 to 0	DSA[2:0]	0000		R	The lower three bits should be set to 0.

45.2.26 JPEG Interface Decompression Line Offset Register (JIFDDOFST)

JIFDDOFST sets the line offset of the output image data to be transferred to the external buffer (refer to section 45.3.4, Storing Image Data). This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DDMW[14:0]														
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
R/W(De-compression):	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W: Compression	R/W: De-compression	Description
31 to 15	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
14 to 3	DDMW[14:3]	H'0000	Invalid	R/W	Output Image Data Lines Offset (in 8-byte units)
2 to 0	DDMW[2:0]			R	The lower three bits should be set to 0.

45.2.27 JPEG Interface Decompression Destination Address Register (JIFDDA)

JIFDDA sets the destination address of the output image data. This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DDA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
R/W(De-compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
R/W(De-compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W Compression	R/W De-compression	Description
31 to 3	DDA[31:3]	H'0000	Invalid	R/W	Output Image Data Destination Address (in 8-byte units)
2 to 0	DDA[2:0]	0000		R	The lower three bits should be set to 0.

45.2.28 JPEG Interface Decompression Source Data Count Register (JIFDSDC)

JIFDSDC sets the amount of input coded data when the count mode for stopping the input of coded data is on (the JINC bit in JIFDCNT is set to 1). This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	JDATAS[15:0]															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(Compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
R/W(De-compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W Compression	R/W De-compression	Description
31 to 16	—	H'FFF8	R		Reserved Values read from these bits are undefined. The write value should always be 0.
15 to 3	JDATAS[15:3]	H'FFF8	Invalid	R/W	Amount of Input Coded Data to be Read (in 8-byte units).
2 to 0	JDATAS[2:0]			R	The lower three bits should be set to 0.

45.2.29 JPEG Interface Decompression Destination Line Count Register (JIFDDLC)

JIFDDLC sets the number of lines of output image data when the count mode for stopping the output of image data lines is on (the DOUTLC bit in JIFECNT is set to 1). This register is used to set the number of lines of output image data in MCU units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	LINES[14:0]														
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(Compression):	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
R/W(De-compression):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 16	—	H'FFF8	R		Reserved Values read from these bits are undefined. The write value should always be 0.
15 to 3 2 to 0	LINES[15:3] LINES[2:0]	H'FFF8	Invalid	R/W R	Specify the number of lines of output image data to be written. The setting is in MCU units. When data are to be output in YCbCr444, YCbCr422 or YCbCr411 format, the number of lines of output image data is <this setting> x 1. When data are to be output in YCbCr420 format, the number of lines of output image data is <this setting> x 2. The lower three bits should be set to 0.

45.2.30 JPEG Interface Decompression α Set Register (JIFDADT)

JIFDADT is used to set the α value when output is in ARGB8888 format.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ALPHA[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
R/W(De-compression):	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 8	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	ALPHA[7:0]	H'00	Invalid	R/W	Setting of the α value for output in ARGB8888 format.

45.2.31 JPEG Interrupt Enable Register 1 (JINTE1)

JINTE1 enables interrupts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CBT EN	DINL EN	JOUT EN	—	DBT EN	JIN EN	DOUT LEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	Invalid	Invalid	Invalid
R/W(De-compression):	R	R	R	R	R	R	R	R	R	Invalid	Invalid	Invalid	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 7	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
6	CBTEN	0	R/W	Invalid	Enables or disables a data transfer processing interrupt request (JDTI) when the CBTF bit in JINTS1 is set to 1. 0: Disables an interrupt request. 1: Enables an interrupt request.
5	DINLEN	0	R/W	Invalid	Enables or disables a data transfer processing interrupt request (JDTI) when the DINLF bit in JINTS1 is set to 1. 0: Disables an interrupt request. 1: Enables an interrupt request.
4	JOUTEN	0	R/W	Invalid	Enables or disables a data transfer processing interrupt request (JDTI) when the JOUTF bit in JINTS1 is set to 1. 0: Disables an interrupt request. 1: Enables an interrupt request.
3	—	0	R		Reserved This bit is always read as 0. The write value should always be 0.
2	DBTEN	0	Invalid	R/W	Enables or disables a data transfer processing interrupt request (JDTI) when the DBTF bit in JINTS1 is set to 1. 0: Disables an interrupt request. 1: Enables an interrupt request.
1	JINEN	0	Invalid	R/W	Enables or disables a data transfer processing interrupt request (JDTI) when the JINF bit in JINTS1 is set to 1. 0: Disables an interrupt request. 1: Enables an interrupt request.
0	DOUTLEN	0	Invalid	R/W	Enables or disables a data transfer processing interrupt request (JDTI) when the DOUTLF bit in JINTS1 is set to 1. 0: Disables an interrupt request. 1: Enables an interrupt request.

45.2.32 JPEG Interrupt Status Register 1 (JINTS1)

JINTS1 indicates the interrupt sources.

The interrupt sources of this register should be cleared by writing 0 to this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CBTF	DINLF	JOUTF	—	DBTF	JINF	DOUTLF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R/W*	R/W*	R/W*	R	Invalid	Invalid	Invalid
R/W(De-compression):	R	R	R	R	R	R	R	R	R	Invalid	Invalid	Invalid	R	R/W*	R/W*	R/W*

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 7	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
6	CBTF	0	R/W*	Invalid	This bit is set to 1 when the last output coded data is written in compression.
5	DINLF	0	R/W*	Invalid	This bit is set to 1 when the number of input image data lines indicated by JIFESLC is read in compression. This bit is valid only when the DINLC bit in JIFECNT is set to 1.
4	JOUTF	0	R/W*	Invalid	This bit is set to 1 when the amount of output coded data indicated by JIFEDDC is written in compression. This bit is valid only when the JOUTC bit in JIFECNT is set to 1.
3	—	0	R		Reserved This bit is always read as 0. The write value should always be 0.
2	DBTF	0	Invalid	R/W*	This bit is set to 1 when the last output image data is written in decompression.
1	JINF	0	Invalid	R/W*	This bit is set to 1 when the amount of input coded data indicated by JIFDSDC is read in decompression. This bit is valid only when the JINC bit in JIFDCNT is set to 1.
0	DOUTLF	0	Invalid	R/W*	In decompression, this bit is set to 1 when the number of lines of output image data indicated by JIFDDLC have been written. This bit is only valid when the DOUTLC bit in JIFDCNT is set to 1.

Note: * When the bit is read as 1, write 0 to clear it.
When the bit is read as 0, write 1 to it.

45.2.33 JPEG Input Image Data CbCr Range Setting Register (JIFESVSZ)

JIFESVSZ sets the CbCr range of input image data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DINYCHG	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	Invalid	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 16	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
15	DINYCHG	0	R/W	Invalid	Input Image Data CbCr Range Setting 0: Range from -128 to 127 1: Range from 0 to 255
14 to 0	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.

45.2.34 JPEG Output Image Data CbCr Range Setting Register (JIFESHSZ)

JIFESHSZ sets the CbCr range of output image data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DOUTY CHG	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(Compression):	Invalid	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(De-compression):	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 16	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.
15	DOUTYCHG	0	Invalid	R/W	Output Image Data CbCr Range Setting 0: Range from -128 to 127 1: Range from 0 to 255
14 to 0	—	All 0	R		Reserved These bits are always read as 0. The write value should always be 0.

45.3 Operation

45.3.1 Compression

(1) Overview of Processing

The compression process flows are described below.

1. The JPEG core is activated.
   A marker is output. (After a marker is output, image data can be input.)
   Approximately 30,000 cycles (necessary for making SOI to SOS markers)
2. Image data is transferred in MCUs from the external buffer to the JCU.
   If the count mode for stopping the input of image data lines is on, reading is stopped each time the number of lines set in JIFESLC is read. Reading is resumed by setting the DINRCMD bit in JIFECNT to 1.
   When the DINRINI bit in JIFECNT is zero, the addresses for reading on resumption are continued from the addresses in the previous round of transfer.
   When the DINRINI bit is one, the address set in JIFESA is used on resumption.
   Reading is also stopped when one frame of image data is completely transferred.
   If the count mode for stopping the input of image data lines is off, reading is continued until one frame of image data is completely transferred.
3. Image data is input to the JPEG core.
   The input data is processed in MCUs at any time in the JPEG core.
4. Coded data is transferred from the JCU to the external buffer.
   When the count mode for stopping the output of coded data is on, writing is stopped each time the amount of coded data set in JIFEDDC is written. Writing is resumed by setting the JOUTRCMD bit in JIFECNT to 1.
   When the JOUTRINI bit in JIFECNT is zero, the addresses for writing on resumption are continued from the addresses in the previous round of transfer.
   When the JOUTRINI bit is one, the address set in JIFEDA is used on resumption.
   Writing is also stopped when one frame of coded data is completely transferred.
   If the count mode for stopping the output of coded data is off, writing is continued until one frame of coded data is completely transferred.
5. Compression is completed after one frame of data is processed completely.

(2) Flowchart (Compression)

(a) Initial Settings

After completing the JPEG core settings and input/output buffer settings and transferring image data to the external buffer, activate this module by setting the JSRT bit in JCCMD to 1. After the JCU has been activated, the JPEG markers (SOI to SOS) are generated and output. It takes approximately 30,000 cycles to generate the markers.

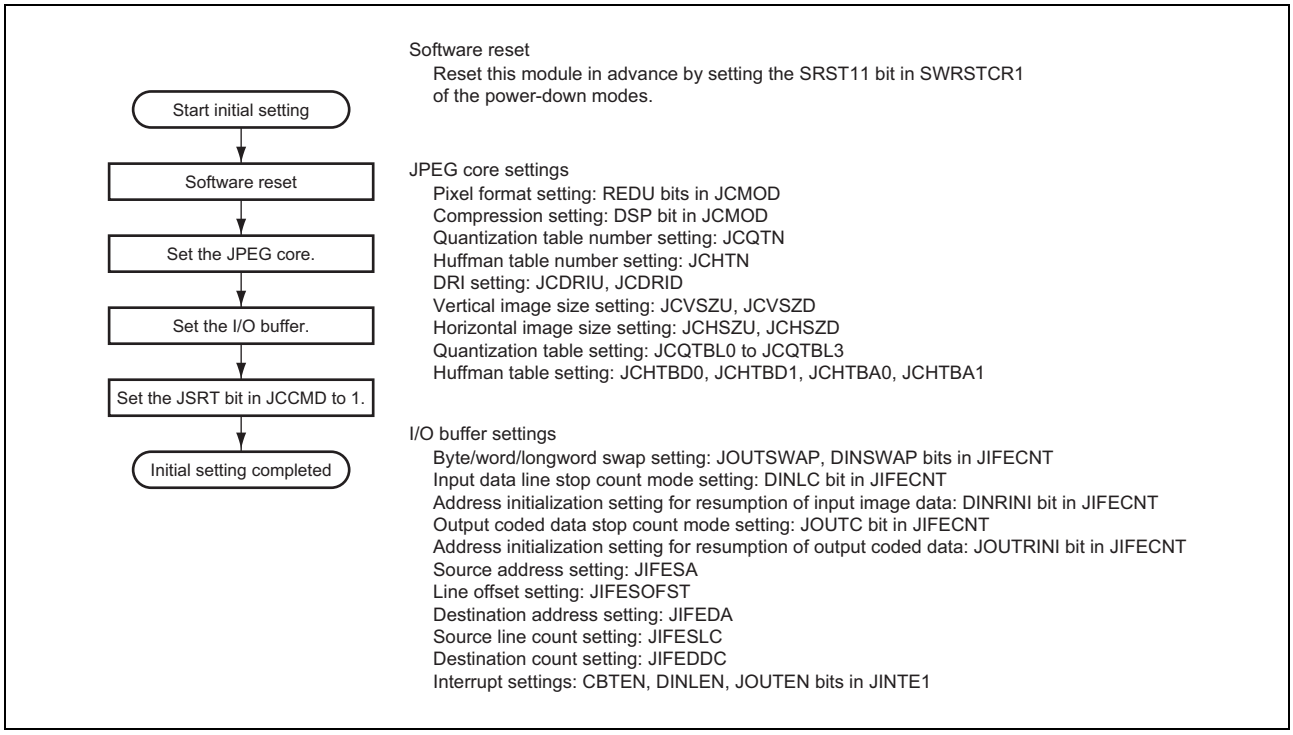


Figure 45.2 Compression Initial Setting Flow

(b) Compression Process

The compression process flows are described below.

- When JPEG compression process has been completed, the INS6 bit in JINTS0 is set to 1. However, the JCU continues processing since the coded data remains to be transferred. The CBTF bit in JINTS1 is set to 1 when the last coded data is transferred. The interrupt source is cleared by writing 0 to the INS6 bit. However, the interrupt request asserted by this interrupt source cannot be cleared by writing 0 to the INS6 bit. Set an interrupt request clear command (by setting the JEND bit in JCCMD to 1) to clear the interrupt request.
- When the JCU has completed compression and all coded data has been transferred, the CBTF flag in JINTS1 is set to 1. When the CBTEN bit in JINTE1 is 1 here, an interrupt is generated. The interrupt source is cleared by writing 0 to the CBTF flag.
- If the count mode for stopping image data lines is on, when the specified number of image data lines set in JIFESLC has been read, the DINLF flag in JINTS1 is set to 1, and reading is stopped. When the DINLEN bit in JINTE1 is 1 here, an interrupt is generated. An interrupt source is cleared by writing 0 to the DINLEN bit. Setting the DINRCMD bit in JIFECNT to 1 resumes reading.
  When the DINRINI bit in JIFECNT is zero, the addresses for reading on resumption are continued from the addresses in the previous round of transfer.
  When the DINRINI bit is one, the address set in JIFESA is used on resumption.

- If the count mode for stopping the output of coded data is on, when the specified amount of coded data set in JIFEDDC has been written, the JOUTF flag in JINTS1 is set to 1, and writing is stopped. When the JOUTEN bit in JINTE1 is 1 here, an interrupt is generated. An interrupt source is cleared by writing 0 to the JOUTF bit. Setting the JOUTRCMD bit in JIFECNT to 1 resumes writing.
  When the JOUTRINI bit in JIFECNT is zero, the addresses for writing on resumption are continued from the addresses in the previous round of transfer.
  When the JOUTRINI bit is one, the address set in JIFEDA is used on resumption.

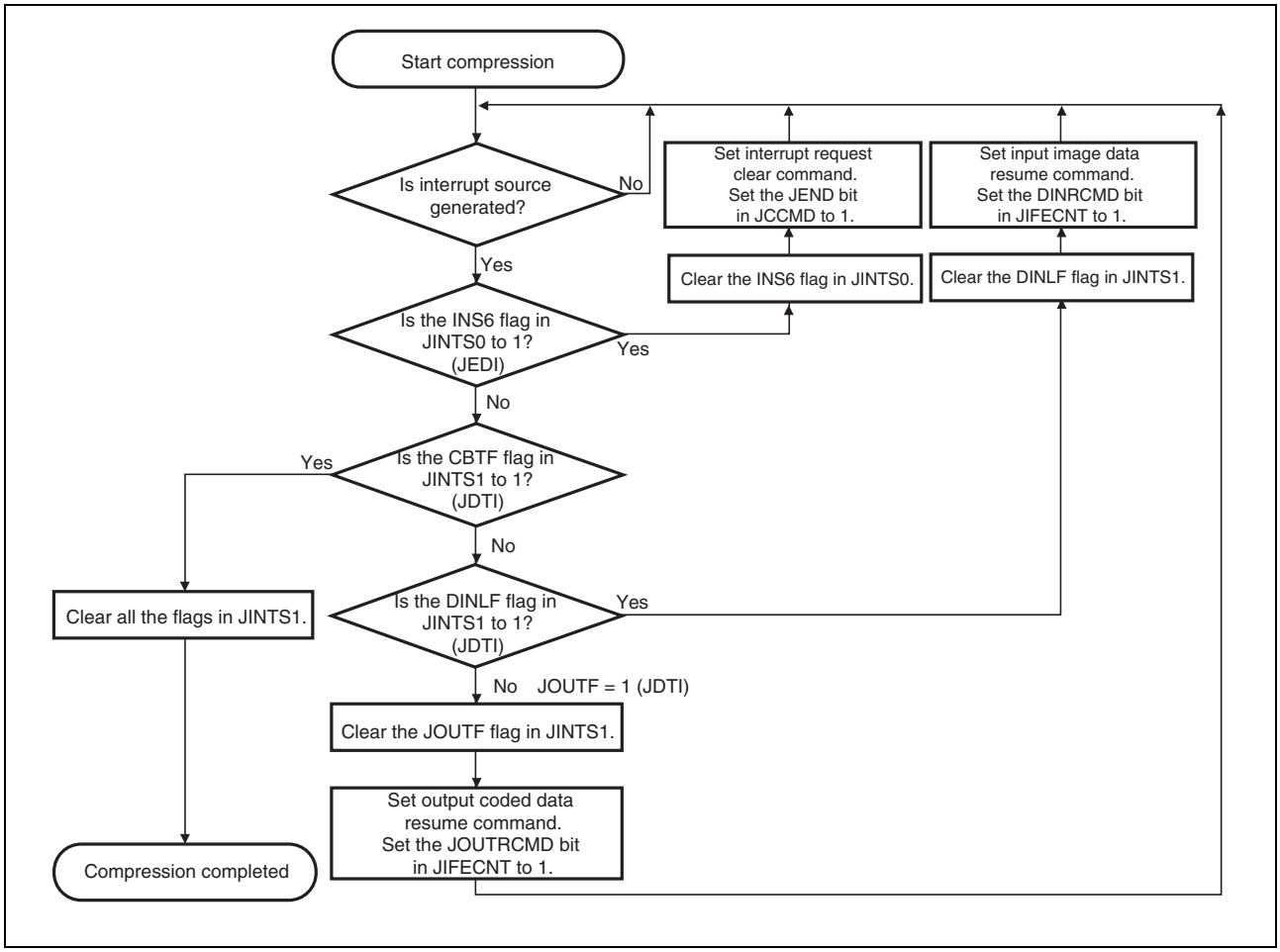


Figure 45.3 Compression Process Flow

(3) JPEG Coded Data Format

Figure 45.4 shows the data output stream in compression. The amount of coded data from SOI to EOI is indicated by JCDTCU, JCDTCM, and JCDTCD. When both JCDRIU and JCDRID are set to H'0000 0000, the following markers are not output.

- DRI marker
- RST marker (in compressed image data)

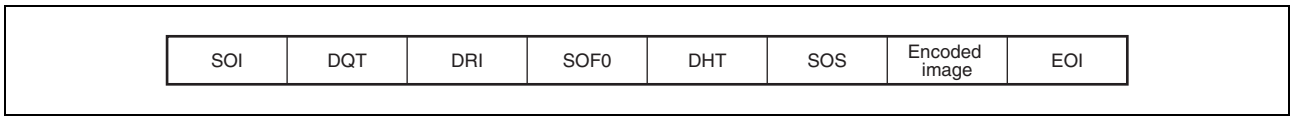


Figure 45.4 JPEG Coded Data Format

DQT: Not output for unused table.
DHT: Output in order DC0, AC0, DC1, and AC1. Not output for unused table.
SOF0: Component identifiers are C1 = first color component, C2 = second color component, and C3 = third color component.
SOS: Scan component selectors are CS1 = first color component, CS2 = second color component, and CS3 = third color component.

Header Volume (Reference):

- SOI: 2 bytes (FFD8)
- DQT: 134 bytes when two quantization tables are used, 199 bytes when three quantization tables are used (±65 bytes/table increase or decrease)
- DRI: 6 bytes
- SOF0: 19 bytes (4:2:2)
- DHT: 420 bytes (two tables are used)
- SOS: 14 bytes (4:2:2)
- EOI: 2 bytes (FFD9)

(4) Table Setting

(a) Quantization Table Specification

The order of addresses shown in 8 × 8 blocks corresponds to that of the register addresses. Do not access this table while the JCU is in processing.

Table 45.2 Quantization Table

00	01	02	03	04	05	06	07
08	09	0A	0B	0C	0D	0E	0F
10	11	12	13	14	15	16	17
18	19	1A	1B	1C	1D	1E	1F
20	21	22	23	24	25	26	27
28	29	2A	2B	2C	2D	2E	2F
30	31	32	33	34	35	36	37
38	39	3A	3B	3C	3D	3E	3F

JCQTBL0 (H'E801 7100) = H'00
JCQTBL0 (H'E801 7101) = H'01
JCQTBL0 (H'E801 7102) = H'02
JCQTBL0 (H'E801 7103) = H'03
:
JCQTBL0 (H'E801 713F) = H'3F

(b) Huffman Table Specification

Examples of the Huffman table specification given in the ITU-T T81 Annex K.3.3 recommended by JPEG are shown below. In compression, the following settings must be specified for all the codes so that Huffman codes can be generated for all the group numbers.

- DC Huffman table:The number of codes for each code length is 12.
  The group numbers in order of frequency of occurrence are 12.
- AC Huffman table:The number of codes for each code length is 162.
  The zero run length/the group numbers in order of frequency of occurrence are 162.

Do not access the following tables while the JCU is in processing. In particular, read access is prohibited.

- Table K.3/T81
  JCHTBD0 (H'E801 7200) = H'00
  JCHTBD0 (H'E801 7201) = H'01
  JCHTBD0 (H'E801 7202) = H'05
  JCHTBD0 (H'E801 7203) = H'01
  :
  JCHTBD0 (H'E801 721B) = H'0B
- Table K.4/T81
  JCHTBD1 (H'E801 7300) = H'00
  JCHTBD1 (H'E801 7301) = H'03
  JCHTBD1 (H'E801 7302) = H'01
  JCHTBD1 (H'E801 7303) = H'01
  :
  JCHTBD1 (H'E801 731B) = H'0B

- Table K.5/T81
  JCHTBA0 (H'E801 7220) = H'00
  JCHTBA0 (H'E801 7221) = H'02
  JCHTBA0 (H'E801 7222) = H'01
  JCHTBA0 (H'E801 7223) = H'03
  :
  JCHTBA0 (H'E801 72D1) = H'FA

- Table K.6/T81
  JCHTBA1 (H'E801 7320) = H'00
  JCHTBA1 (H'E801 7321) = H'02
  JCHTBA1 (H'E801 7322) = H'01
  JCHTBA1 (H'E801 7323) = H'02
  :
  JCHTBA1 (H'E801 73D1) = H'FA

(5) Input Pixel Format

Image data in the YCbCr422 format can be input to this module. Allocation of data in the YCbCr422 format can be changed by the DINSWAP bits in JIFECNT as shown below.

- When the DINSWAP bits = 000

b63 b56	b55 b48	b47 b40	b39 b32	b31 b24	b23 b16	b15 b8	b7 b0
Y0 8 bits	Cb0 8 bits	Y1 8 bits	Cr0 8 bits	Y2 8 bits	Cb1 8 bits	Y3 8 bits	Cr1 8 bits

- When the DINSWAP bits = 001

b63 b56	b55 b48	b47 b40	b39 b32	b31 b24	b23 b16	b15 b8	b7 b0
Cb0 8 bits	Y0 8 bits	Cr0 8 bits	Y1 8 bits	Cb1 8 bits	Y2 8 bits	Cr1 8 bits	Y3 8 bits

- When the DINSWAP bits = 010

b63 b56	b55 b48	b47 b40	b39 b32	b31 b24	b23 b16	b15 b8	b7 b0
Y1 8 bits	Cr0 8 bits	Y0 8 bits	Cb0 8 bits	Y3 8 bits	Cr1 8 bits	Y2 8 bits	Cb1 8 bits

- When the DINSWAP bits = 100

b63 b56	b55 b48	b47 b40	b39 b32	b31 b24	b23 b16	b15 b8	b7 b0
Y2 8 bits	Cb1 8 bits	Y3 8 bits	Cr1 8 bits	Y0 8 bits	Cb0 8 bits	Y1 8 bits	Cr0 8 bits

- When the DINSWAP bits = 101

b63 b56	b55 b48	b47 b40	b39 b32	b31 b24	b23 b16	b15 b8	b7 b0
Cb1 8 bits	Y2 8 bits	Cr1 8 bits	Y3 8 bits	Cb0 8 bits	Y0 8 bits	Cr0 8 bits	Y1 8 bits

- When the DINSWAP bits = 110

b63 b56	b55 b48	b47 b40	b39 b32	b31 b24	b23 b16	b15 b8	b7 b0
Y3 8 bits	Cr1 8 bits	Y2 8 bits	Cb1 8 bits	Y1 8 bits	Cr0 8 bits	Y0 8 bits	Cb0 8 bits

- When the DINSWAP bits = 111

b63 b56	b55 b48	b47 b40	b39 b32	b31 b24	b23 b16	b15 b8	b7 b0
Cr1 8 bits	Y3 8 bits	Cb1 8 bits	Y2 8 bits	Cr0 8 bits	Y1 8 bits	Cb0 8 bits	Y0 8 bits

(6) Output Coded Data

In the case of compression, coded data are output. This module handles the output of coded data in 16-bit units. For this reason, if the coded data have an odd code length (are fractional), the final code for output will be H'D9FF.
The JOUTSWAP bits in JIFECNT can be used to alter the arrangement of coded data in the output.

45.3.2 Decompression

(1) Overview of Processing

The decompression process flows are described below.

1. The JPEG core is activated.
2. Coded data is transferred from the external buffer to the JCU.
   If the count mode for stopping the input of coded data is on, reading is stopped each time the amount of coded data set in JIFDSLC is read. Reading is resumed by setting the JINRCMD bit in JIFDCNT to 1. When the JINRINI bit in JIFDCNT is zero, the addresses for reading on resumption are continued from the addresses in the previous round of transfer.
   When the JINRINI bit is one, the address set in JIFDSA is used on resumption. Reading is stopped when the end of the coded data is detected.
   If the count mode for stopping the input of coded data is off, reading is continued until the end of code is detected. With this module, more coded data may be read than the coded data size since coded data reading is continued until the end of code is detected.
3. Coded data is input to the JPEG core.
   The input data is processed in MCUs at any time in the JPEG core.
4. Image data is transferred in MCUs from the JCU to the external buffer.
   When the count mode for stopping the output of image data lines is on, writing is stopped each time the number of image data lines set in JIFDDLC is written. Writing is resumed by setting the DOUTRCMD bit in JIFECNT to 1. When the DOUTRINI bit in JIFDCNT is zero, the addresses for writing on resumption are continued from the addresses in the previous round of transfer.
   When the DOUTRINI bit is one, the address set in JIFDDA is used on resumption. Writing is stopped when one frame of image data is completely transferred.
   If the count mode for stopping the output of image data lines is off, writing is continued until one frame of image data is completely transferred.
5. Decompression is completed after one frame of data is processed completely.

(a) Initial Settings

- When the INT3 bit in JINTE0 is set to 0:
  After completing the JPEG core settings and input/output buffer settings and transferring coded data to the external buffer, activate this module by setting the JSRT bit in JCCMD to 1.
- When the INT3 bit in JINTE0 is set to 1:
  After completing the JPEG core settings and input buffer settings and transferring coded data to the external buffer, activate this module by setting the JSRT bit in JCCMD to 1.
  When the image size and pixel format become readable after the coded data has been decompressed, the INS3 bit in JINTS0 is set. At this time, decompression is temporarily stopped.
  After the image size and pixel format have been read, set the output buffer.
  Setting the JRST bit in JCCMD to 1 after interrupt handling resumes decompression.

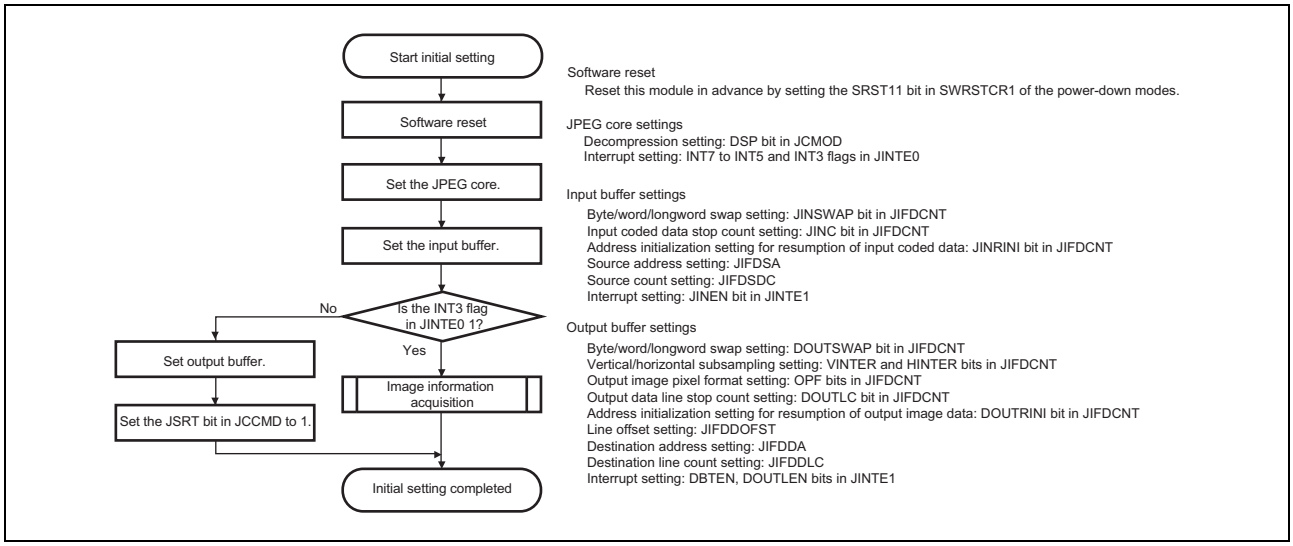


Figure 45.5 Decompression Initial Setting Flow

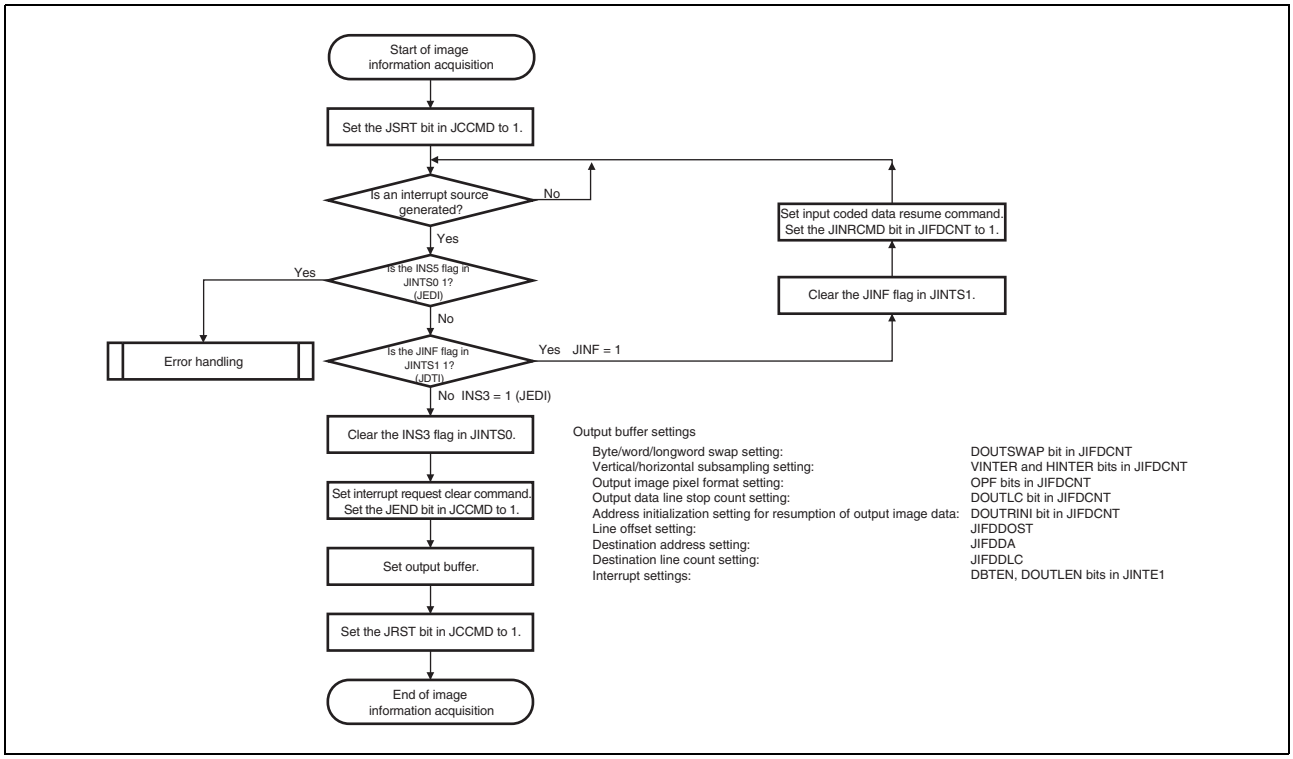


Figure 45.6 Image Information Acquisition Flow

(b) Decompression Process

The decompression process flows are described below.

- When JPEG decompression process has been completed, the INS6 bit in JINTS0 is set to 1. However, the JCU continues processing since the image data remains to be transferred. The DBTF bit in JINTS1 is set to 1 when the last image data is transferred. The interrupt source is cleared by writing 0 to the INS6 bit. However, the interrupt request asserted by this interrupt source cannot be cleared by writing 0 to the INS6 bit. Set an interrupt request clear command (by setting the JEND bit in JCCMD to 1) to clear the interrupt request.
- When the JCU has completed decompression process and all image data has been transferred, the DBTF flag in JINTS1 is set to 1. When the DBTEN bit in JINTE1 is 1 here, an interrupt is generated. The interrupt source is cleared by writing 0 to the DBTF flag.
- If the count mode for stopping input coded data is on, when the specified amount of coded data set in JIFDSDC have been read, the JINF flag in JINTS1 is set to 1, and reading is stopped. When the JINEN bit in JINTE1 is 1 here, an interrupt is generated. An interrupt source is cleared by writing 0 to the JINF bit. Setting the JINRCMD bit in JIFDCNT to 1 resumes reading.
  When the JINRINI bit in JIFDCNT is zero, the addresses for reading on resumption are continued from the addresses in the previous round of transfer.
  When the JINRINI bit is one, the address set in JIFDSA is used on resumption.
- If the count mode for stopping the output image data is on, when the specified number of image data lines set in JIFDDLC have been written, the DOUTLF flag in JINT1 is set to 1, and writing is stopped. When the DOUTLEN bit in JINTE1 is 1 here, an interrupt is generated. An interrupt source is cleared by writing 0 to the DOUTLF bit. Setting the DOUTRCMD bit in JIFDCNT to 1 resumes writing.
  When the DOUTRINI bit in JIFDCNT is zero, the addresses for writing on resumption are continued from the addresses in the previous round of transfer.
  When the DOUTRINI bit is one, the address set in JIFDDA is used on resumption.

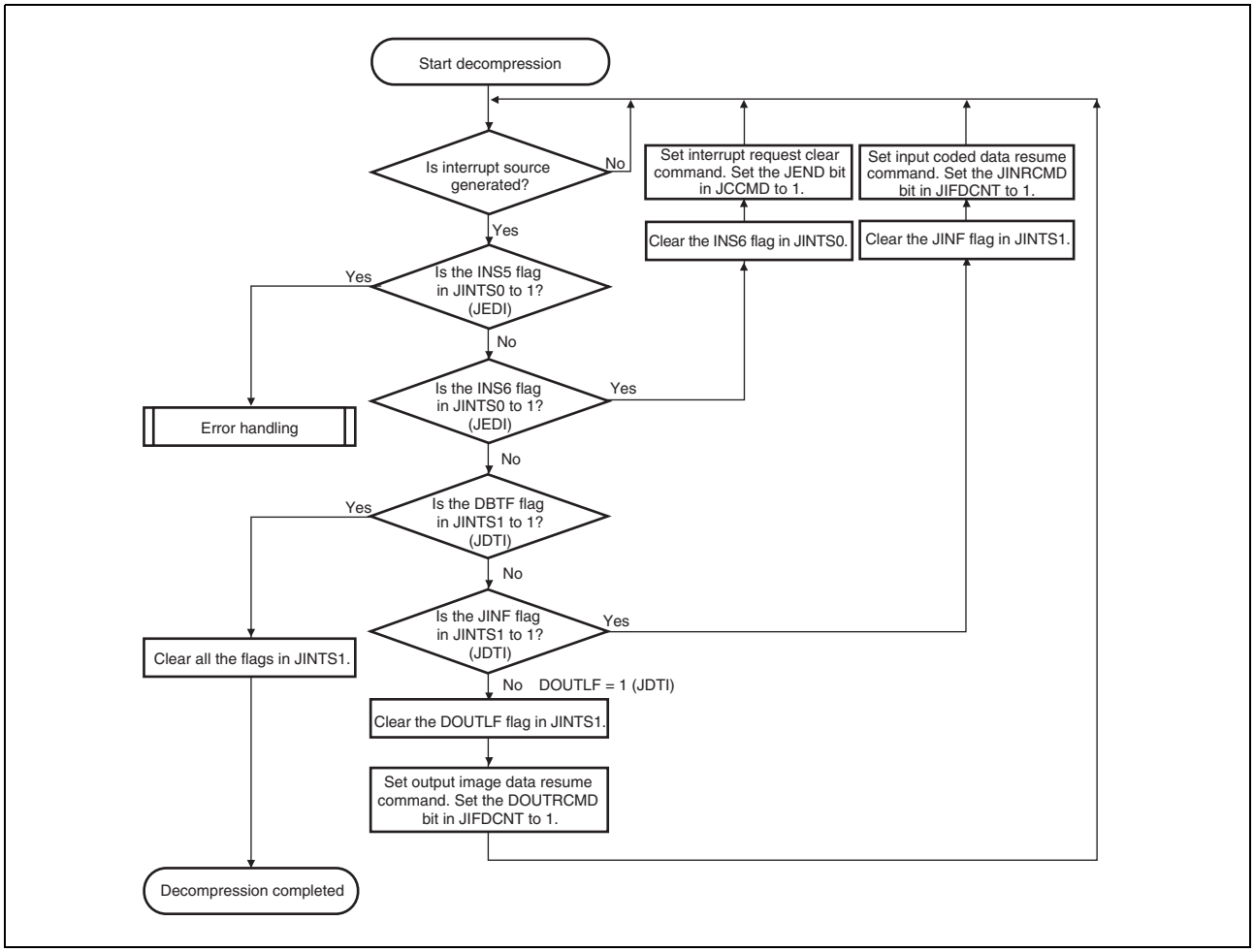


Figure 45.7 Decompression Process Flow

(c) Error Handling

If the INS5 bit in JINTS0 is 1, it indicates that there is an error in the input JPEG coded data and that the decompression process by this module has been ended. Read the ERR bits in JCDERR to determine the cause of the error. The interrupt signal asserted due to the interrupt source indicated by the INS5 bit cannot be negated by clearing the interrupt status through 0-writing. To clear the interrupt request, set the interrupt request clear command (by setting the JEND bit in JCCMD to 1).
If decompression or compression is to proceed after error handling is completed, start by making the initial settings.

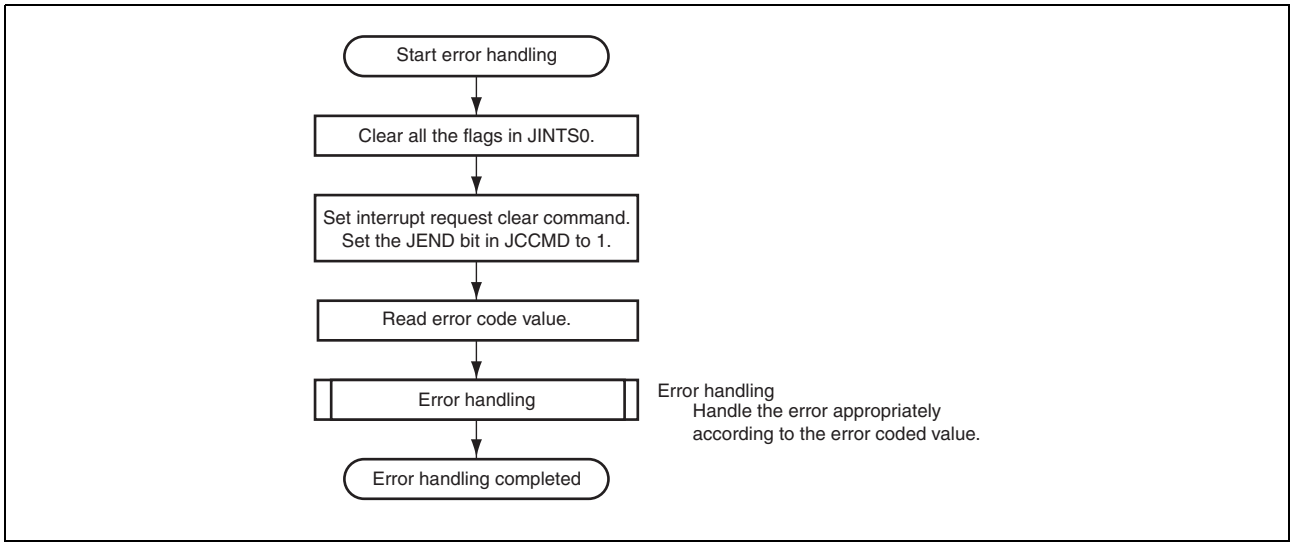


Figure 45.8 Error Handling Flow

(2) Input JPEG Coded Data

Markers to be processed in decompression are SOI, SOF0, SOS, DQT, DHT, DRI, RSTm, and EOI. Other markers except for the error markers shown below are ignored even if they are read.
The JINSWAP bits in JIFDCNT can be used to alter the arrangement for the input of coded data.

(3) JPEG Decompression Errors

(a) Error Marker

If a marker error is found while analyzing compressed data for decompression, the code to identify the error type (shown in Table 45.3) is set to ERR bits in JCDERR. When an error is detected, the JCU generates an interrupt signal and terminates decoding. The stored code value will be set to B'1010 (default value) at the start of processing of the next frame or after a bus reset.

Table 45.3 Decompression Error Codes

Code	Description
B'0000	Normal
B'0001	SOI not detected: SOI not detected until EOI detected
B'0010	SOF1 to SOFF detected
B'0011	Unprovided pixel format detected
B'0100	SOF accuracy error: Other than 8 detected
B'0101	DQT accuracy error: Other than 0 detected
B'0110	Component error 1: The number of SOF0 header components detected is other than 1, 3, or 4
B'0111	Component error 2: The number of components differs between SOF0 header and SOS
B'1000	SOF0, DQT, and DHT not detected when SOS detected
B'1001	SOS not detected: SOS not detected until EOI detected
B'1010	EOI not detected (default)
B'1011	Restart interval data number error detected
B'1100	Image size error detected
B'1101	Last MCU data number error detected
B'1110	Block data number error detected

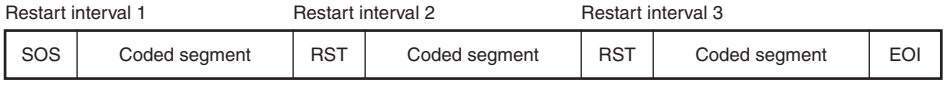
(b) Huffman Coded Segment Error

During the compressed data analysis in decompression operation, if there is an increase or decrease in the decoded data count due to an error resulting from bit reversal or missing data in the Huffman-coded segment, determine the error type, and set the error code in the ERR bit in JCDERR. Table 45.4 lists the segment error codes. The error code is set, interrupt signal is issued, and the process is ended only if the bits INT7 to INT5 in JINTE0 corresponding to the detected error is set to 1. The set code value will turn to the default value (B'1010) at the start of processing of the next frame or after a bus reset.

However, in this error detection, if an error in the Huffman-coded segment does not result in an alteration in the decoded data count, the error will go undetected.

[Example]

The number of data in a Huffman coded segment with pixel format setting YCbCr422, DRI = 2, X = 80 pixels, and Y = 8 pixels



- Number of data in the last MCU: Pixel format setting is YCbCr422, the number of data to be decoded in each MCU is 256.
- Number of data in the restart interval: In restart intervals 1 and 2, there are data of two MCUs; the number of data to be decoded is 512.
- Image size: The total number of data to be decoded is 1280.

Figure 45.9 Huffman Coded Segment

Table 45.4 Segment Error Codes

Code	Description
B'0000	Normal
B'1011	Restart interval data number error: The number of data in each interval is compared with the number of data specified by the DRI marker. If an interval has more or less data that is specified by the DRI marker, the decompression error code (1011) is set. The last interval which is shorter than the restart interval is not compared. If the DRI marker segment is not placed or the specified number is 00, an error is not detected even if the RSTm marker is placed. Also an m which indicates the order of RSTm marker modulo 8 (m = 0 to 7) is exempt from the error detection analysis. When the INT7 bit in JINTE0 is set to 0, this error is not detected.
B'1100	Image size error: The data number of an image which is calculated from the number of lines specified by the frame parameter and the number of samples per line is compared with the total number of data from SOS to EOI (in pixel units). If the numbers of data do not match, the decompression error code (1100) is set. When the INT6 bit in JINTE0 is set to 0, this error is not detected. The data number of an image is shown in MCU units. Thus the number of lines and the number of samples per line for calculation need to be shown in MCU units.
B'1101	Last MCU data number error: Whether the number of data in the MCUs at the EOI detection is shown in MCU units is checked and fractions are detected. If error (1100) occurs simultaneously, error (1100) has priority. When the INT5 bit in JINTE0 is set to 0, this error is not detected.
B'1110	Block data number error: Whether a block is an 8×8 array is checked; the check is performed for fractions. When bits INT7 to INT5 in JINTE0 are all set to 0, this error is not detected.

45.3.3 Output Pixel Format in Decompression

This module is capable of decompressing JPEG encoded data created in the YCbCr444, YCbCr422, YCbCr411 and YCbCr420 formats. The pixel format of the output image will be YCbCr422, ARGB8888, or RGB565. The flow of conversion of decompressed data to the given output pixel format is shown below.

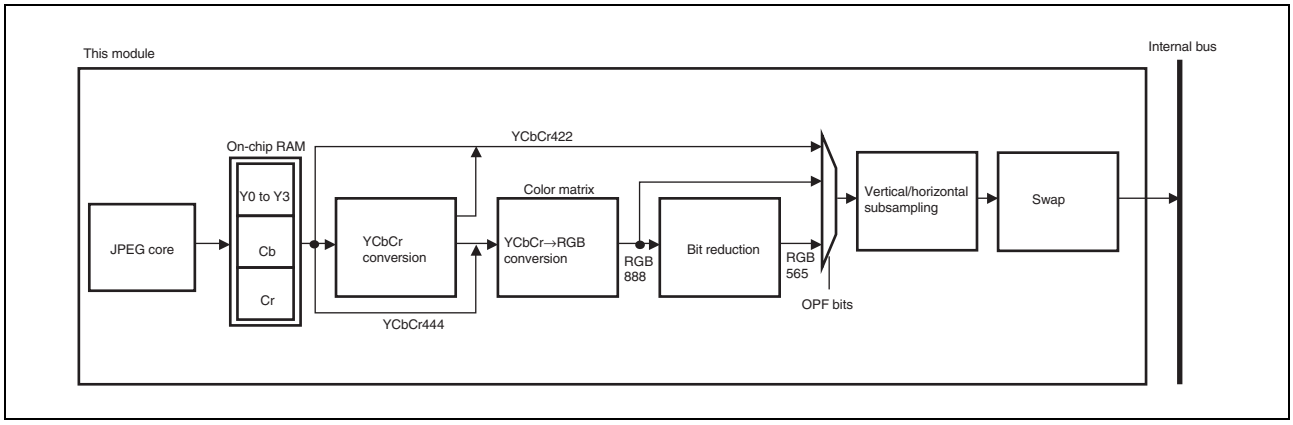


Figure 45.10 Block Diagram of Output Pixel Format Conversion in Decompression

(1) On-chip RAM

Data decoded by the JPEG core are stored in MCUs on RAM in this module.

(2) YCbCr Conversion

When data are to be output in the ARGB8888 or RGB565 format, data in the YCbCr422, YCbCr411 or YCbCr420 format are first converted to the YCbCr444 format.
When data are to be output in the YCbCr422 format, data in the YCbCr444, YCbCr411 or YCbCr420 format are converted to the YCbCr422 format.
Conversion is performed using simple interpolation.

(3) YCbCr → RGB Conversion

Data in the YCbCr444 format are converted to the RGB888 format. The following formulae are used.
R = 1.000Y + 1.402Cr
G = 1.000Y - 0.344Cb - 0.714Cr
B = 1.000Y + 1.772Cb

(4) Bit Reduction

RGB888 data is reduced to RGB565 data. The lower three bits of red and blue, and lower two bits of green are removed.

(5) Output Pixel Format Selection

The pixel format to be output is selected by the OPF bit in JIFDCNT.
Allocation of data (while the DOUTSWAP bits in JIFDCNT = 000) in the pixel format is shown below.

- YCbCr422 (32 bits/pixel)

b31 – b24	b23 – b16	b15 – b8	b7 – b0
Y0 8 bits	Cb 8 bits	Y1 8 bits	Cr 8 bits

- ARGB8888 (32 bits/pixel)

b31 – b24	b23 – b16	b15 – b8	b7 – b0
*	Red 8 bits	Green 8 bits	Blue 8 bits

Note: * This value is determined by the ALPHA[7:0] bits in JIFDADT.

- RGB565 (16 bits/pixel)

b15 – b11	b10 – b5	b4 – b0
Red 5 bits	Green 6 bits	Blue 5 bits

(6) Vertical/Horizontal Subsampling

The output data can be horizontally and vertically subsampled according to the VINTER and HINTER bit setting in JIFDCNT.
Figure 45.11 to Figure 45.13 show line subsampling modes.
For the output formats ARGB8888 and RGB565, one cell represents one pixel in the figures.
For the output format YCbCr422, one cell represents one set of Y0Cb0Y1Cr0 in the figures.
As subsampling is carried out by minimum coded units (MCU), the numbers of the horizontal and vertical block units will vary according to the decompressed pixel format.
Table 45.5 and Table 45.6 show the values of n and m in the figures.

Horizontal:

Table 45.5 Number of Horizontal Blocks

Compression Format	Output Format	n
YCbCr444	YCbCr422	1/2
YCbCr444	ARGB8888, RGB565	1
YCbCr422	YCbCr422	1
YCbCr422	ARGB8888, RGB565	2
YCbCr411	YCbCr422	2
YCbCr411	ARGB8888, RGB565	4
YCbCr420	YCbCr422	1
YCbCr420	ARGB8888, RGB565	2

Vertical:

Table 45.6 Number of Vertical Blocks

Compression Format	Output Format	m
YCbCr444	YCbCr422	1
YCbCr444	ARGB8888, RGB565	1
YCbCr422	YCbCr422	1
YCbCr422	ARGB8888, RGB565	1
YCbCr411	YCbCr422	1
YCbCr411	ARGB8888, RGB565	1
YCbCr420	YCbCr422	2
YCbCr420	ARGB8888, RGB565	2

- Subsampling into 1/2
  Even lines are skipped by subsampling.

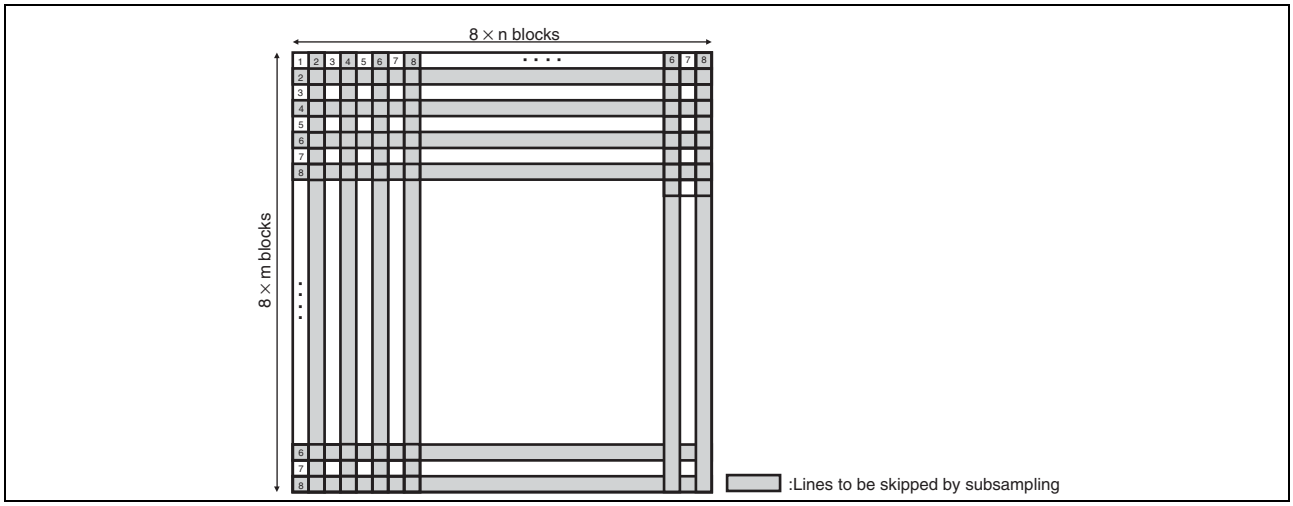


Figure 45.11 MCU when subsampling into 1/2 is selected

- Subsampling into 1/4
  The second, third, and fourth lines are skipped by subsampling.

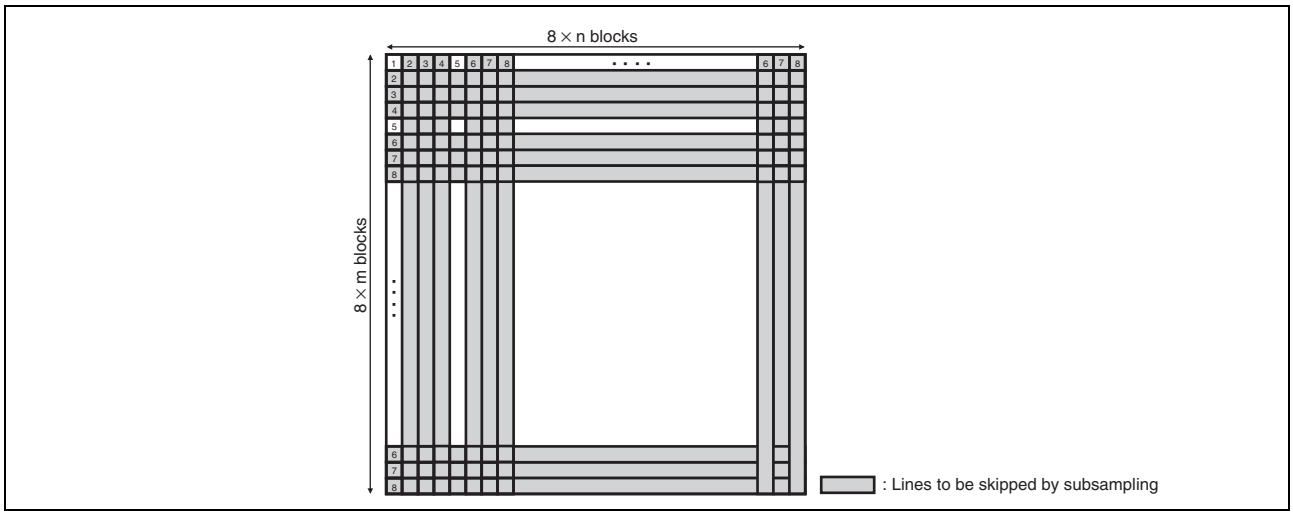


Figure 45.12 MCU when subsampling into 1/4 is selected

- Subsampling into 1/8
  The second, third, fourth, fifth, sixth, seventh, and eighth lines are skipped by subsampling.

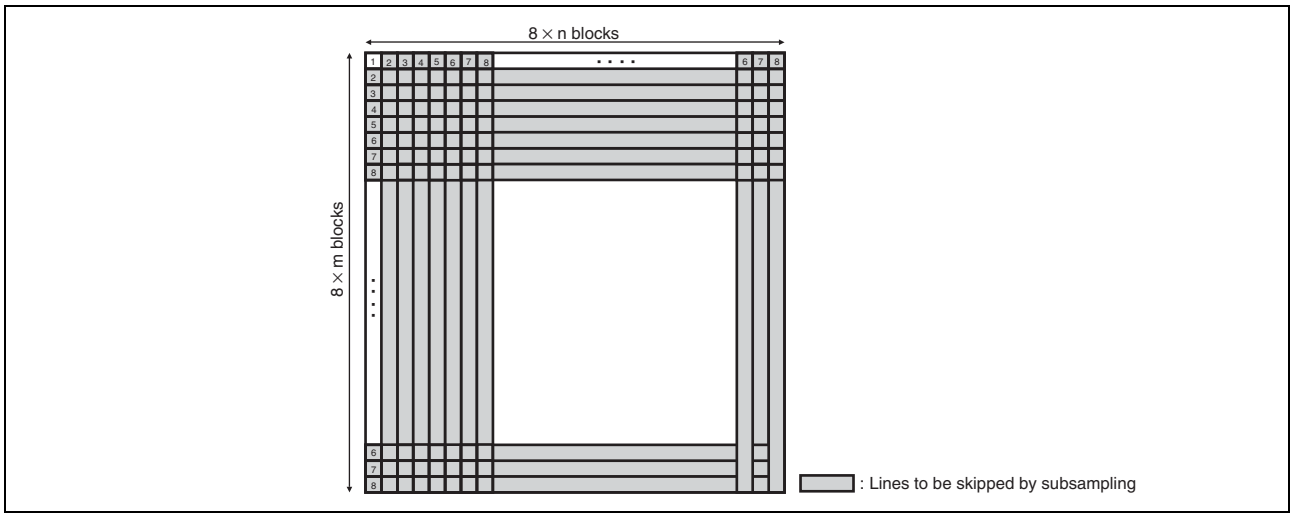


Figure 45.13 MCU when subsampling into 1/8 is selected

(7) Swap

Allocation of data can be changed by the DOUTSWAP bits in JIFECNT.

45.3.4 Storing Image Data

Figure 45.14 shows the buffer area for storing the image data.

- Start address
  Compression: JIFESA
  Decompression: JIFDDA
- Horizontal size
  Compression, decompression: JCHSZU, JCHSZD
- Vertical size
  Compression, decompression: JCVSAU, JCVSZD
- Offset
  Compression: JIFESOFST
  Decompression: JIFDDOFST

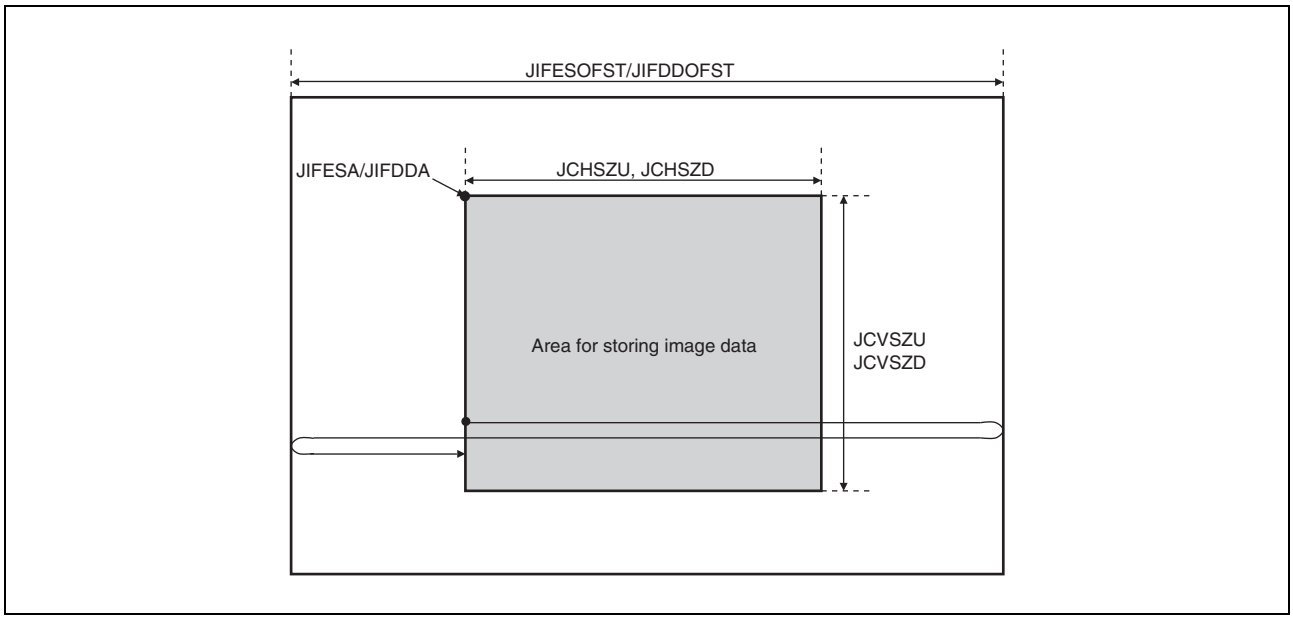


Figure 45.14 Image of Storing Image Data

45.4 Interrupts

Two types of interrupt requests, namely compression/decompression process interrupt request (JEDI) and data transfer interrupt request (JDTI), are available in this module. The two types of interrupt requests are each related to multiple sources. The interrupt request cancellation methods differ depending on the source of the interrupt request.

45.4.1 Compression/Decompression Process Interrupt Request (JEDI)

The flags in JINTS0 indicate compression/decompression-related sources. The interrupt requests asserted by these interrupt sources cannot be negated by clearing the corresponding interrupt status bits to 0. Issue an interrupt request clear command (by setting the JEND bit in JCCMD to 1) to clear the interrupt request. When a flag in JINTS0 is set to 1, a compression/decompression process interrupt request is sent to the interrupt controller.

(1) Compression

- JPEG compression process end
  When the INS6 bit in JINTS0 is 1, the JPEG compression process has been successfully completed. After all of the coded data is transferred, the JCU completes compression.
-

(2) Decompression

- JPEG decompression process end
  When the INS6 bit in JINTS0 is 1, the JPEG decompression process has been successfully completed. After all of the image data is transferred, JCU completes decompression.
- JPEG decompression error occurrence
  When the INS5 bit in JINTS0 is 1, the input JPEG coded data has an error and the JCU has stopped the decompression process. Read the error code (ERR bits in JCDERR) and identify the error source. This interrupt occurs when any of the INT7 to INT5 bits in JINTE0 is 1.
- Request for reading the image size and pixel format
  When the INS3 bit in JINTS0 is 1, JPEG coded data has been input and information regarding the image size and pixel format can be read. Since the JPEG decompression process is suspended, resume the JPEG decompression process by setting the process stop clear command after accessing the necessary registers. This interrupt occurs when the INT3 bit in JINTE0 is 1.

45.4.2 Data Transfer Interrupt Request (JDTI)

The flags in JINTS1 are the interrupt sources for transferring the image data and coded data. The interrupt requests asserted by these interrupt sources can be negated by clearing the corresponding interrupt status bits to 0.

(1) Compression

- Interrupt request generated after the specified number of input image data lines has been read
  When the DINLF bit in JINTS1 is 1, the number of image data lines specified by JIFESLC has been transferred; transfer the rest of the image data to the external buffer and resume transferring the data from the external buffer. A data transfer interrupt request is sent when the DINLEN bit in JINTE1 is 1.
- Interrupt request generated after the specified amount of output coded data have been written to
  When the JOUTF bit in JINTS1 is 1, the amount of coded data specified by JIFEDDC has been transferred. Secure a space for the next coded data in the external buffer, and resume transfer process. The data transfer interrupt request is sent when the JOUTEN bit in JINTE1 is 1.
- Interrupt request generated after all processes are completed
  When the CBTF bit in JINTS1 is 1, the JCU has completed compression and transferred all of the coded data. The data transfer interrupt request is sent when the CBTEN bit in JINTE1 is set to 1.

(2) Decompression

- Interrupt request generated after the specified number of output image data lines has been written to
  When the DOUTLF bit in the JINTS1 is 1, the number of image data lines specified by JIFDDLC has been transferred. Secure a space for the next coded data in the external buffer, and resume transfer process. A data transfer interrupt request is sent when the DOUTLEN bit in JINTE1 is 1.
- Interrupt request generated after the specified amount of input coded data has been read
  The JINF bit in JINTS1 becomes 1 when the amount of coded data specified by JIFDSDC has been transferred. Secure the next coded data in the external buffer, and resume transfer process. A data transfer interrupt is also sent at this time if the JINEN bit in JINTE1 is 1.
- Interrupt request generated after all processes are completed
  The DBTF bit in JINTS1 becomes 1 when the JCU has completed decompression and transferred all of the coded data. A data transfer interrupt request is also sent at this time if the DBTEN bit in JINTE1 is set to 1.

45.5 Bus Reset Processing

Issuing the bus reset command (setting the BRST bit in JCCMD to 1) causes a bus reset.
When the JCU is in operation, the bus reset command should not be issued. Registers below are initialized by a bus reset.

- JPEG code data count upper register (JCDTCU)
- JPEG code data count middle register (JCDTCM)
- JPEG code data count lower register (JCDTCD)
- JPEG interrupt status register 0 (JINTS0)
- JPEG code decode error register (JCDERR)
- JPEG code reset register (JCRST)

46. Capture Engine Unit

The capture engine unit (CEU) is a capture module that fetches image data externally input and transfers it to the memory. The CEU is connected to the system bus via bus bridge modules.

46.1 Features of CEU

Lists the features of CEU as follows.

(1) Image data fetch

- Captures an image output from an external module and writes YCbCr data to the memory with it separated into Y data and CbCr data.
- Fetches image data other than YCbCr data, e.g. JPEG data, RGB565, from an externally connected module, such as a camera, and sequentially writes the image data to the memory.
- Fetches an interlace source image in both-field units or one-field units and writes it to the memory. In both-field capture, an image can be stored in the memory as a frame image.

(2) Filter processing

- Performs scale-down and removal of high-frequency components (only in the horizontal direction) for an image using internal filters. Note that the scaled-down image must not exceed VGA. The filter processing can be applied to only YCbCr input data.

(3) Format conversion

- Converts image data input in the YCbCr422 format into the YCbCr420 format and writes it to the memory. Note that the conversion algorithm is simple skipping in which the chrominance component (CbCr) of the even-numbered lines is skipped.

46.2 Functional Overview of CEU

The functional overview of the CEU is shown in Table 46.1, and the main functions and their details are shown in Table 46.2.

Table 46.1 Functional Overview of CEU

Classification	Item	Function	Description	Note
Connectable camera	Size	5 megapixels	2,560 pixels × 1,920 lines	Horizontal: 4-pixel units Vertical: 4-line units The range of the image size that can be input is as follows. Horizontal: 2,560 pixels to 128 pixels Vertical: 1,920 lines to 96 lines [Note] This depends on the AC characteristics of the device to be connected, frame rate of the connected device, and transfer speed to the destination RAM.
		3 megapixels	2,048 pixels × 1,536 lines	
		2 megapixels	1,632 pixels × 1,224 lines	
		UXGA	1,600 pixels × 1,200 lines	
		SXGA (1)	1,280 pixels × 1,024 lines	
		SXGA (2)	1,280 pixels × 960 lines	
		WXGA	1,280 pixels × 768 lines	
		XGA	1,024 pixels × 768 lines	
		SVGA	800 pixels × 600 lines	
		WVGA	800 pixels × 480 lines	
		VGA	640 pixels × 480 lines	
		CIF	352 pixels × 288 lines	
		WQVGA	480 pixels × 240 lines	
		QVGA	320 pixels × 240 lines, 240 pixels × 320 lines	
		QCIF	176 pixels × 144 lines	
		QQVGA	160 pixels × 120 lines	
		Sub-QCIF	128 pixels × 96 lines	
	Input format	YCbCr422 8 bits	Cb_0, Y_0, Cr_0, $Y_{1...}$	Supports clock ratio of 1:1
			Cr_0, Y_0, Cb_0, $Y_{1...}$	
			Y_0, Cb_0, Y_1, $Cr_{0...}$	
			Y_0, Cr_0, Y_1, $Cb_{0...}$	
		YCbCr422 16 bits	{Y0, Cb0}, {Y1, Cr0}, …	
			{Y0, Cr0}, {Y1, Cb0}, …	
		Binary data	Specified amount to be fetched on edges of the sync signal	Written sequentially
			Data is fetched with the horizontal sync signal as an enable signal.	
	Horizontal and vertical sync signal polarities	Arbitrary	High-active and low-active	
	Capture start location	Arbitrary	Can be specified in camera input clock units	Horizontal: 1-cycle units Vertical: 1-HD (horizontal sync signal) units
	Number of captured pixels	Arbitrary	Can be specified in 4-pixel units horizontally and in 4-line units vertically	
	Interlace	Both-field capture	Stored as a field image	Capture: 2-VD (vertical sync signal) units
			Stored as a frame image	
		One-field capture	Top field or bottom field can be specified	Capture: 1-VD units
Memory write	Output format	YCbCr422 YCbCr420	YCbCr420 is realized by simple skipping	

Table 46.1 Functional Overview of CEU

Classification	Item	Function	Description	Note
Filter function	No scaling or scale-down	Scale-down of captured display	Desired scaling factor from 1/16 to 1 (scaled-down display must not exceed VGA)	
	Low-pass filter		Removal of high-frequency components	Only in the horizontal direction

Table 46.2 Main Functions of CEU and Their Details

Main Function	Detailed Description
Image data fetch	• Captures an image output from an external module and writes YCbCr data to the memory with it separated into Y data and CbCr data. • Fetches image data other than YCbCr data, e.g. JPEG data, from an externally connected module, such as a camera, and sequentially writes the image data to the memory. • Fetches an interlace source image in both-field units or one-field units and writes it to the memory. In both-field capture, an image can be stored in the memory as a frame image.
Filter processing	Performs scale-down and removal of high-frequency components (only in the horizontal direction) for an image using internal filters. Note that the scaled-down image must not exceed VGA. The filter processing can be applied to only YCbCr input data.
Format conversion	Converts image data input in the YCbCr422 format into the YCbCr420 format and writes it to the memory. Note that the conversion algorithm is simple skipping in which the chrominance component (CbCr) of the even-numbered lines is skipped.

Figure 46.1 shows a block diagram of the CEU.

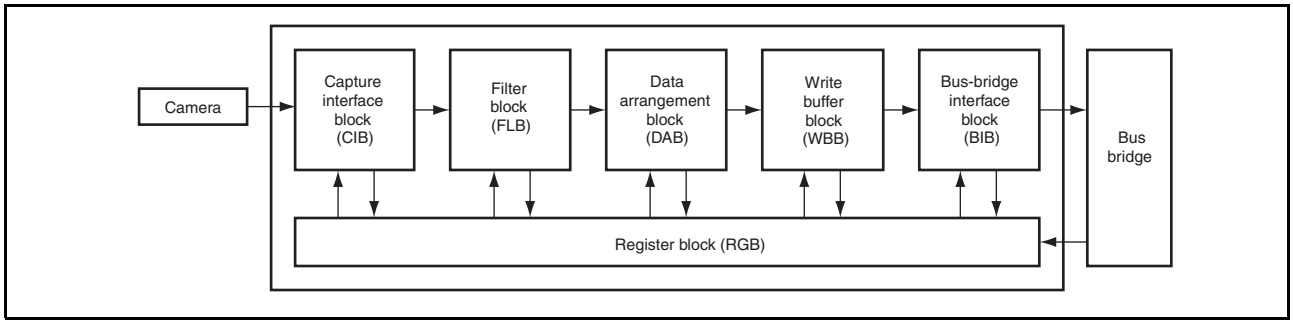


Figure 46.1 Block Diagram of CEU

46.3 Pin Configuration of CEU

The pin configuration of the CEU is shown in Table 46.3.

Table 46.3 Pin Configuration of CEU

Pin Name	Function	I/O	Description
VIO_D15 to VIO_D0*	CEU data bus	Input	Camera image data input to the CEU
VIO_CLK	CEU clock	Input	Camera clock input to the CEU
VIO_VD	CEU vertical sync	Input	Camera vertical sync signal input to the CEU
VIO_HD	CEU horizontal sync	Input	Camera horizontal sync signal input to the CEU
VIO_FLD	Field signal	Input	Field identification signal to the CEU

Note: * When the distinction according to the bus width for the data bus is not needed, VIO_D is used in this manual. Otherwise, VIO_D15 to VIO_D0 are used.

46.4 Register Descriptions of CEU

The register configuration of the CEU is shown in Table 46.4.

Most CEU registers have a 2-plane configuration (plane A and plane B). The CEU switches the planes when using these 2-plane registers. A mirror address, which is an address that can always access the register on the unused plane, is provided for each 2-plane register. Figure 46.2 shows the timing to switch the register planes. The CEU switches the register planes at the same time a VD interrupt is asserted.

In the following register descriptions, "during operation" indicates the period that begins when the CEU is activated by the CE bit in the capture start register (CAPSR) and ends when a capture end interrupt (CPE) of the capture event flag clear register (CETCR) occurs. In the read-only bits in each register, the write value should always be 0. If a value other than 0 is written to any of these bits, correct operation cannot be guaranteed.

Table 46.4 Register Configuration of CEU

			Addresses			
Register Name	Abbr.	R/W	Address (Plane A)	Address (Plane B)	Mirror Address	Access Size
CEU Capture start register	CAPSR	R/W	H'E821 0000	—	—	32
CEU Capture control register	CAPCR	R/W	H'E821 0004	—	—	32
CEU Capture interface control register*	CAMCR	R/W	H'E821 0008	—	—	32
CEU Capture interface cycle register*	CMCYR	R/W	H'E821 000C	—	—	32
CEU Capture interface offset register	CAMOR	R/W	H'E821 0010	H'E821 1010	H'E821 2010	32
CEU Capture interface width register	CAPWR	R/W	H'E821 0014	H'E821 1014	H'E821 2014	32
CEU Capture interface input format register	CAIFR	R/W	H'E821 0018	—	—	32
CEU register control register	CRCNTR	R/W	H'E821 0028	—	—	32
CEU register forcible control register	CRCMPR	R/W	H'E821 002C	—	—	32
CEU Capture filter control register	CFLCR	R/W	H'E821 0030	H'E821 1030	H'E821 2030	32
CEU Capture filter size clip register	CFSZR	R/W	H'E821 0034	H'E821 1034	H'E821 2034	32
CEU Capture destination width register	CDWDR	R/W	H'E821 0038	H'E821 1038	H'E821 2038	32
CEU Capture data address Y register	CDAYR	R/W	H'E821 003C	H'E821 103C	H'E821 203C	32
CEU Capture data address C register	CDACR	R/W	H'E821 0040	H'E821 1040	H'E821 2040	32
CEU Capture data bottom-field address Y register	CDBYR	R/W	H'E821 0044	H'E821 1044	H'E821 2044	32
CEU Capture data bottom-field address C register	CDBCR	R/W	H'E821 0048	H'E821 1048	H'E821 2048	32
CEU Capture bundle destination size register	CBDSR	R/W	H'E821 004C	H'E821 104C	H'E821 204C	32
CEU Firewall operation control register	CFWCR	R/W	H'E821 005C	—	—	32
CEU Capture low-pass filter control register	CLFCR	R/W	H'E821 0060	H'E821 1060	H'E821 2060	32
CEU Capture data output control register	CDOCR	R/W	H'E821 0064	H'E821 1064	H'E821 2064	32
CEU Capture event interrupt enable register	CEIER	R/W	H'E821 0070	—	—	32
CEU Capture event flag clear register	CETCR	R/W	H'E821 0074	—	—	32
CEU Capture status register	CSTSR	R	H'E821 007C	—	—	32
CEU Capture data size register	CDSSR	R/W	H'E821 0084	—	—	32
CEU Capture data address Y register 2	CDAYR2	R/W	H'E821 0090	H'E821 1090	H'E821 2090	32
CEU Capture data address C register 2	CDACR2	R/W	H'E821 0094	H'E821 1094	H'E821 2094	32
CEU Capture data bottom-field address Y register 2	CDBYR2	R/W	H'E821 0098	H'E821 1098	H'E821 2098	32
CEU Capture data bottom-field address C register 2	CDBCR2	R/W	H'E821 009C	H'E821 109C	H'E821 209C	32

Note: * After changing the setting of a register (CAMCR or CMCYR) that is determined by the external module characteristics, do not start capture until at least 10 external input clock cycles have elapsed.

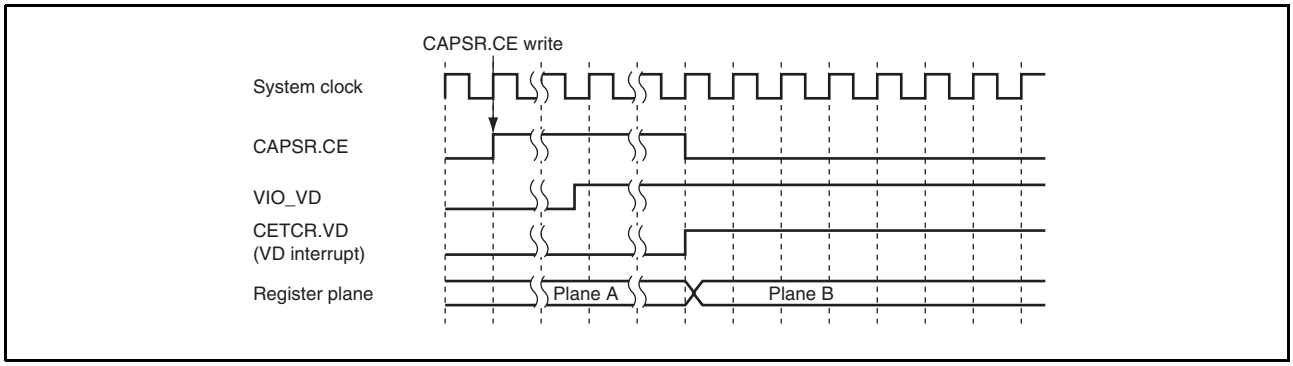


Figure 46.2 Register Plane Switching Timing (VD Polarity is High-Active in Data Enable Fetch Mode)

46.4.1 Capture Start Register (CAPSR)

CAPSR captures data input to the CEU from an external module.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CPKIL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CPKIL	0	R/W	Write 1 to this bit to perform a software reset of capturing. At a software reset, capturing ends immediately without completing capture operation until the end of a frame. Clear the CE bit to 0 when writing 1 to this bit. Processing of the capture software reset is indicated by this bit being set to 1. When this bit is 1, do not start capturing since reset processing is in progress. When restarting capture operations, after referring to the CPTON bit in CSTSR to ensure that the CEU is halted (in the idle state), wait until this bit is cleared to 0. The timing of restarting capture operations is shown in Figure 46.6. When a software reset is generated by this bit, a capture end interrupt (CPE bit in CETCR) may be output immediately after the software reset. However, such kind of interrupt should be ignored. Also, even if the capture end interrupt is not output, the interrupt source (CPE bit) must be cleared before capturing of the next frame. 0: Normal state 1: Software reset of capturing
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	CE	0	R/W	• Single-capture This bit reserves capturing of the next frame. When 1 is written to this bit, the capture of one frame starts from the next VD input, and stops when the one-frame capture end interrupt (CPE bit in CETCR) is asserted (Figure 46.7). To perform capture again, write 1 to this bit. After the VD or HD polarity is changed, do not write 1 to this bit until the next VD interrupt is asserted. As this bit indicates the capture reserve state, this bit is read as 1 after it is set to 1 and until VD is input. When VD is input, this bit returns to 0 and so is read as 0. The capture end is determined by the one-frame capture end interrupt (CPE bit). This is similar in data fetch mode. Registers should be set before the VD interrupt of the frame where capture starts next. The new register settings take effect at the next VD input. When registers are modified during capturing, the register settings take effect from the capture operations of the next VD input. If a setting register to which writing during capturing is prohibited is modified during capturing, an interrupt source (IGRW bit in CETCR) is generated. For details on the interrupt source, see the description on CETCR. • Continuous capture When this bit is set to 1 while the CTNCP bit in CAPCR is set to 1, continuous capture starts from the next frame (Figure 46.8). Note that this bit is not cleared to 0 but remains as 1. To stop capturing, clear this bit to 0; capturing stops after the current frame is completed. Continuous capture operations are possible in only image capture mode. The start address of the memory to which the captured data is written to must be set for each frame. 0: Stops capturing 1: Starts capturing

When both the VD (vertical sync signal) and HD (horizontal sync signal) polarities are high-active, one frame is defined as a period from a VD rising edge to the next VD rising edge, and one line as a period from an HD rising edge to the next HD rising edge. Figure 46.3 shows the timing of one frame (when both the VD and HD polarities are high-active).

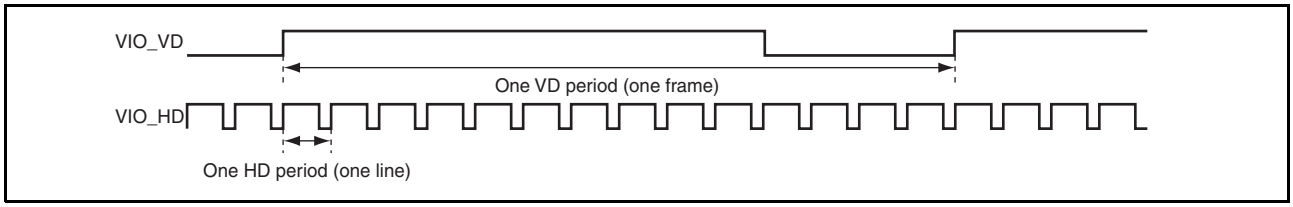


Figure 46.3 Frame Timing

When both the VD and HD polarities are high-active, similar to one frame, one field is defined as follows:

- Period from a VD rising edge to the next VD rising edge
- One line is a period from an HD rising edge to the next HD rising edge

The field identification signal FLD should be fixed for at least 1-HD period from a VD input. Figure 46.4 shows the timing of one field (when both the VD and HD polarities are high-active).

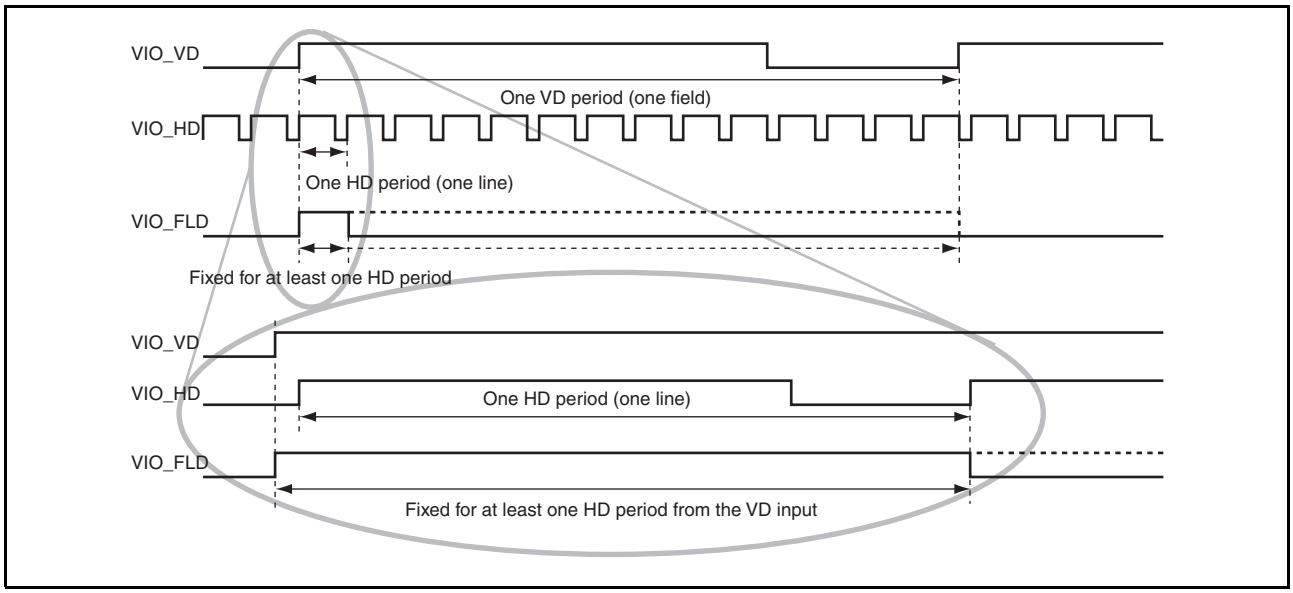


Figure 46.4 One Field Timing

In data enable fetch, one frame is defined as a period from a VD rising edge to the VD falling edge. With the HD as an enable signal (positive polarity), data of a cycle in which the HD is asserted is fetched while the VD is high. Figure 46.5 shows the timing of one frame for data enable fetch.

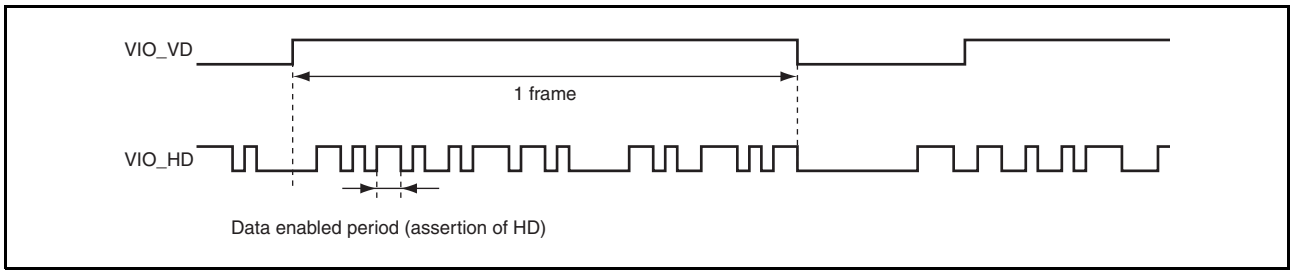


Figure 46.5 Frame Timing (Data Enable Fetch)

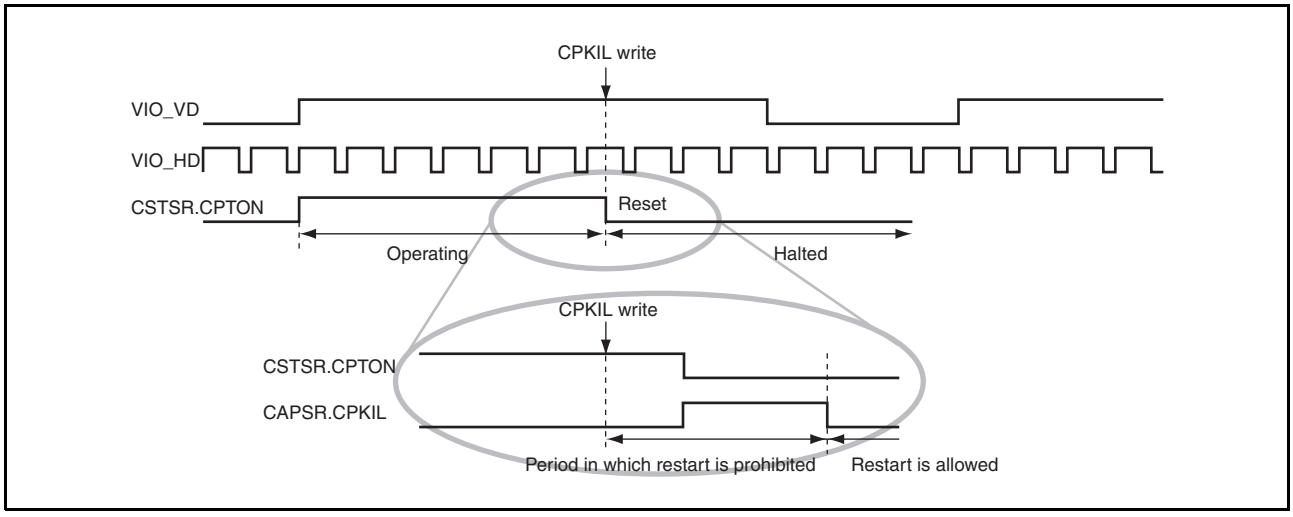


Figure 46.6 Timing of Software Reset and Restart of Capturing

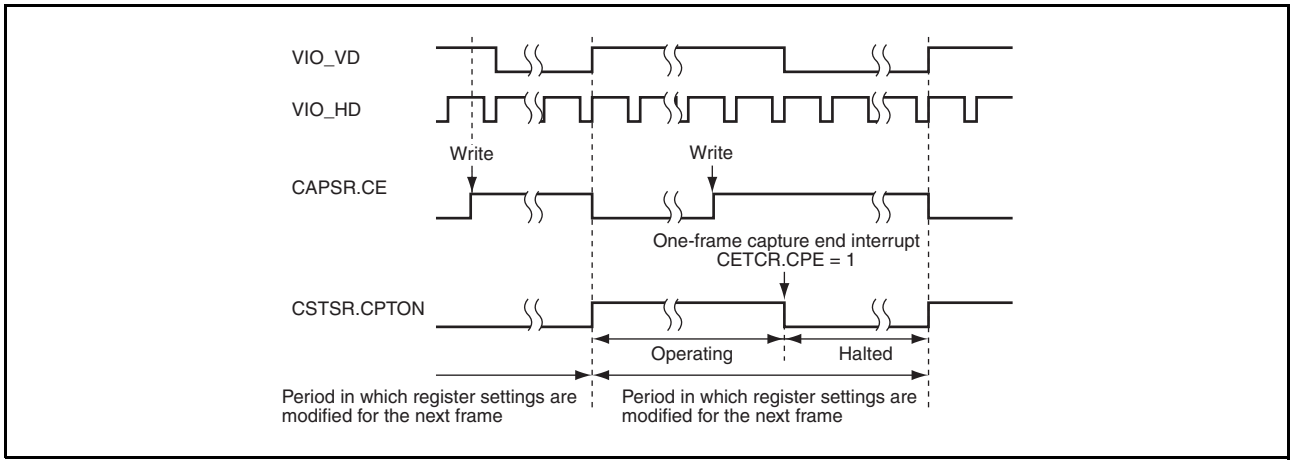


Figure 46.7 Timing of Modifying CE Bit and Register Setting in One Frame Capture

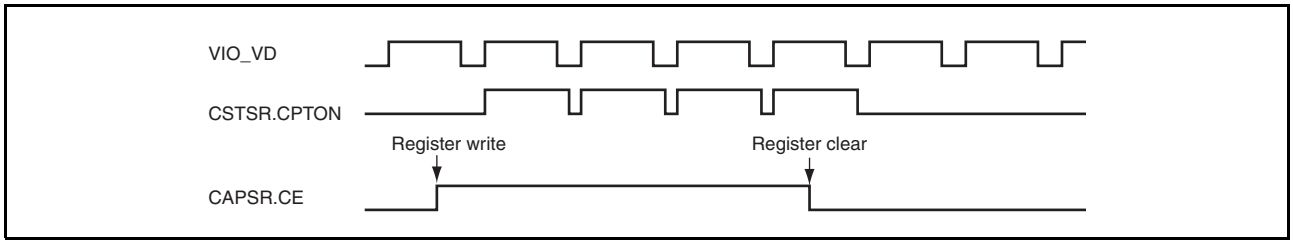


Figure 46.8 Continuous-Frame Capture

46.4.2 Capture Control Register (CAPCR)

CAPCR sets continuous-frame capture and the frame drop intervals.

Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FDRP[7:0]								—	—	MTCM[1:0]		—	—	—	CTNCP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	FDRP[7:0]	H'00	R/W	These bits set the frame drop interval in continuous-frame capture. When these bits are cleared to 0, frame drop is not performed, and all frames are captured. Figure 46.9 shows the value set in these bits and the timing of captured frames. The frame drop interval unit differs according to the capture setting. Table 46.5 shows the relationship between the capture setting and frame drop interval unit. The image of the frame drop timing for each capture setting when these bits are set to 2 is shown in Figure 46.10. In both-field capture, capturing is performed continuously for 2-VD periods, regardless of whether the second field is the top field or bottom field. In addition, in both-field capture, the frame drop counter is incremented when the first field has been identified as the top field or bottom field, regardless of whether the second field is the top field or bottom field. When 0 is written to the CE bit in CAPSR, capturing terminates after the current frame has been captured for a capture frame. However, for a drop frame, capturing is forcibly terminated in the CEU so no capture end interrupt (CPE bit in CETCR) is output. While CE bit is 1, do not change the setting of these bits. Note: Do not change the setting of these bits during continuous capture operations. To change the setting of these bits, stop continuous capture (CE bit = 0), clear the CTNCP bit (continuous capture) in CAPCR to 0, and then restart continuous capture. Continuous capture is performed during the period of CAPSR.CE = 1 shown in Figure 46.9.
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	MTCM[1:0]	00	R/W	These bits specify the unit for transferring data to a bus bridge module. The efficiency of writing image data can be improved by continuously accessing the addresses. To improve the write efficiency, set these bits to 11. The setting of these bits appear to be unchanged from the outside. 00: Transferred to the bus in 32-byte units 01: Transferred to the bus in 64-byte units 10: Transferred to the bus in 128-byte units 11: Transferred to the bus in 256-byte units (1) Image capture 00: Y data and C data are transferred in 32-byte units 01: Y data and C data are transferred in 64-byte units 10: Y data and C data are transferred in 128-byte units 11: Y data and C data are transferred in 256-byte units (2) Data fetch 00: Data is transferred in 32-byte units 01: Data is transferred in 64-byte units 10: Data is transferred in 128-byte units 11: Data is transferred in 256-byte units

Bit	Bit Name	Initial Value	R/W	Description
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CTNCP	0	R/W	When capturing is started with this bit set to 1, capturing continues until the CE bit in CAPSR is cleared to 0 or a software reset is initiated by the CPKIL bit in CAPSR (see Figure 46.8). Continuous capture must be set before capturing is started. This bit is modified only after 0 is written to the CE bit to stop capturing. If this bit is modified during capturing, correct operation cannot be guaranteed. In data fetch mode, clear this bit to 0. 0: One-frame capture when the CE bit is 1 1: Continuous capture until the CE bit is cleared to 0
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

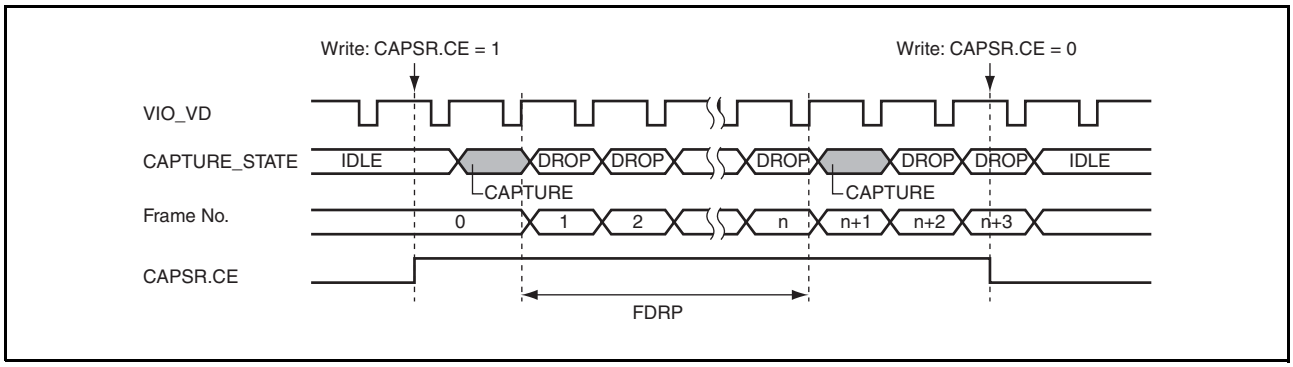


Figure 46.9 Setting of FDRP Bits and Frame Drop Timing

Table 46.5 Relationship between Capture Setting and Frame Drop Interval Unit

Input Mode	Captured Image	First Captured Image	Frame Drop Interval Unit	Capture Setting
Progressive	Frame	Frame immediately after capture start	Frame	A
Interlace	Both-field (2-VD capture)	Field immediately after capture start	2 fields (first capture field count)	B
				C
		Top field	2 fields (top-field count)	D
		Bottom field	2 fields (bottom-field count)	E
	One-field (1-VD capture)	Field immediately after capture start	First capture field	F
				G
		Top field	Top field	H
		Bottom field	Bottom field	I

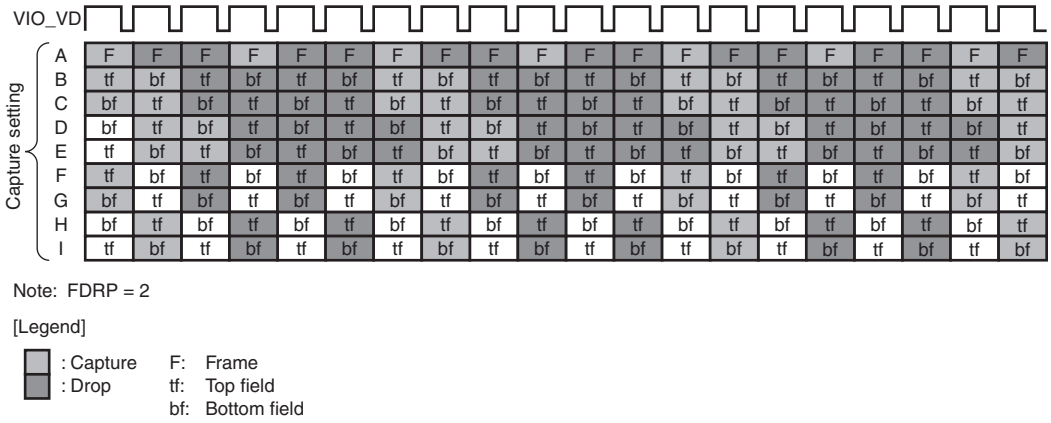


Figure 46.10 Image of Frame Drop Timing for Capture Settings (FDRP Bits = 2)

46.4.3 Capture Interface Control Register (CAMCR)

CAMCR sets the capture interface.

The following items are set by CAMCR.

- Selection between the rising edge or falling edge of the camera clock for signal capture or data fetch operation
- Selection between image capture operation or data fetch operation
- Polarities of the vertical and horizontal sync signals
- Input order of image data components (Y, Cb, and Cr) (only for image capture mode)
- Selection of digital image input pins (8 bits or 16 bits)
- Polarity of the field identification signal

CAMCR must be set according to the module connected. In data fetch mode, set the DTARY bits to B'0. Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

Note: After changing the setting of this register, do not start capture until at least 10 external input clock cycles have elapsed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VDSEL	HDSEL	FLD SEL	DSEL	—	—	—	—	—	—	—	FLD POL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	DTIF	—	—	DTARY[1:0]		—	—	JPG[1:0]		—	—	VDPOL	HDPOL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27	VDSEL	0	R/W	Sets the edge for capturing the vertical sync signal (VD) from an external module. 0: VD is captured at the rising edge of the camera clock. 1: VD is captured at the falling edge of the camera clock.
26	HDSEL	0	R/W	Sets the edge for capturing the horizontal sync signal (HD) from an external module. 0: HD is captured at the rising edge of the camera clock. 1: HD is captured at the falling edge of the camera clock.
25	FLDSEL	0	R/W	Sets the edge for capturing the field identification signal (FLD) from an external module. 0: FLD is captured at the rising edge of the camera clock. 1: FLD is captured at the falling edge of the camera clock.
24	DSEL	0	R/W	Sets the edge for fetching the image data (D7 to D0) from an external module. 0: D7 to D0 are fetched at the rising edge of the camera clock. 1: D7 to D0 are fetched at the falling edge of the camera clock.
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
16	FLDPOL	0	R/W	Sets the polarity of the field identification signal (FLD) from an external module. 0: When the FLD signal is high-active, the field is detected as the top field and when low-active, the field is detected as the bottom field. 1: When the FLD signal is low-active, the field is detected as the top field and when high-active, the field is detected as the bottom field.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	DTIF	0	R/W	Sets the digital image input pins from which data is to be captured. 0: Data input to 8-bit digital image input pins is captured 1: Data input to 16-bit digital image input pins is captured
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	DTARY[1:0]	00	R/W	These bits set the input order of the luminance component and chrominance component. The order in which the luminance component (Y) and chrominance component (Cb and Cr) are input from an external module differs among modules. The CEU supports the input orders shown in Figure 46.12. Set the corresponding value in these bits. In data fetch mode, set these bits to 00. (1) 8-bit interface 00: Image input data is fetched in the order of Cb_0, Y_0, Cr_0, and Y_1 01: Image input data is fetched in the order of Cr_0, Y_0, Cb_0, and Y_1 10: Image input data is fetched in the order of Y_0, Cb_0, Y_1, and Cr_0 11: Image input data is fetched in the order of Y_0, Cr_0, Y_1, and Cb_0 (2) 16-bit interface 00: Image input data is fetched in the order of {Cb0, Y0} and {Cr0, Y1} 01: Image input data is fetched in the order of {Cr0, Y0} and {Cb0, Y1} 10: Image input data is fetched in the order of {Y0, Cb0} and {Y1, Cr0} 11: Image input data is fetched in the order of {Y0, Cr0} and {Y1, Cb0}
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	JPG[1:0]	00	R/W	These bits select the fetched data type. 00: Image capture mode (input data are separated into Y data and CbCr data for output to the memory) 01: Data synchronous fetch mode (specified size of input data are output to the specified memory addresses in order of input and in synchronization with the sync signal) 10: Data enable fetch mode (input data are fetched with HD as an enable signal and output to the specified addresses in memory in order of input)
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	VDPOL	0	R/W	Sets the polarity for detection of the vertical sync signal input from an external module. Figure 46.14 and Figure 46.15 show the relationship between the VIO_VD and VIO_HD signals and VD interrupt when high-active is selected. Since a VD interrupt may occur when this bit value is modified, the VD bit in CETCR must always be cleared to 0 when this bit value is changed. In data enable fetch mode, this bit is not used and the sense for detection is always active high. 0: Vertical sync signal (VD) from an external module is detected as high-active 1: Vertical sync signal (VD) from an external module is detected as low-active
0	HDPOL	0	R/W	Sets the polarity for detection of the horizontal sync signal input from an external module. Figure 46.16 shows the relationship between the HD and HD interrupt when high-active is selected. Since an HD interrupt may occur when this bit value is modified, the HD bit in CETCR must always be cleared to 0 when this bit value is changed. 0: Horizontal sync signal (HD) from an external module is detected as high-active 1: Horizontal sync signal (HD) from an external module is detected as low-active

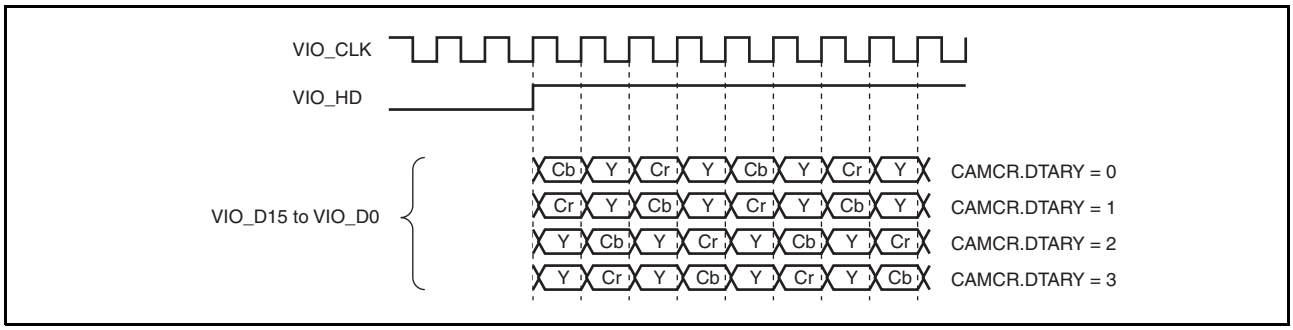


Figure 46.11 Input Order of Image Data

The JPG bit in CAMCR selects whether digital image data is fetched or data such as JPEG is fetched. In addition, when data such as JPEG is fetched, select whether the specified amount of data is continuously fetched in synchronization with the sync signal or data is fetched while the horizontal sync signal is enabled.

In data enable fetch mode, one frame is defined as a period from the rising edge to the falling edge of the vertical sync signal (VD) for data fetching. The horizontal sync signal (HD) is enabled only when the VD is high and treated as an enable signal. Data input in the cycle in which the HD is asserted (high) is fetched and output to the memory continuously.
This module starts fetching data at the rising edge of the VD and stops fetching data at the falling edge of the VD in data enable fetch mode. Thus, if the VD remains high and does not go low, end processing does not start. In addition, if the VD remains high and the HD also remains asserted, data continues to be fetched.
Figure 46.12 and Figure 46.13 show the interface timing in data enable fetch mode.

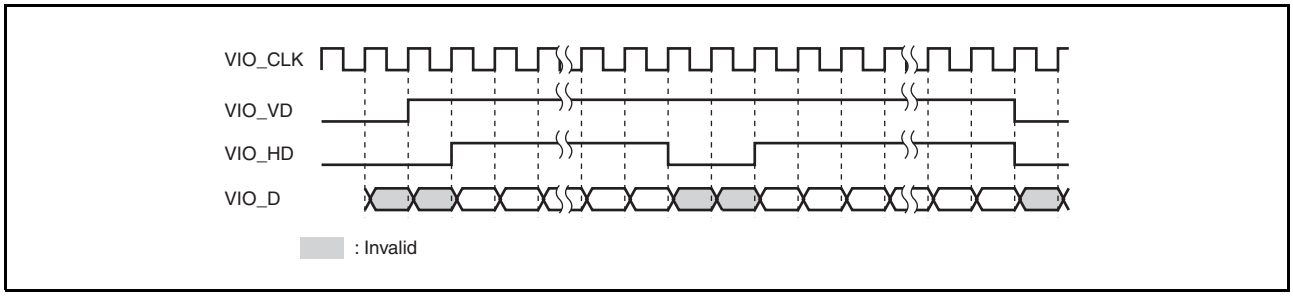


Figure 46.12 Data Enable Fetch Timing (HD Asserted (High) While VD is High)

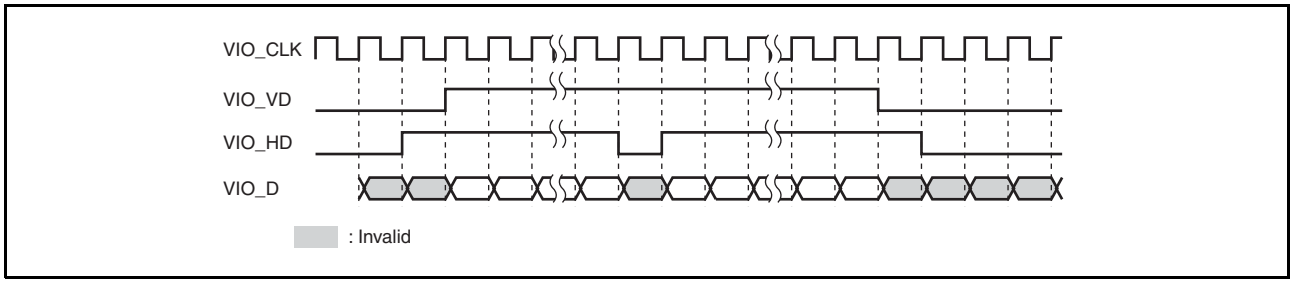


Figure 46.13 Data Enable Fetch Timing (HD Asserted (High) When VD is Not High)

In data enable fetch mode, this module generates a VD interrupt in response to detection of the active level of VD. In image capture mode and data synchronous fetch mode, this module generates a VD interrupt in response to the first detection of the active level of HD following detection of the active level of VD. Note that, when VD and HD are asserted and detected at the same time, this module generates a VD interrupt at that time.
Figure 46.14 to Figure 46.16 show the relationships between the VIO_VD signal and the VD interrupt, the VIO_VD and VIO_HD signals and the VD interrupt, and the VIO_HD signal and the HD interrupt.

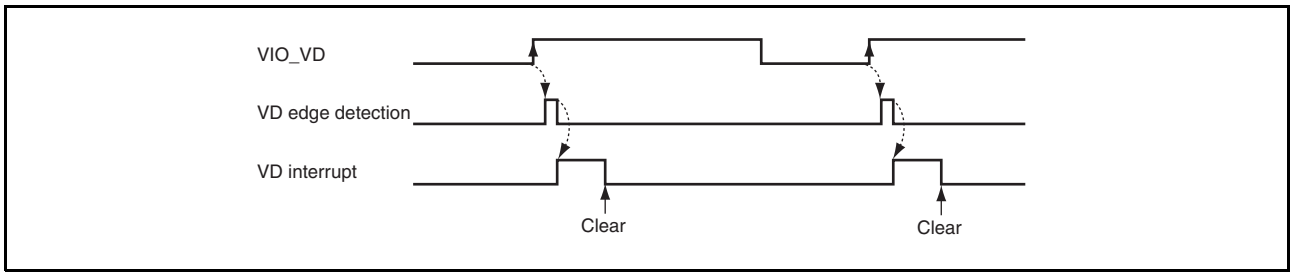


Figure 46.14 Relationship between VIO_VD and VD Interrupt when VD is High-Active (In Data Enable Fetch Mode)

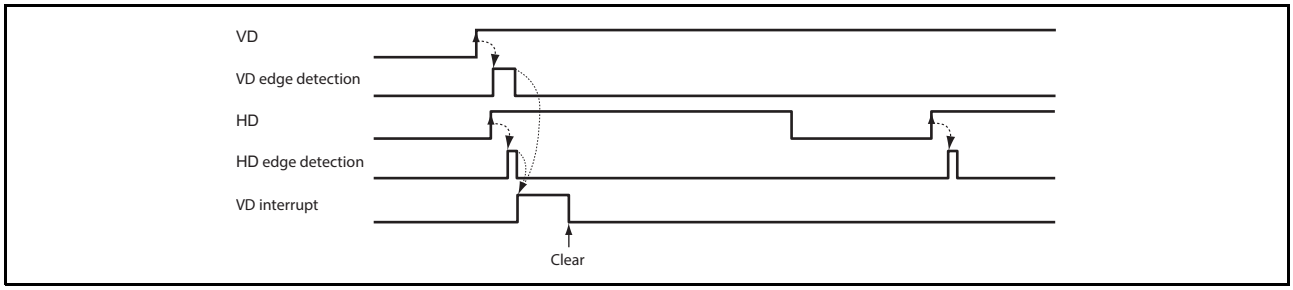


Figure 46.15 Relationship between the VIO_VD and VIO_HD signals and the VD Interrupt when VD and HD are High-Active (in Image Capture Mode or Data Synchronous Fetch Mode)

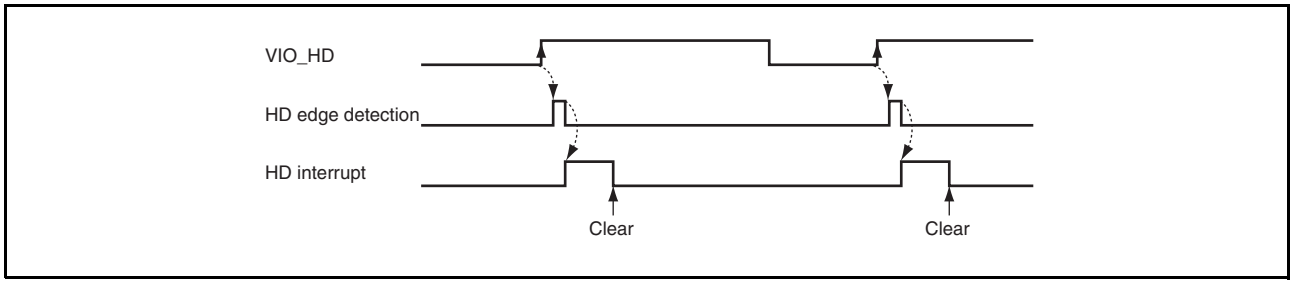


Figure 46.16 Relationship between VIO_HD and HD Interrupt when HD is High-Active

46.4.4 Capture Interface Cycle Register (CMCYR)

CMCYR is used to detect an illegal VD and an illegal HD. For HD, the number of cycles from a rising edge of HD to the next rising edge is set (falling edges when low-active is selected for HD). For VD, the number of HD inputs from a rising edge of VD to the next rising edge is set (falling edges when low-active is selected for VD).

Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

Set 0 in all bits of this register, during data enable fetch mode.

Note: After changing the setting of this register, do not start capture until at least 10 external input clock cycles have elapsed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	VCYL[13:0]													
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	HCYL[13:0]													
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 16	VCYL[13:0]	H'0000	R/W	Vertical HD Count of External Module These bits set the number of VD cycles of an external module with the number of HD inputs. The interrupt source bit IGVS in CETCR is set to 1 when the actual number of VD cycles input from the external module differs from this setting. Set these bits for detecting an illegal VD. When these bits are all cleared to 0, the interrupt source bit IGVS in CETCR is not set to 1. Though the interrupt source bit IGVS in CETCR may be set to 1 after the VDPOL bit (VD polarity) in CAMCR is changed, this interrupt should be ignored.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 0	HCYL[13:0]	H'0000	R/W	Horizontal Cycle Count of External Module These bits set the number of HD cycles of an external module. The interrupt source bit IGHS in CETCR is set to 1 when the actual number of HD cycles input from the external module differs from this setting. Set these bits for detecting an illegal HD. When these bits are all cleared to 0, the interrupt source bit IGHS in CETCR is not set to 1. Though the interrupt source bit IGHS in CETCR may be set to 1 after the HDPOL bit (HD polarity) in CAMCR is changed, this interrupt should be ignored.

46.4.5 Capture Interface Offset Register (CAMOR)

CAMOR sets the location to start capturing when capturing images.

Since the number of HD (horizontal sync signal) inputs from a VD (vertical sync signal) input to the start of a valid image period, and the number of clock cycles from an HD input to the start of a valid image period differ among external modules, these must be set in CAMOR. By setting a value greater than the valid image area, part of the image can be clipped for capture. When fetching data, the setting of this register becomes the number of cycles (HD count) up to the start of a valid data period.

This register is not used, during data enable fetch mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VOFST[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	HOFST[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	VOFST[11:0]	H'000	R/W	These bits specify the capture start location in terms of the HD count from a vertical sync signal (1-HD units). The blanking period from a vertical sync signal differs among external modules. Therefore, the vertical capture start location must be specified by these bits in terms of the HD count from a vertical sync signal so that an image can be captured from the valid image area (see Figure 46.17). Some external modules output a vertical sync signal as a data enable signal. In this case, there is no blanking period so these bits must be cleared to 0 (see Figure 46.18).
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	HOFST[12:0]	H'0000	R/W	These bits specify the capture start location in terms of the number of clock cycles from a horizontal sync signal (1-cycle units). The blanking period from a horizontal sync signal differs among external modules. Therefore, the horizontal capture start location must be specified by these bits in terms of external input clock cycles from a horizontal sync signal so that an image can be captured from the valid image area. This is similar in data synchronous fetch mode (see Figure 46.19). Some external modules output a horizontal sync signal as a data enable signal. In this case, there is no blanking period so these bits must be cleared to 0 (see Figure 46.20). Note: The first HD (horizontal sync signal) being input simultaneously or after the first VD (vertical sync signal) is the operating condition of the CEU. These inputs are affected by the polarities (set by the VDPOL and HDPOL bits in CAMCR).

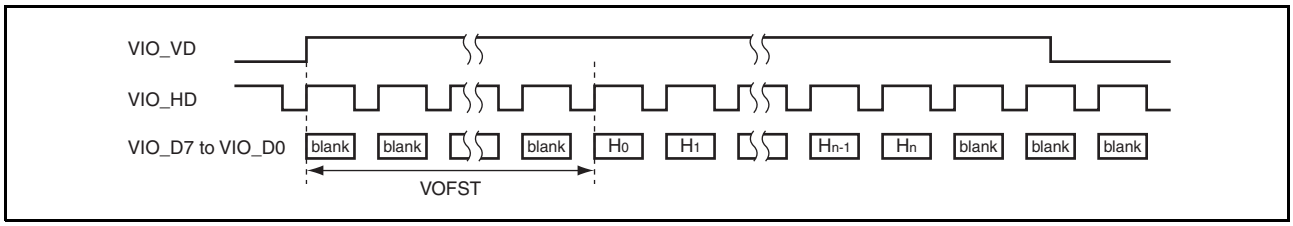


Figure 46.17 Vertical Offset

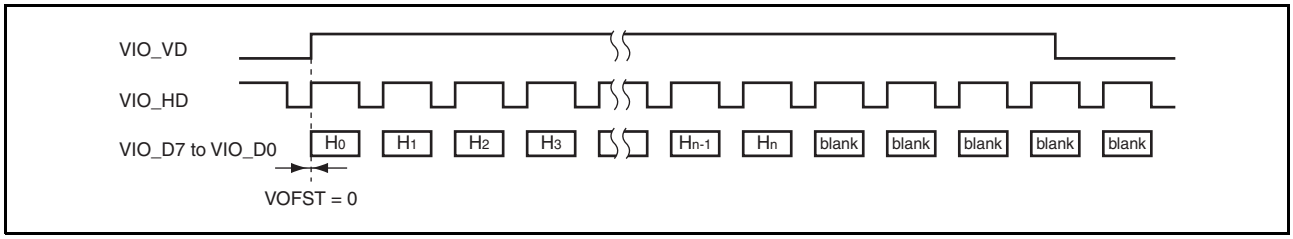


Figure 46.18 Timing when VD is Data Enable Signal

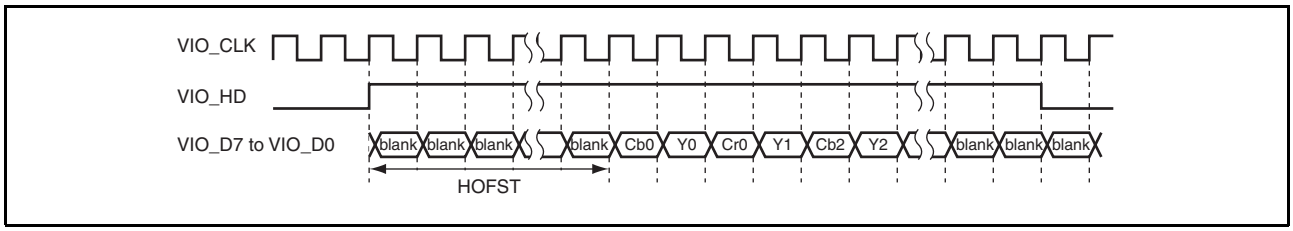


Figure 46.19 Horizontal Offset

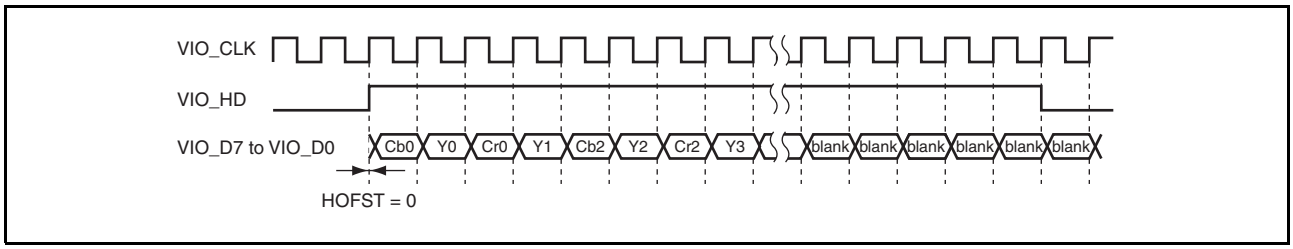


Figure 46.20 Timing when HD is Data Enable Signal (8-bit Interface)

46.4.6 Capture Interface Width Register (CAPWR)

CAPWR sets the fetch (capture) cycle width when capturing images.

The cycle width unit differs according to the interface and the data type to be captured. For each setting unit, see Table 46.6.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VWDTH[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	HWDTH[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 18 17, 16	VWDTH[11:2] VWDTH[1:0]	H'000	R/W R	These bits specify the vertical capture period (4-HD units). These bits specify the number of lines (HD count) to be captured from the location specified by the VOFST bits in CAMOR. Figure 46.21 shows the timing when the vertical blanking period is 0. The CEU captures only the number of lines (HD count) specified by these bits in the vertical direction. Make a setting in the same way to obtain data synchronization. The maximum value to be set is 1,920 HD (5 megapixels).
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	HWDTH[12:1] HWDTH[0]	H'0000	R/W R	These bits specify the horizontal capture period. These bits specify the number of cycles to be captured from the location specified by the HOFST bits. Figure 46.22 shows the timing when the horizontal blanking period is 0. The CEU captures for only the number of cycles specified by these bits in the horizontal direction. Make a similar setting for data synchronous fetch. The maximum value to be set is as follows: • 8-bit interface Image capture (8-cycle units) : 5,120 cycles (2,560 pixels) Data synchronous fetch (4-cycle units) : 2,560 cycles (2,560 bytes) • 16-bit interface Image capture (4-cycle units) : 2,560 cycles (2,560 pixels) Data synchronous fetch (2-cycle units) : 1,280 cycles (2,560 bytes) Note: In data synchronous fetch mode, set CFSZR and CDWDR according to the values set in this register. For details, see the descriptions on CFSZR and CDWDR.

Table 46.6 Unit for Setting Fetch (Capture) Cycle Width

	Vertical Direction		Horizontal Direction	
Interface	Image Capture	Data Synchronous Fetch	Image Capture	Data Synchronous Fetch
8-bit interface	4 HD	4 HD	8 cycles	4 cycles
16-bit interface	4 HD	4 HD	4 cycles	2 cycles

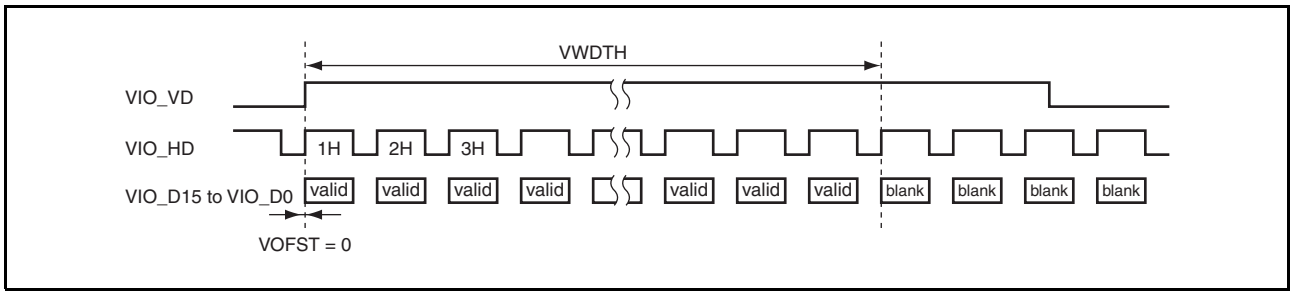


Figure 46.21 Vertical Capture Timing

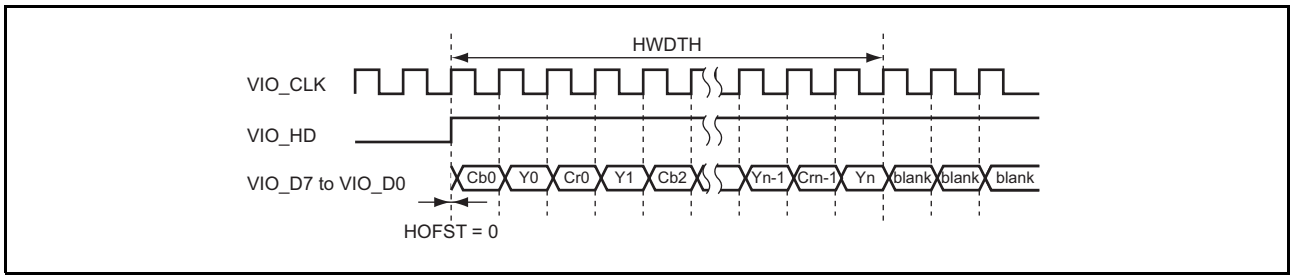


Figure 46.22 Horizontal Capture Timing (Image Capture with 8-bit Interface)

46.4.7 Capture Interface Input Format Register (CAIFR)

CAIFR sets the input mode (progressive or interlace) for capturing images, the images to be captured (frame, both-field, or one-field), the image from which capturing starts (top field or bottom field), etc. CAIFR is not used in data fetch mode.

Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

The items set by CAIFR are listed in Table 46.7.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	IFS	—	—	—	CIM	—	—	FCI[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	IFS	0	R/W	Sets the input mode for capturing images. 0: Progressive 1: Interlace
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	CIM	0	R/W	Sets the images to be captured. Clear this bit to 0 when the input mode for image capture is progressive (frame image) or when the input mode for image capture is interlace for continuous capture of both the top and bottom fields. Set this bit to 1 when the input mode for image capture is interlace for capture of only a one-field image. 0: Capture of frame image (1 VD) or both-field image (2 VD) 1: Capture of one-field image (1 VD)
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	FCI[1:0]	00	R/W	These bits set the timing to start capturing. The timing to start capturing is set by specifying the image to be captured first. Set these bits to 00 when the input mode is progressive. 00: Capture starts from the VD input immediately after the CEU activation regardless of it being a top or bottom field 01: After the CEU activation, input of a top-field image is waited, and then capture starts from the top field 10: After the CEU activation, input of a bottom-field image is waited, and then capture starts from the bottom field 11: Setting prohibited

Table 46.7 CAIFR Setting Items

Input Mode	IFS Bit	Captured Image	CIM Bit	Image to Start Capture	FCI Bits
Progressive	0	Frame	0	Frame immediately after activation	00
Interlace	1	Both-field (2-VD capture)	0	Field immediately after activation	00
				Top field	01
				Bottom field	10
				Setting prohibited	11
		One-field (1-VD capture)	1	Field immediately after activation	00
				Top field	01
				Bottom field	10
				Setting prohibited	11

In frame image capture and one-field image capture, a one-frame capture end interrupt occurs when capture for 1 VD finishes. In both-field image capture, a one-field capture end interrupt occurs when capture for 1 VD finishes and a one-frame capture end interrupt occurs when capture for 2 VD finishes. At this time, a one-field capture end interrupt occurs simultaneously with a one-frame capture end interrupt. Figure 46.23 shows the timing of a one-frame capture end interrupt and one-field capture end interrupt in both-field image capture.

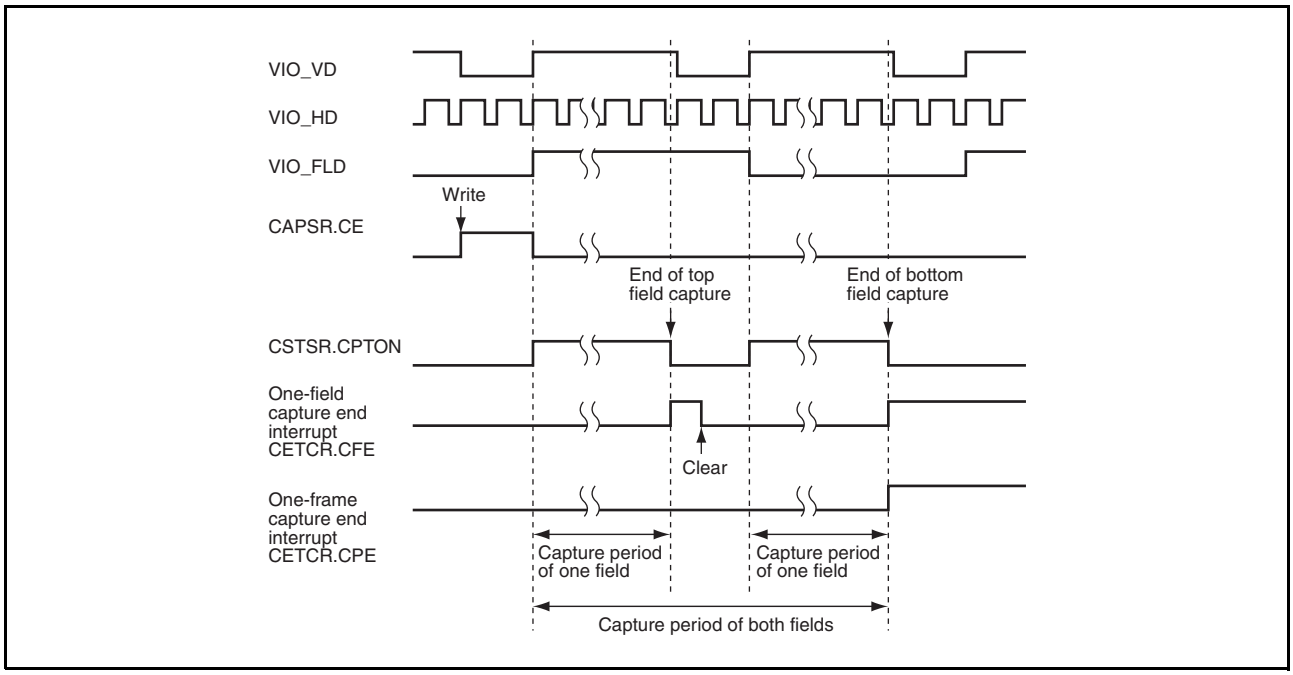


Figure 46.23 One-Frame Capture End Interrupt and One-Field Capture End Interrupt in Both-Field Image Capture

A captured frame image or captured one-field image is stored in the memory from the addresses set in CDAYR and CDACR (Figure 46.24). Captured both-field images are stored in different memory areas depending on whether it is a top-field or bottom-field image. A top-field image is stored in the memory from the addresses set in CDAYR and CDACR whereas a bottom-field image is stored in the memory from the addresses set in CDBYR and CDBCR (Figure 46.25).

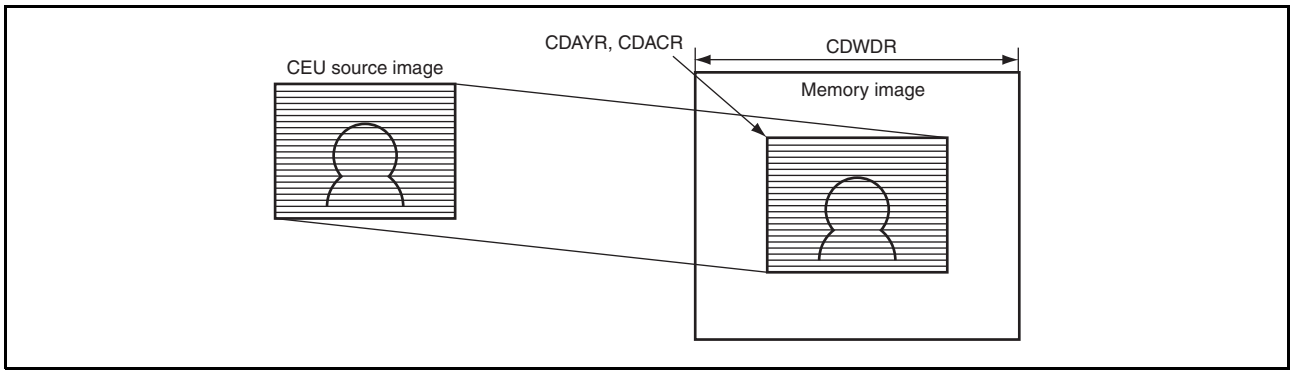


Figure 46.24 Image of Storing Captured Frame Image or Captured One-Field Image in Memory

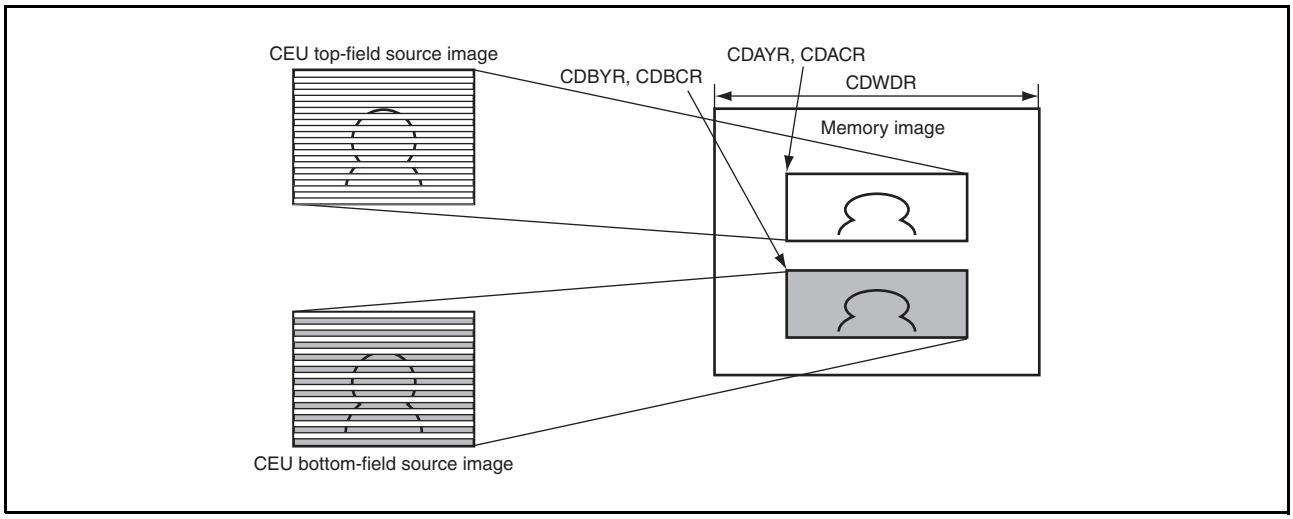


Figure 46.25 Image of Storing Captured Both-Field Images in Memory

If the FCI bits are set to B'00 for continuous capture in interlace input mode, images are continuously captured for 2 VD with the first captured field as the reference in both-field image capture (Figure 46.26). In one-field image capture, only the first captured field is continuously captured for 1 VD (Figure 46.27).

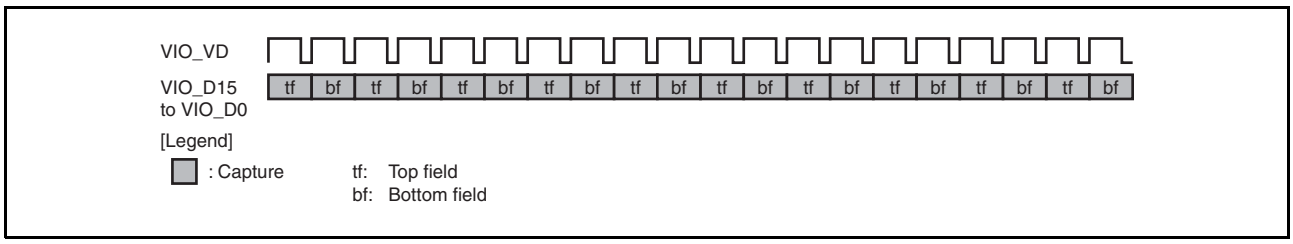


Figure 46.26 Continuous Both-Field Capture in Interlace Mode
(Image Immediately after Activation is Top Field (FCI Bits = B'00))

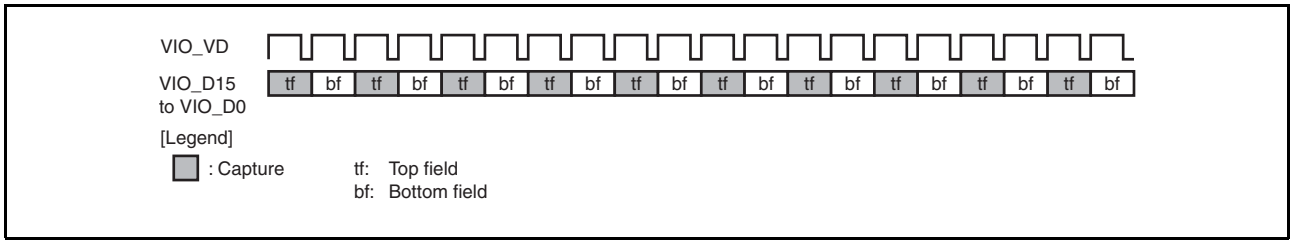


Figure 46.27 Continuous One-Field Capture in Interlace Mode
(Image Immediately after Activation is Top Field (FCI Bits = B'00))

(1) Storage of Interlace Input as Frame Image

The CEU can store an interlace source image in the memory as a frame image. To store an interlace source image as a frame image, make the following register settings:

Input mode: Interlace (IFS bit = B'1)
Capture image: Both-field (CIM bit = B'0)
Image to start capture: Any setting other than the prohibited setting (FCI bits = as desired)

Figure 46.28 shows a memory image of capturing both fields of an interlace input and storing it as a frame image in the memory. Set the start addresses of the memory destination for the captured top-field image in CDAYR and CDACR, and the start addresses of the memory destination for the captured bottom-field image in CDBYR and CDBCR. When storing an interlace image as a frame image in the memory, set the horizontal image size of the memory area in CDWDR with the top-field image and bottom-field image placed next to each other as shown in Figure 46.28. In addition, set the number of captured lines of the field image in the VWDTH bits in CAPWR.

A memory image of folding the horizontal image size of the memory area in Figure 46.28 at CDWDR/2 is shown in Figure 46.29. Setting the registers to form the image in Figure 46.28 enables an interlace image to be stored as a frame image in the memory as shown in Figure 46.29.

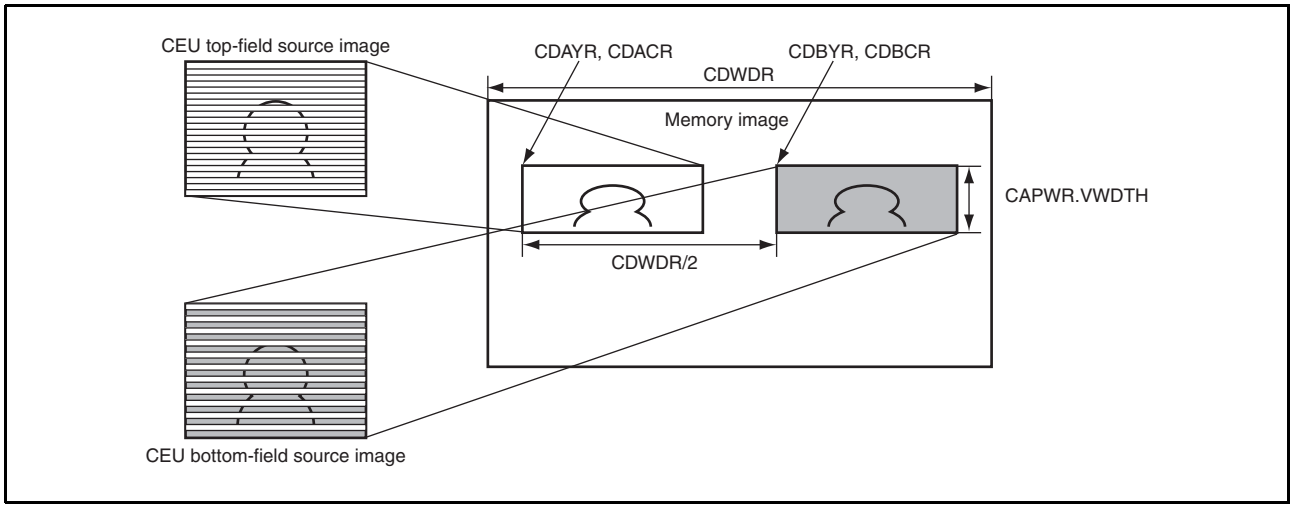


Figure 46.28 Image of Storing Captured Both-Fields of Interlace Input in Memory

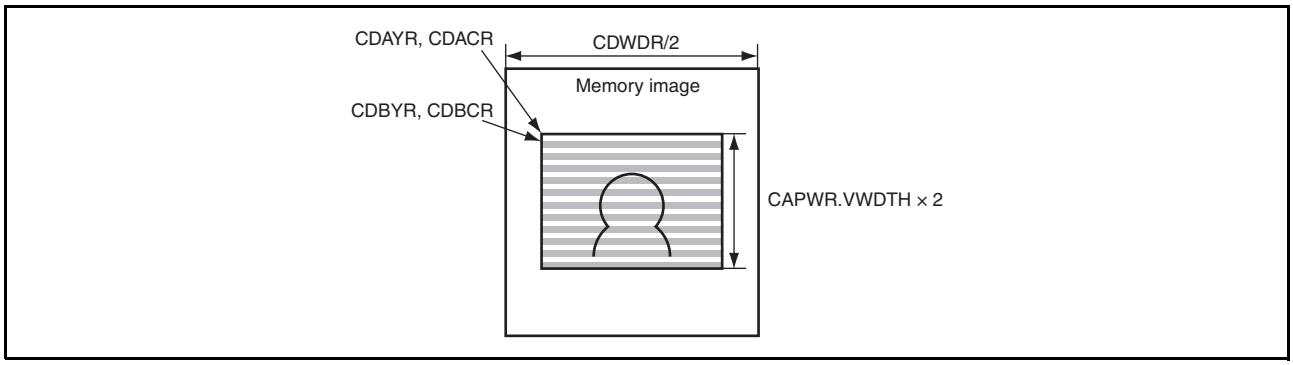


Figure 46.29 Image of Storing Interlace Input as Frame Image in Memory

46.4.8 CEU Register Control Register (CRCNTR)

CRCNTR controls switching of the planes of registers with a 2-plane configuration.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	RVS	—	—	RS	RC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	RVS	0	R/W	Sets the timing to switch the register plane in both-field capture. The setting of this bit is valid only when the RC bit is 1 in both-field capture. 0: Switches the register plane every 2 VD 1: Switches the register plane every 1 VD
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	RS	0	R/W	Specifies which register plane is used by the CEU in synchronization with VD. The setting of this bit is valid only when the RC bit is 0. 0: Uses plane A of the register 1: Uses plane B of the register
0	RC	0	R/W	Specifies switching of the register plane used by the CEU in synchronization with VD. If the register plane is not switched, the register plane specified by the RS bit is used. 0: Uses the specified register plane in synchronization with VD 1: Switches the register plane in synchronization with VD

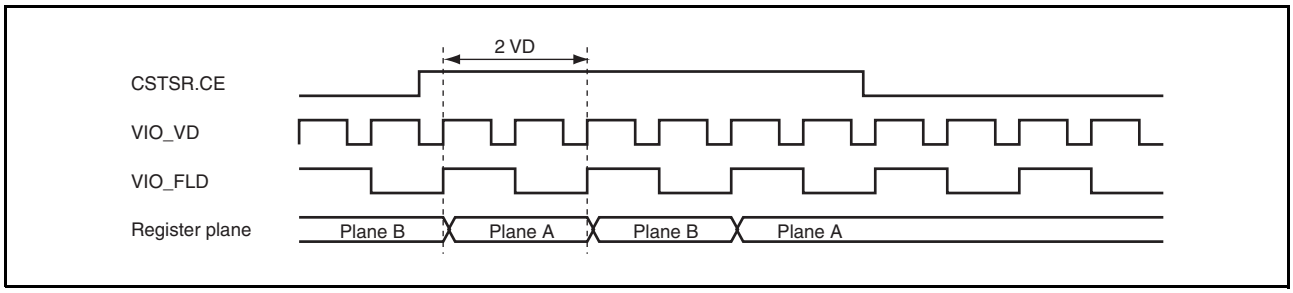


Figure 46.30 Timing for Register Plane Switching when RVS Bit is B'0

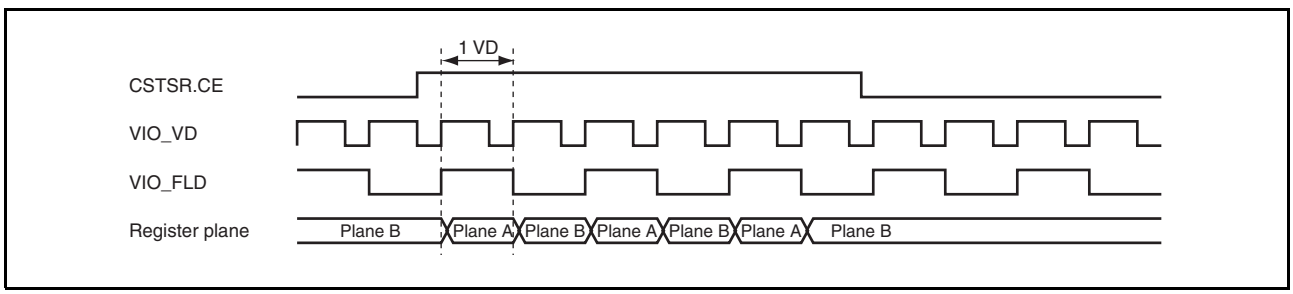


Figure 46.31 Timing for Register Plane Switching when RVS Bit is B'1

46.4.9 CEU Register Forcible Control Register (CRCMPR)

CRCMPR forcibly controls switching of the planes of registers with a 2-plane configuration. Setting this register enables direct control of register plane switching.

Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RA	0	R/W	Indicates the register plane currently specified. This register value automatically changes in synchronization with VD for starting capture. To start capture with plane A of the register when a setting to switch the register plane in synchronization with VD has been made (RC bit in CRCNTR is 1), specify plane B of the register using this bit. 0: Specifies plane A of the register 1: Specifies plane B of the register

46.4.10 Capture Filter Control Register (CFLCR)

CFLCR sets the scale-down factor for the filter to scale images down.

The CEU has an image scale-down filter which can be used to scale down the captured images before storing them in the memory. Set CFLCR to 0 when not performing scale-down (same size output). If a value other than 0 is set in CFLCR, scale-down is performed. In data fetch mode, set CFLCR to 0.

When handling an interlace source image as a frame image, set CFLCR to 0 not to use the filter.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VMANT[3:0]				VFRAC[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HMANT[3:0]				HFRAC[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	VMANT[3:0]	H'0	R/W	Mantissa Part of Vertical Scale-Down Factor The specifiable range is H'0 to H'F. When H'0 is set for the VMANT bits and H'000 is set for the VFRAC bits, the scale-down filter is not used.
27 to 19 18 to 16	VFRAC[11:3] VFRAC[2:0]	H'000	R/W R	Fraction Part of Vertical Scale-Down Factor The specifiable range is H'000 to H'FF8. The fraction of the scale-down factor that cannot be set with only the VMANT bits must be set with these bits.
15 to 12	HMANT[3:0]	H'0	R/W	Mantissa Part of Horizontal Scale-Down Factor The specifiable range is H'0 to H'F. When H'0 is set for the HMANT bits and H'000 is set for the HFRAC bits, the scale-down filter is not used.
11 to 3 2 to 0	HFRAC[11:3] HFRAC[2:0]	H'000	R/W R	Fraction Part of Horizontal Scale-Down Factor The specifiable range is H'000 to H'FF8. The fraction of the scale-down factor that cannot be set with only the HMANT bits must be specified with these bits.

An image scale-down filter is installed in the CEU, and the captured images can be scaled down and stored in the memory.

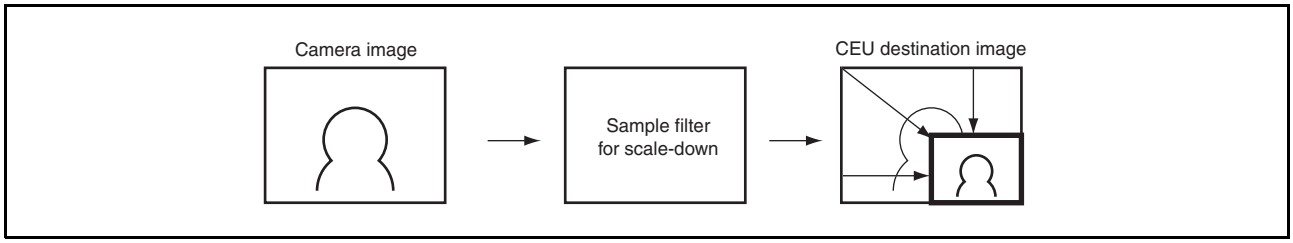


Figure 46.32 Scale-Down of Captured Image

The formulas for obtaining the MANT (VMANT or HMANT) and FRAC (VFRAC or HFRAC) values from the input pixel count and output pixel count of the filter are shown below. Set the MANT and FRAC bits in order to obtain the desired output pixel count from the number of pixels input to the CEU.

First, calculate preliminary MANT and FRAC values. The parameters needed for calculation are defined as follows:

$$\alpha = MANT \times 4096 + FRAC \quad \text{... Formula 1}$$

$$SCL \text{ (scaling factor)} = \frac{4096}{\alpha} \quad \text{... Formula 2}$$

Assuming an operator $\lfloor x \rfloor$ which discards fractions of an integer x, the MANT and FRAC values can be temporarily set as follows, according to formula 1 and formula 2.

$$MANT = \left\lfloor \frac{1}{SCL} \right\rfloor, \quad FRAC = \left\lfloor 512 \times \left(\frac{1}{SCL} - MANT \right) \right\rfloor \times 8$$

Here, the scaled-down filter output size ($SIZE_D$) can be calculated using the input image size S_{in} (8-bit interface: half of the CAPWR setting, 16-bit interface: CAPWR setting) in the following formula.

$$SIZE_D = \left\lfloor 1 + \left(\left\lfloor \frac{1}{2} + \frac{S_{in} - 1}{MANT_{pre}} \right\rfloor - 1 \right) \times \frac{MANT_{pre} \times 4096}{\alpha} \right\rfloor \quad \text{... Formula 3}$$

$$\begin{bmatrix} MANT_{pre} = 1 \longrightarrow (0 \le MANT < 2) \\ MANT_{pre} = 2 \longrightarrow (2 \le MANT < 4) \\ MANT_{pre} = 4 \longrightarrow (4 \le MANT < 8) \\ MANT_{pre} = 8 \longrightarrow (8 \le MANT\ ) \end{bmatrix}$$

The number of output pixels can be obtained by substituting the temporarily calculated MANT, FRAC, and input image size into these formulas. If the calculated number of output pixels is smaller than the number of output pixels used to obtain the preliminary MANT and FRAC values, recalculate with a smaller FRAC (α) value, and set the MANT and FRAC values in this register so that a pixel value greater than the desired number of output pixels can be obtained.

Example: Scale down 640 pixels to 480 pixels

SCL = 480/640 = 3/4, and the preliminary settings of MANT = 1, $MANT_{pre}$ = 1, and FRAC = H'550 are made. Substituting these in the following formula results in an output pixel count of 479.

$$SIZE_D = \left\lfloor 1 + \left(\left\lfloor \frac{1}{2} + \frac{S_{in} - 1}{MANT_{pre}} \right\rfloor - 1 \right) \times \frac{MANT_{pre} \times 4096}{\alpha} \right\rfloor \quad \text{... Formula 3}$$

Since this output pixel count is smaller than the desired output pixel count of 480, the formula is recalculated with a FRAC value of H'548, a value eight less than the previous time. The obtained result of output pixel count = 480 is equal to the desired output pixel count of 480, so this register is set as MANT = 1 and FRAC = H'548.

Table 46.8 Setting Examples for Each Scale-Down Filter Factor

Scale-Down Factor	FRAC		MANT	Input Pixel Count	Output Pixel Count	Clipping Size (CFSZR)
	Decimal	Hexadecimal				
7/8	576	H'240	1	640	560	560
3/4	1352	H'548	1	640	480	480
5/8	2448	H'990	1	640	400	400
1/2	0	H'0	2	640	320	320
3/8	2728	H'AA8	2	640	240	240
1/3	0.0	H'0	3	640	213	212
1/4	0.0	H'0	4	640	160	160
1/5	0.0	H'0	5	640	128	128
1/6	0.0	H'0	6	640	107	104
1/7	0.0	H'0	7	640	91	88
1/8	0.0	H'0	8	640	80	80
1/16	4088	H'FF8	15	640	40	40

Note: This scale-down filter uses a VGA-size line memory for scale-down. Therefore, when an image larger than the VGA size is input for scale-down, settings must be made so that the output image size is equal to or larger than the SubQCIF size and equal to or smaller than the VGA size. When an image is not scaled down (same size output), this restriction does not apply.

46.4.11 Capture Filter Size Clip Register (CFSZR)

CFSZR sets the clipping size for fine adjustment of the image size output from the filter, and must be set in combination with CFLCR. When clipping the output size of the filter, set the clipping size as a number of pixels, and the setting unit should be four pixels. CFSZR must be set even when scale-down is not performed (same size output).

In data synchronous fetch mode, set CFSZR according to the setting of CAPWR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VFCLP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	HFCLP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 18 17, 16	VFCLP[11:2] VFCLP[1:0]	H'000	R/W R	These bits set the vertical clipping value of the filter output size (4-pixel units).
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 2 1, 0	HFCLP[11:2] HFCLP[1:0]	H'000	R/W R	These bits specify the horizontal clipping value of the filter output size (4-pixel units).

The scale-down filter in the CEU may output an odd number of pixels or lines depending on the settings. To adjust the output size of the filter, the CEU clips the destination image by using the number of pixels specified in CFSZR, as shown in Figure 46.33. The clipping size must be specified vertically and horizontally in 4-pixel units.

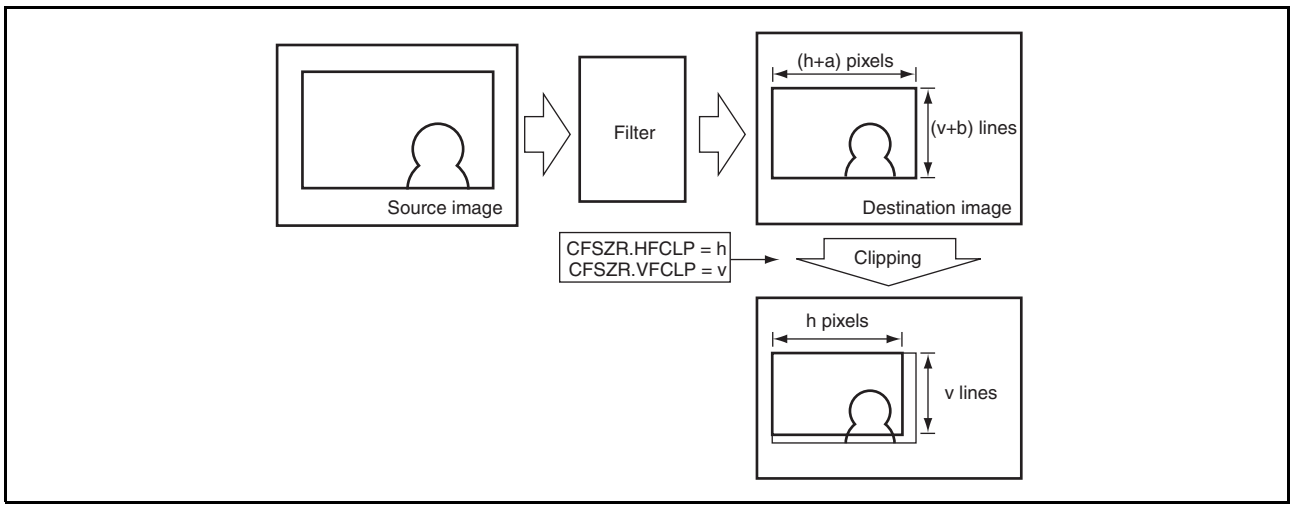


Figure 46.33 Clipping of Image Output from Filter

The pixels to be clipped are counted from the top-left corner of a display. The pixels located to the right of the specified number of pixels or below the specified number of lines are discarded by the clipping function. If the number of pixels specified in CFSZR is larger than that output from the filter, correct operation cannot be guaranteed. To avoid this, the clipping size specified in CFSZR must be equal to or smaller than the number of pixels output from the filter.

Note: In data synchronous fetch mode, the following settings are required. Data cannot be fetched correctly unless the following settings are made.

· 8-bit interface:
  VFCLP = CAPWR.VWDTH
  HFCLP = CAPWR.HWDTH/2

· 16-bit interface:
  VFCLP = CAPWR.VWDTH
  HFCLP = CAPWR.HWDTH

46.4.12 Capture Destination Width Register (CDWDR)

CDWDR sets the horizontal image size in the memory area where the captured image is to be output in 4-byte units (4-pixel units).

In data synchronous fetch mode, set CDWDR according to the setting of CAPWR.

This register is not used, during data enable fetch mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CHDW[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 2 1, 0	CHDW[12:2] CHDW[1:0]	H'0000	R/W R	These bits specify the horizontal image size in the memory area where the captured image is to be stored (4-byte units). The image data captured by the CEU is stored in the memory. If the right end of the captured image does not match the horizontal image size in the memory area as shown in Figure 46.34, some addresses must be skipped at the right end of the image when storing the captured image. Therefore, the horizontal image size in the memory area where the captured image is to be stored must be set in these bits. The maximum value to be set is 8188 bytes (8188 pixels). In data synchronous fetch mode, set as follows: 8-bit interface: CHDW = CAPWR.HWDTH 16-bit interface: CHDW = CAPWR.HWDTH × 2

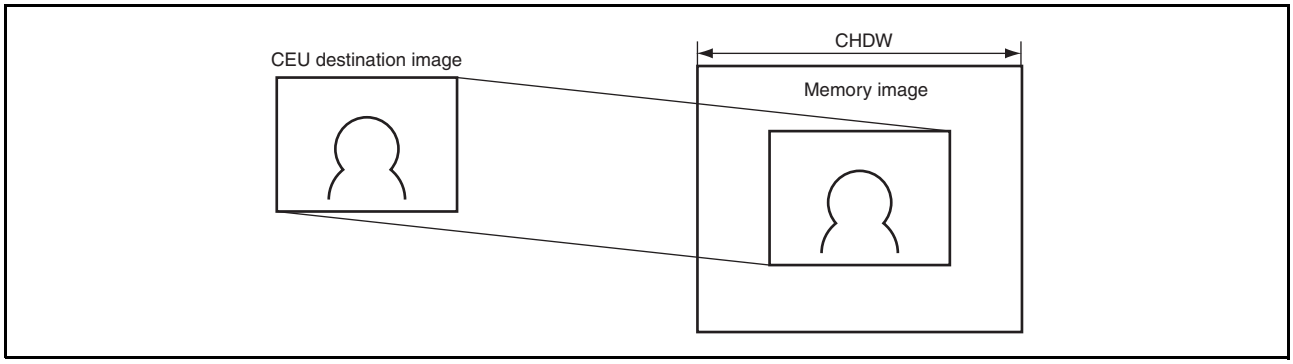


Figure 46.34 Captured Image and Memory Area Image

46.4.13 Capture Data Address Y Register (CDAYR)

CDAYR specifies the address where the luminance (Y) component of the captured data is to be stored in frame image capture or one-field image capture, the address where the luminance (Y) component of the captured top field is to be stored in both-field image capture, and the address where the fetched data is to be stored in data fetch. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. In frame image capture or one-field image capture, set the start address of the memory area where the Y (luminance) component of the captured data is to be stored by CDAYR. In both-field image capture, set the start address of the memory area where the Y (luminance) component of the captured top-field image is to be stored by CDAYR. In data fetch, set the start address of the memory area where data is to be stored by CDAYR.

Because the address must be specified in 32 bits, the address set by CDAYR must be in longword units. As the setting is in 4-pixel units for image capture and 4-byte units for data fetch, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAYR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAYR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CAYR[31:2]	H'0000 0000	R/W	• Frame image capture: These bits set the address for storing the Y (luminance) component data of the captured data (4-pixel units). • One-field image capture: These bits set the address for storing the Y (luminance) component data of the captured data (4-pixel units). • Both-field image capture: These bits set the address for storing the Y (luminance) component data of the captured top-field data (4-pixel units). • Data fetch: These bits set the address for storing data (4-byte units). • Data enable fetch bundle write: These bits set the address for storing data (32-byte units).
1, 0	CAYR[1:0]		R	

Set the address of the starting point of the memory area where the fetched data is to be stored in this register, as shown in Figure 46.35.

- Frame image capture: Set the address of the starting point of the memory area where the Y component of the captured image is to be stored.
- One-field image capture: Set the address of the starting point of the memory area where the Y component of the captured image is to be stored.
- Both-field image capture: Set the address of the starting point of the memory area where the Y component of the captured top-field image is to be stored.
- Data fetch: Set the address of the starting point of the memory area where the fetched data is to be stored. In data fetch mode, the data is simply stuffed in order from the start address so the end address becomes as follows:

  End address = CDAYR + number of fetched bytes
- Data enable fetch bundle write: Set the address in 32-byte units.

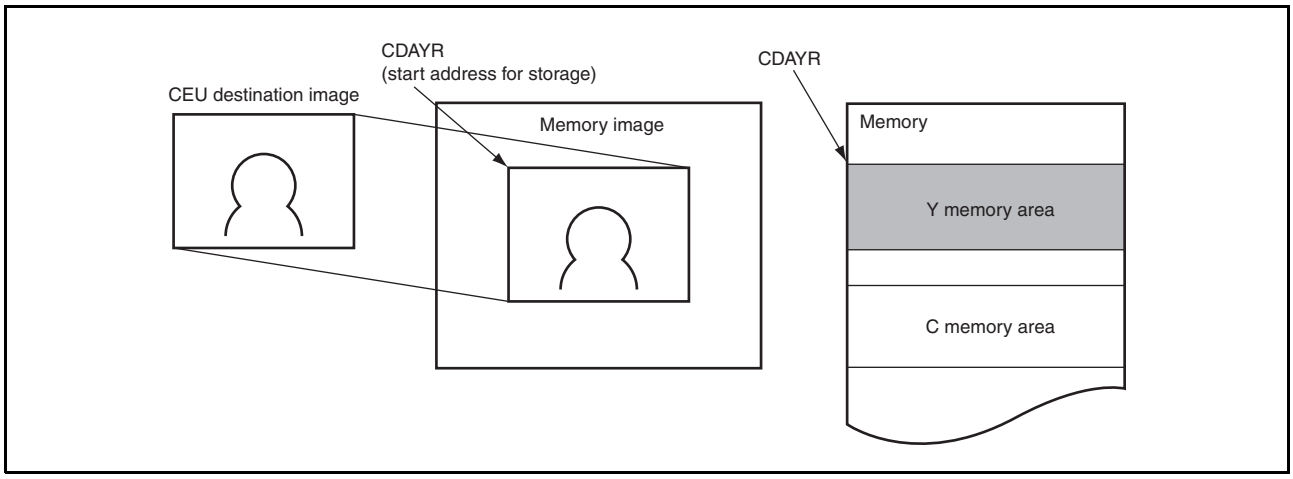


Figure 46.35 Relationship between Captured Image and Y Component Memory Area

46.4.14 Capture Data Address C Register (CDACR)

CDACR specifies the address where the chrominance (C) component of the captured data is to be stored in frame image capture or one-field image capture, and the address where the chrominance (C) component of the captured top field is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. In frame image capture or one-field image capture, set the start address of the memory area where the C (chrominance) component of the captured data is to be stored by CDACR. In both-field image capture, set the start address of the memory area where the C (chrominance) component of the captured top-field image is to be stored by CDACR. CDACR is not used in data fetch.

Because the address must be specified in 32 bits, the address set by CDACR must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CACR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CACR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CACR[31:2]	H'0000 0000	R/W	• Frame image capture: These bits set the address for storing the C (chrominance) component data of the captured data (4-pixel units). • One-field image capture: These bits set the address for storing the C (chrominance) component data of the captured data (4-pixel units). • Both-field image capture: These bits set the address for storing the C (chrominance) component data of the captured top-field data (4-pixel units).
1, 0	CACR[1:0]		R	

Set the address of the starting point of the memory area where the C component of the captured image is to be stored in this register, as shown in Figure 46.36. The C component has an output data format like that in Figure 46.37, and is saved in the memory in this format.

- Frame image capture: Set the address of the starting point of the memory area where the C component of the captured image is to be stored.
- One-field image capture: Set the address of the starting point of the memory area where the C component of the captured image is to be stored.
- Both-field image capture: Set the address of the starting point of the memory area where the C component of the captured top-field image is to be stored.

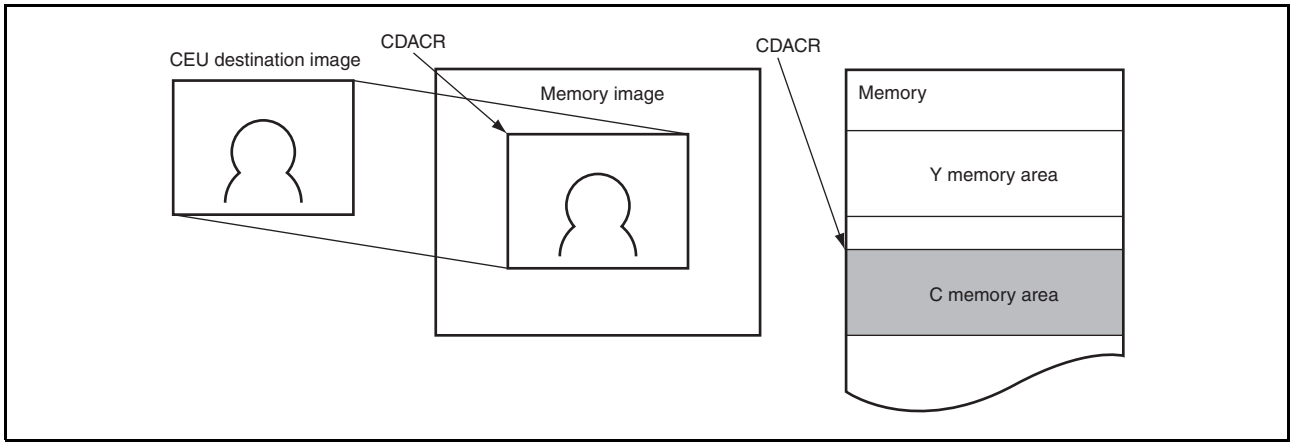


Figure 46.36 Relationship between Captured Image and C Component Memory Area

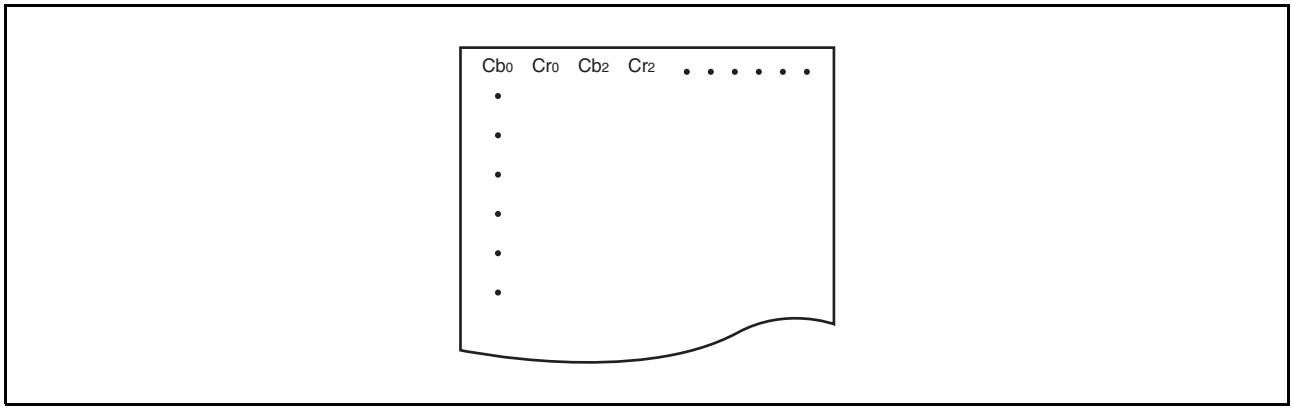


Figure 46.37 Image of Storing C Components in Memory

46.4.15 Capture Data Bottom-Field Address Y Register (CDBYR)

CDBYR specifies the address where the luminance (Y) component of the captured bottom-field data is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. Set the start address of the memory area where the Y (luminance) component of the captured bottom-field image is to be stored by CDBYR. CDBYR is not used in frame image capture, one-field image capture, or data fetch.

Because the address must be specified in 32 bits, the address set by CDBYR must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CBYR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CBYR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CBYR[31:2]	H'0000 0000	R/W	These bits set the address for storing the Y (luminance) component data of the captured bottom-field data (4-pixel units).
1, 0	CBYR[1:0]		R	

Set the address of the starting point of the memory area where the Y component of the captured bottom-field image is to be stored in this register, as shown in Figure 46.38.

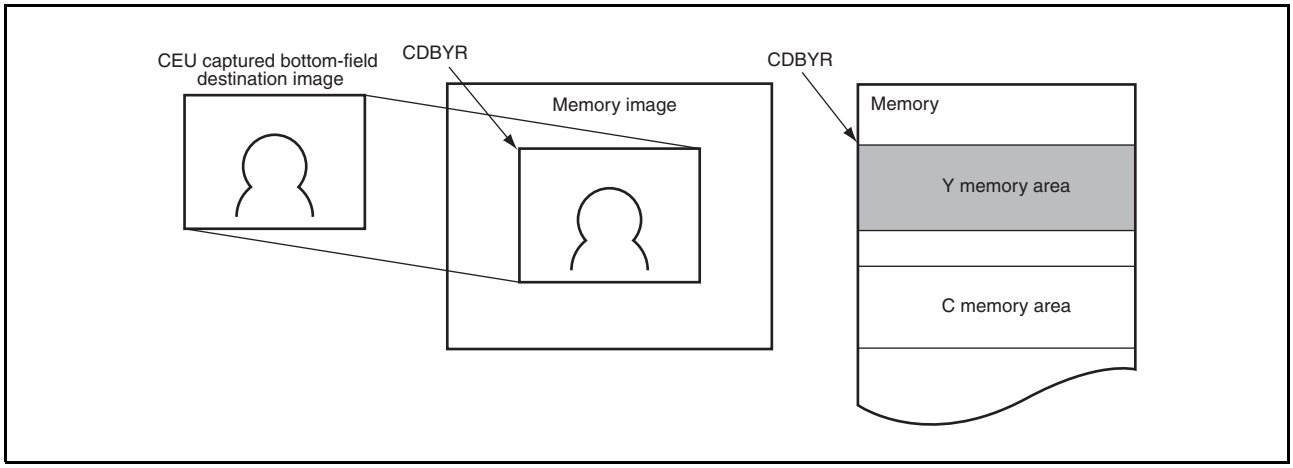


Figure 46.38 Relationship between Captured Bottom-Field Image and Y Component Memory Area

46.4.16 Capture Data Bottom-Field Address C Register (CDBCR)

CDBCR specifies the address where the chrominance (C) component of the captured bottom-field data is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. Set the start address of the memory area where the C (chrominance) component of the captured bottom-field image is to be stored by CDBCR. CDBCR is not used in frame image capture, one-field image capture, or data fetch.

Because the address must be specified in 32 bits, the address set by CDBCR must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CBCR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CBCR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CBCR[31:2]	H'0000 0000	R/W	These bits set the address for storing the C (chrominance) component data of the captured bottom-field data (4-pixel units).
1, 0	CBCR[1:0]		R	

Set the address of the starting point of the memory area where the C component of the captured bottom-field image is to be stored in this register, as shown in Figure 46.39. The C component has an output data format like that in Figure 46.40, and is saved in the memory in this format.

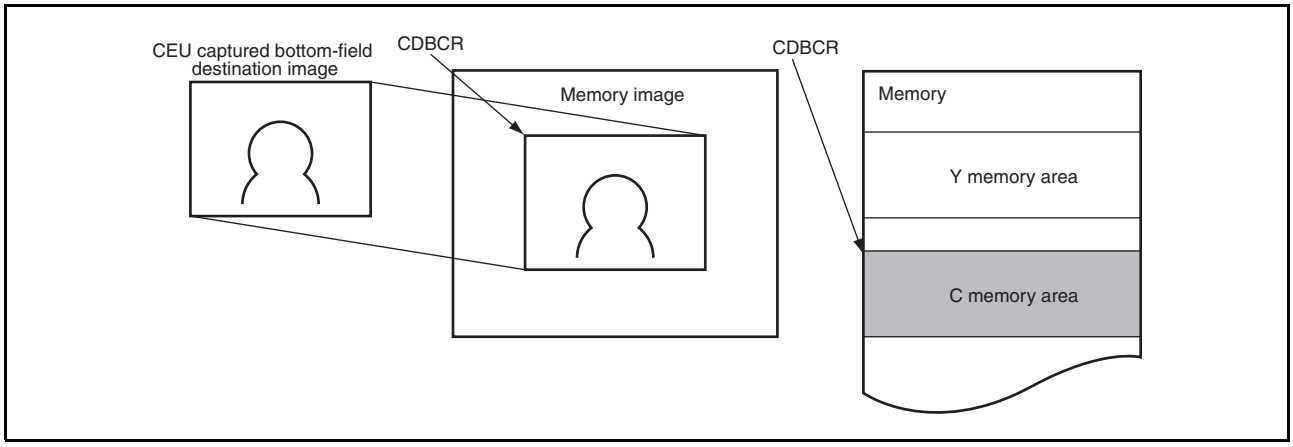


Figure 46.39 Relationship between Captured Bottom-Field Image and C Component Memory Area

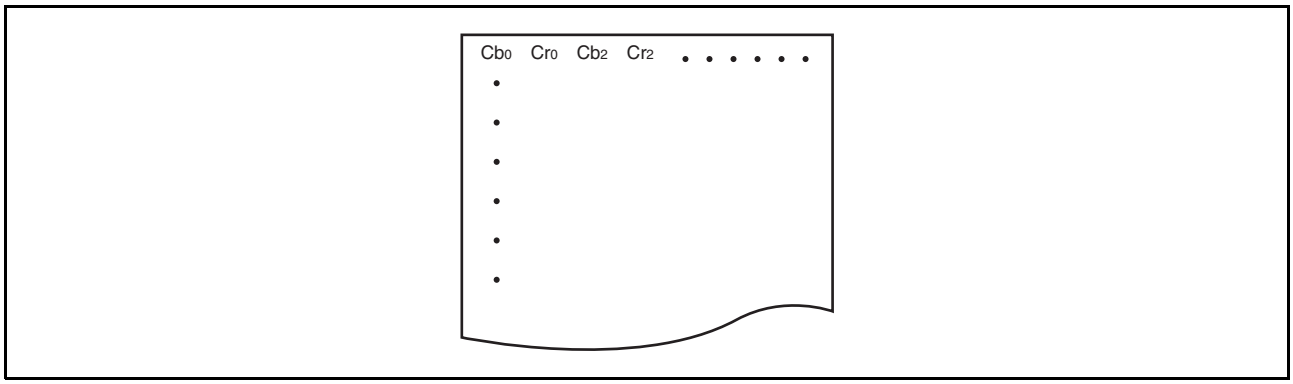


Figure 46.40 Image of Storing C Components in Memory

46.4.17 Capture Bundle Destination Size Register (CBDSR)

CBDSR sets the size of output to memory in a bundle write. The number of output lines should be specified for image capture or data synchronous fetch. The number of bytes should be specified for data enable fetch.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	CBVS[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CBVS[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	CBVS[22:3] CBVS[2:0]	H'000	R/W R	These bits select the number of lines or number of bytes for output to the memory in a bundle write. Image capture and data synchronous fetch: Number of lines for output to the memory in a bundle write. Unit: 8 lines, min.: 8 lines, max.: 1,920 lines (H'780) Data enable fetch: Number of bytes for output to the memory in a bundle write. Unit: 32 bytes, min.: 512 bytes, max.: 6291456 lines (H'600000)

(a) Image capture and data synchronous fetch

Set the number of lines of captured data to be written to the memory by a bundle write as a multiple of eight. This register is valid only when the CBE bit in CDOCR is 1. When the CBE bit in CDOCR is 1 and this register cleared to H'0, this module operates with the number of lines of captured data to be written to the memory as eight. The maximum number of lines that can be set is 1,920 (H'780). Only bits CBVS[11:3] are valid.

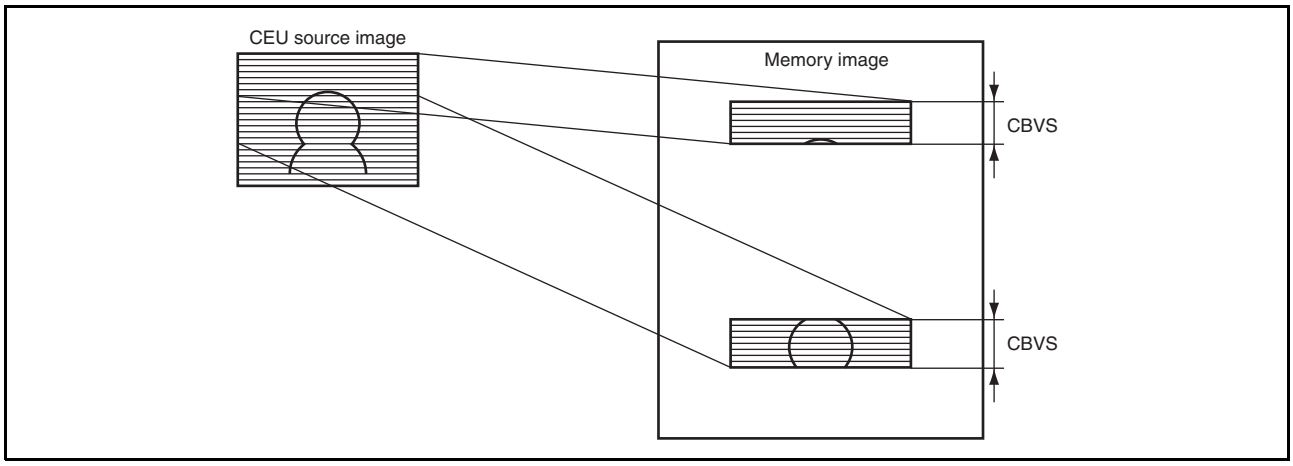


Figure 46.41 Image of Storing Captured Image in Memory by Bundle Write

(b) Data enable fetch

Set the number of bytes of captured data to be written to the memory by a bundle write as a multiple of 32. This register is valid only when the CBE bit in CDOCR is 1. The minimum settable size is 512 bytes. When a number smaller than 512 bytes is specified, operation is not guaranteed.

46.4.18 Capture Low-Pass Filter Control Register (CLFCR)

CLFCR specifies whether or not to operate the low-pass filter. In data fetch mode, clear the LPF bit to B'0.

The characteristic of the low-pass filter installed in the CEU causes the phase location of the image processed by the low-pass filter to be shifted right by one pixel compared to the raw image.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LPF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	LPF	0	R/W	Enables or disables operation of the low-pass filter. The low-pass filter removes high-frequency components from the destination image in the horizontal direction. Clear this bit to 0 in data fetch mode. 0: Low-pass filter not used 1: Low-pass filter used (only in the horizontal direction)

46.4.19 Firewall Operation Control Register (CFWCR)

CFWCR specifies the upper limit of the write addresses in data enable fetch. When the VD input from an external module dose not go low and end notification is not given, this register can prevent writing to memory from being out of control.

This register is enabled only in data enable fetch.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FWV[26:11]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FWV[10:0]											—	—	—	—	FWE
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	FWV[26:0]	H'0000008	R/W	These bits specify the upper limit of a write address. Specify the upper 27 bits of the 32-bit address. The upper limit of an address is FWV[26:0]<<5 + H'1F.
4 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	FWE	0	R/W	With the setting of FWE = 1, when an address exceeds the value set with FWV, the address is retained and an interrupt source FWF is set. After this, the address is not incremented and data is overwritten on the upper limit address. 0: Firewall is not activated. 1: Firewall is activated.

46.4.20 Capture Data Output Control Register (CDOCR)

CDOCR sets the format for outputting captured data to the memory. In data fetch mode, set the CDS bit to B'1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CBE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CDS	—	COLS	COWS	COBS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CBE	0	R/W	Controls the number of lines of captured data to be written to the memory. • Image capture This bit controls the number of lines of captured data to be written to the memory. When bundle write is set by this register, captured data is written in line units specified by CBDSR to the addresses specified by CDAYR and CDACR, and CDAYR2 and CDACR2 (CDBYR and CDBCR, and CDBYR2 and CDBCR2 for the bottom field in both-field capture) alternately (Figure 46.42). When captured data has been written for the number of lines set by CBDSR, a write end interrupt corresponding to each address setting register occurs. However, after write for one-frame (one-field) capture ends, a bundle write end interrupt does not occur even when bundle write has finished. • Data synchronous fetch This bit controls the number of lines of captured data to be written to the memory. When bundle write is set by this register, captured data is written in line units specified by CBDSR to the addresses specified by CDAYR and CDAYR2 alternately. When captured data has been written for the number of lines set by CBDSR, a write end interrupt corresponding to each address setting register occurs. However, after write for one-frame capture ends, a bundle write end interrupt does not occur even when bundle write has finished. • Data enable fetch This bit controls the number of bytes of captured data to be written to the memory. When bundle write is set by this register, captured data is written in byte units specified by CBDSR to the addresses specified by CDAYR and CDAYR2 alternately. When captured data has been written for the number of bytes set by CBDSR, a write end interrupt corresponding to each address setting register occurs. Also, only in data enable fetch, a bundle write end interrupt occurs when bundle write has finished after write for one-frame capture ends. Table 46.9 shows the correspondence between address setting registers and write end interrupt sources. Figure 46.43 shows the timing of write end interrupts in image capture and data synchronous fetch. Figure 46.44 shows the timing of write end interrupts in data enable fetch. 0: Normal write 1: Bundle write
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	CDS	0	R/W	Sets the image format when outputting the image data captured in the YCbCr422 format to the memory. When 0 is written to this bit, only the luminance component (Y) is output and no chrominance components (Cb and Cr) are output for the odd-numbered lines. With an interlace source image, similarly only the luminance component (Y) is output and no chrominance components (Cb and Cr) are output for the odd-numbered lines of the field. In data fetch mode, set this bit to 1. 0: Converts the YCbCr422 format to the YCbCr420 format before outputting data to the memory 1: Outputs data in the YCbCr422 format to the memory without conversion

Bit	Bit Name	Initial Value	R/W	Description
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	COLS	0	R/W	Controls swapping in 32-bit units for data output from the CEU. 0: Data is not swapped in 32-bit units 1: Data is swapped in 32-bit units
1	COWS	0	R/W	Controls swapping in 16-bit units for data output from the CEU. 0: Data is not swapped in 16-bit units 1: Data is swapped in 16-bit units
0	COBS	0	R/W	Controls swapping in 8-bit units for data output from the CEU. 0: Data is not swapped in 8-bit units 1: Data is swapped in 8-bit units

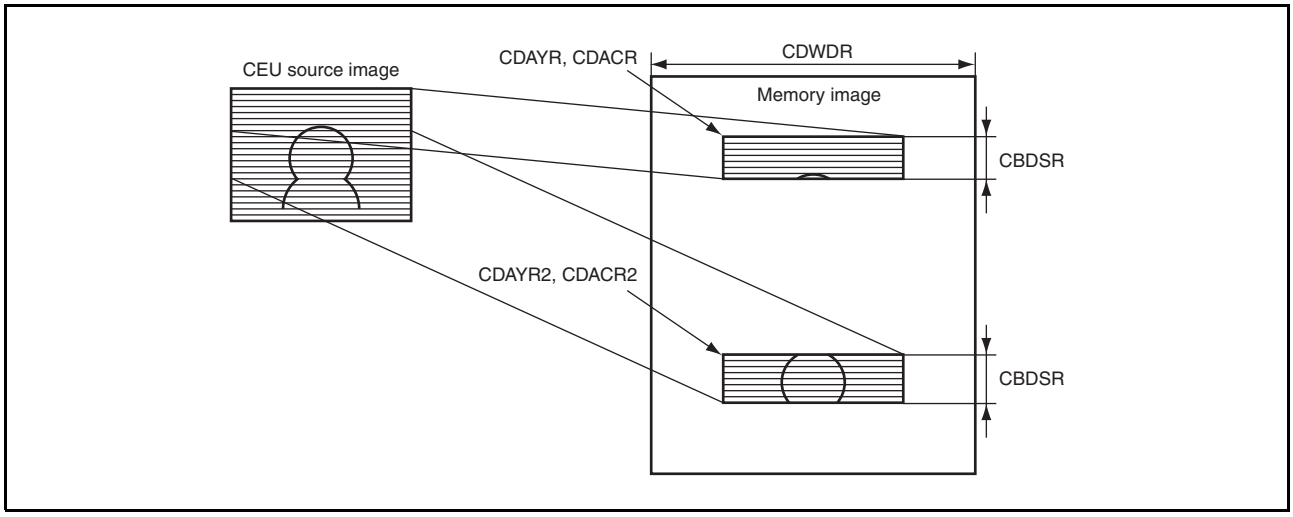


Figure 46.42 Image of Bundle Write to Memory

Table 46.9 Correspondence between Address Setting Registers and Write End Interrupt Sources

Address Setting Registers	Bundle Write End Interrupt Source
CDAYR, CDACR	CPBE1 bit in CETCR
CDAYR2, CDACR2	CPBE2 bit in CETCR
CDBYR, CDBCR	CPBE3 bit in CETCR
CDBYR2, CDBCR2	CPBE4 bit in CETCR

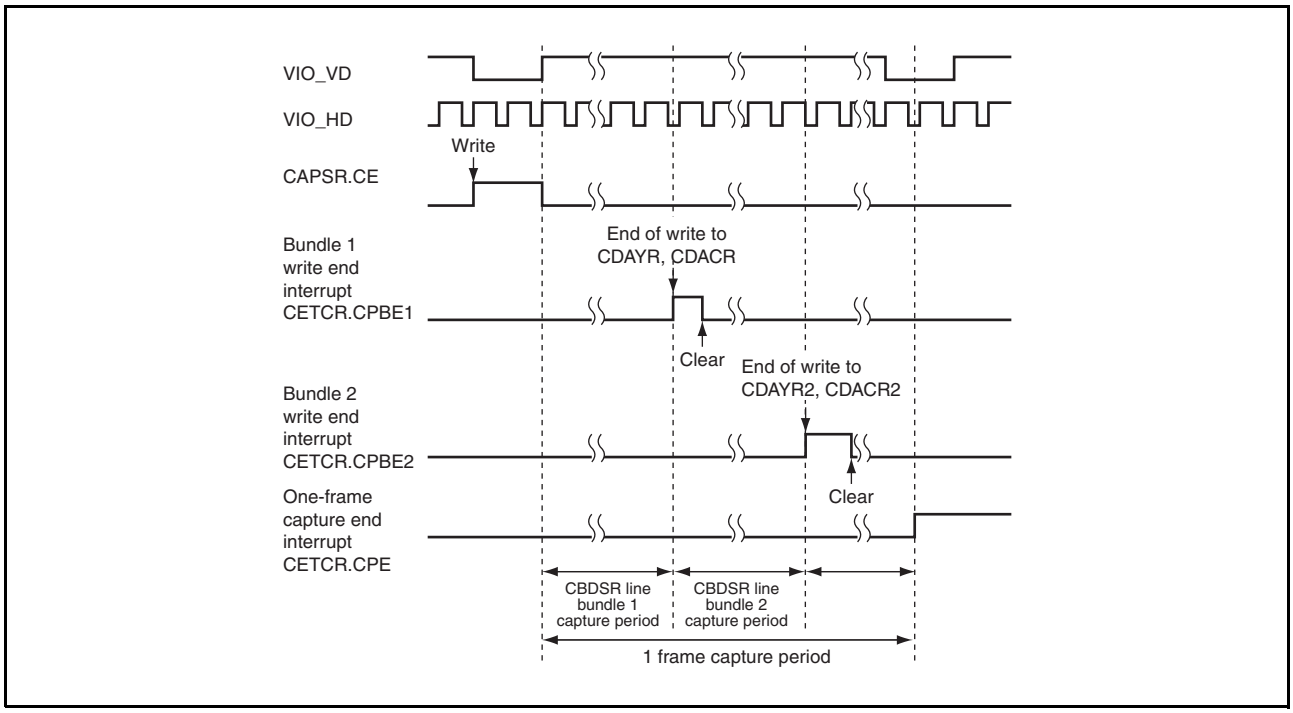


Figure 46.43 Timing of Write End Interrupts (Image Capture, Data Synchronous Fetch)

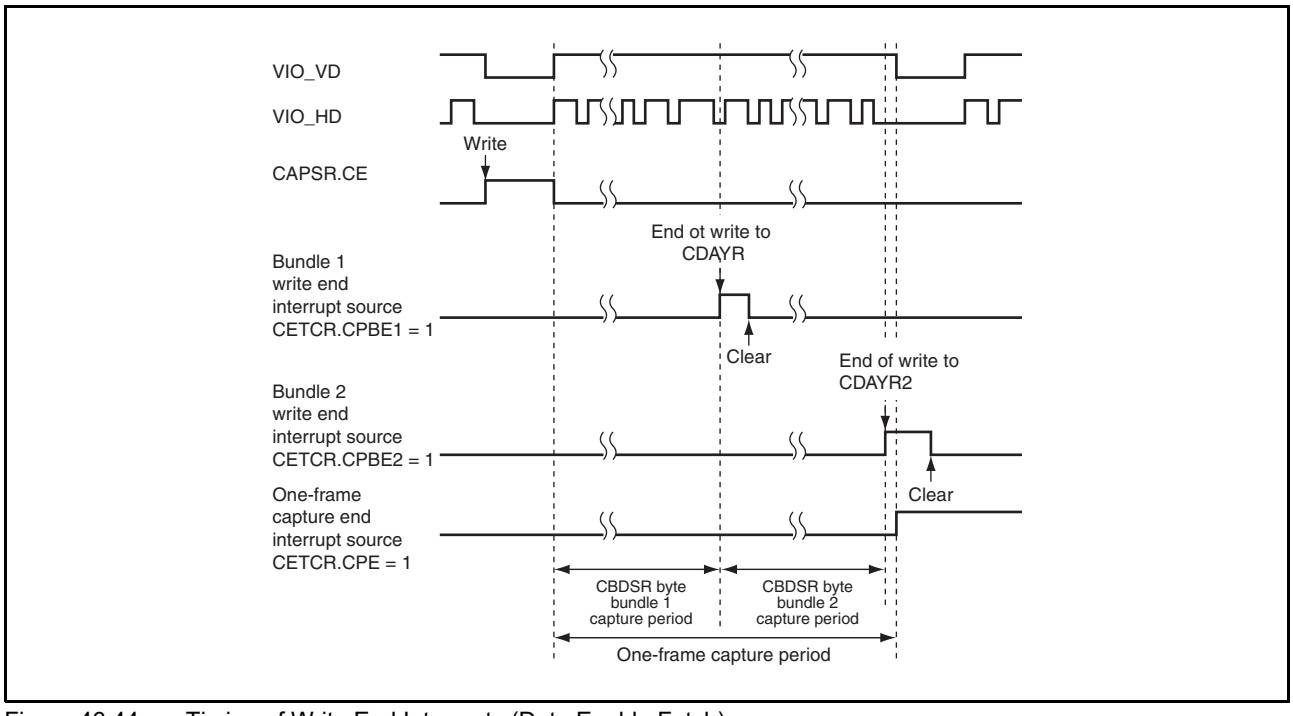


Figure 46.44 Timing of Write End Interrupts (Data Enable Fetch)

For data output from the CEU, the COLS, COWS, and COBS bits control swapping in 32-bit, 16-bit, and 8-bit units, respectively. Set these bits when data is misaligned because of a difference in endian. The data swapping bits are shown below. These bits can be set similarly in data fetch mode.

Data can be swapped in 8-bit, 16-bit, 32-bit units, or in 32 bits, 16 bits and 8 bits, as shown in Figure 46.45. To enable data swapping, set the corresponding control bit to B'1.

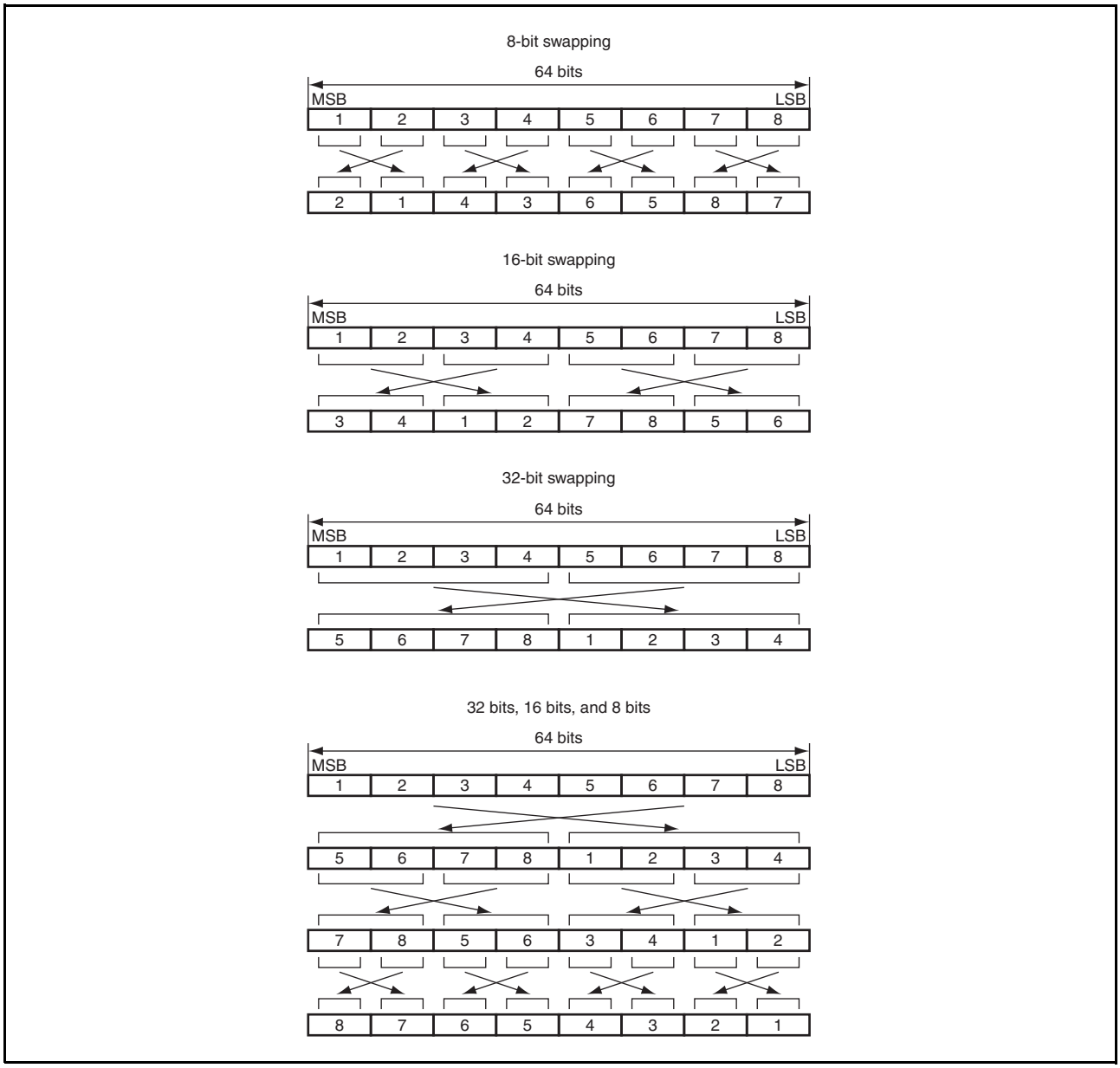


Figure 46.45 Data Swapping by Data Aligner

46.4.21 Capture Event Interrupt Enable Register (CEIER)

CEIER enables or disables interrupts of the event flag register that generates CEU interrupts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	NVDIE	NHDIE	FWFIE	—	—	VBPIE	—	IGVSIE	IGHSIE	CDTOFIE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CPBE4IE	CPBE3IE	CPBE2IE	CPBE1IE	—	—	VDIE	HDIE	—	—	—	IGRWIE	—	—	CFEIE	CPEIE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	NVDIE	0	R/W	Non-VD Interrupt Enable Disable this interrupt (NVDIE = 0) for data enable fetch. 0: Disables a non-VD interrupt 1: Enables a non-VD interrupt
24	NHDIE	0	R/W	Non-HD Interrupt Enable Disable this interrupt (NHDIE = 0) for data enable fetch. 0: Disables a non-HD interrupt 1: Enables a non-HD interrupt
23	FWFIE	0	R/W	FWF Interrupt Enable 0: Disables a FWF interrupt 1: Enables a FWF interrupt
22, 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	VBPIE	0	R/W	VBP Interrupt Enable 0: Disables a VBP interrupt 1: Enables a VBP interrupt
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	IGVSIE	0	R/W	IGVS Interrupt Enable 0: Disables an IGVS interrupt 1: Enables an IGVS interrupt
17	IGHSIE	0	R/W	IGHS Interrupt Enable 0: Disables an IGHS interrupt 1: Enables an IGHS interrupt
16	CDTOFIE	0	R/W	CDTOF Interrupt Enable 0: Disables a CDTOF interrupt 1: Enables a CDTOF interrupt
15	CPBE4IE	0	R/W	CPBE4 Interrupt Enable 0: Disables a CPBE4 interrupt 1: Enables a CPBE4 interrupt
14	CPBE3IE	0	R/W	CPBE3 Interrupt Enable 0: Disables a CPBE3 interrupt 1: Enables a CPBE3 interrupt
13	CPBE2IE	0	R/W	CPBE2 Interrupt Enable 0: Disables a CPBE2 interrupt 1: Enables a CPBE2 interrupt
12	CPBE1IE	0	R/W	CPBE1 Interrupt Enable 0: Disables a CPBE1 interrupt 1: Enables a CPBE1 interrupt
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9	VDIE	0	R/W	VD Interrupt Enable 0: Disables a VD interrupt 1: Enables a VD interrupt
8	HDIE	0	R/W	HD Interrupt Enable 0: Disables an HD interrupt 1: Enables an HD interrupt
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	IGRWIE	0	R/W	Register-Access-During-Capture Interrupt Enable 0: Disables a register-access-during-capture interrupt 1: Enables a register-access-during-capture interrupt
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	CFEIE	0	R/W	CFE Interrupt Enable 0: Disables a CFE interrupt 1: Enables a CFE interrupt
0	CPEIE	0	R/W	One-Frame Capture End Interrupt Enable 0: Disables a one-frame capture end interrupt 1: Enables a one-frame capture end interrupt

46.4.22 Capture Event Flag Clear Register (CETCR)

CETCR notifies the CPU of the source of an interrupt that is generated in the CEU. The flags of CETCR can be used as interrupt signals. When the corresponding interrupt is enabled, an interrupt is generated. To clear the interrupt, clear the bit corresponding to the interrupt source to 0. After several cycles have passed after modifying the bit, the interrupt is cleared.

To clear the bit corresponding to the interrupt source to be cleared to 0 and retain that state, write 1 to that bit. For example, to clear only the CPE bit to 0, write H'FFFF FFFE to CETCR.

In CETCR, only bits to which 0 is written are cleared. Bits to which 1 is written retain their current values. To clear an interrupt source, write 0 only to the bit corresponding to the interrupt source to be cleared, and 1 to the other bits.

Note: Since the CETCR value becomes undefined in the following cases, clear all bits in CETCR to 0.

- VD and HD bits immediately after power-on reset or the deep-standby mode is entered
- All bits after the software standby or module standby mode is entered
- VD and HD bits after the polarities of the capture interface sync signals are changed

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	NVD	NHD	FWF	—	—	VBP	—	IGVS	IGHS	CDTOF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CPBE4	CPBE3	CPBE2	CPBE1	—	—	VD	HD	—	—	—	IGRW	—	—	CFE	CPE
Initial value:	0	0	0	0	0	0	—	—	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	NVD	0	R/W	This bit is used for an interrupt indicating that no VD was input. A non-VD interrupt occurs when the 14-bit internal counter becomes full. Accordingly, this bit is set to 1 when no VD has been input for at least 16,383 lines since the last VD was input.
24	NHD	0	R/W	This bit is used for an interrupt indicating that no HD was input. The timing for a non-HD interrupt to occur differs depending on the bit width of the digital image input pins. 8-bit digital image input pins: Occurs when the 11-bit internal counter that is incremented every eight cycles becomes full. Accordingly, this bit is set to 1 when no HD has been input for at least 16,376 cycles since the last HD was input. 16-bit digital image input pins: Occurs when the 12-bit internal counter that is incremented every four cycles becomes full. Accordingly, this bit is set to 1 when no HD has been input for at least 16,380 cycles since the last HD was input. When connecting a camera whose HD is fixed low when VD is low, this bit may be set to 1. Ignore this interrupt during data enable fetch.
23	FWF	0	R/W	The interrupt is generated when data is written to the address that exceeds the value specified with CFWCR.FMV. This bit is set to 1 when data is written to the address that exceeds the value specified with CFWCR.FMV while CFWCR.FWE = 1.
22, 21	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
20	VBP	0	R/W	This bit is used for an interrupt indicating that VD has been input while the CEU holds data (insufficient vertical-sync front porch). The conditions for a VBP interrupt to occur are as follows: • Condition 1 VD is input when there is captured data within the CEU • Condition 2 The last transfer data cannot be internally detected due to a write buffer overflow or an illegal HD so that the end timing is unclear until the next VD (By generating a VBP interrupt at the VD input timing, capture fail can be announced.) When a VBP interrupt occurs, a capture end interrupt (CPE bit in CETCR) does not occur and the image of that frame is not captured correctly. Though a capture end interrupt (CPE bit) will occur on rare occasions, it should be ignored in this case. Capturing cannot be performed until the next VD (even if the CE bit (capture reservation signal) in CAPSR is 1, capturing does not start). In the case of condition 2, instead of waiting for a VBP interrupt to occur, execute a software reset (CPKIL bit in CAPSR) to stop capturing and then restart capturing. In this case, since capture operation is terminated without waiting for the next VD, a VBP interrupt does not occur and capturing can be performed from the next VD.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	IGVS	0	R/W	This bit is used for an interrupt generated when the number of VD cycles set in CMCYR differ from the number of VD cycles input from an external module. This bit is set to 1 when there is an illegal VD input from an external module. This bit is set to 1 when the number of HD cycles for the VD input to the CEU differs from the value set in the VCYL bits in CMCYR. Note however that when the VCYL bits are cleared to 0, this interrupt does not occur.
17	IGHS	0	R/W	This bit is used for an interrupt generated when the number of HD cycles set in CMCYR differ from the number of HD cycles input from an external module. This bit is set to 1 when there is an illegal HD input from an external module. This bit is set to 1 when the number of clock cycles for the HD input to the CEU differs from the value set in the HCYL bits in CMCYR. Note however that when the HCYL bits are cleared to 0, this interrupt does not occur.
16	CDTOF	0	R/W	This bit is used for an interrupt indicating that data overflowed in the CRAM of the write buffer. Since data is input at realtime from an external module in capture operations, the frame image is overwritten unless the captured data is transferred from the CEU internal buffer to the memory at a certain or higher transfer rate. This bit is set to 1 when writing the data in the CRAM of the CEU internal write buffer to the bus is not performed within time and data has overflowed.
15	CPBE4	0	R/W	This bit is used for an interrupt indicating that writing to CDBYR2 and CDBCR2 in a bundle write has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next HD input. This bit is set to 1 when data for the number of lines set in CBDSR has been captured and the last data transfer to the bus has completed. However, this interrupt does not occur when the last captured data in a bundle write is the last captured data of a frame (field).
14	CPBE3	0	R/W	This bit is used for an interrupt indicating that writing to CDBYR and CDBCR in a bundle write has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next HD input. This bit is set to 1 when data for the number of lines set in CBDSR has been captured and the last data transfer to the bus has completed. However, this interrupt does not occur when the last captured data in a bundle write is the last captured data of a frame (field).

Bit	Bit Name	Initial Value	R/W	Description
13	CPBE2	0	R/W	This bit is used for an interrupt indicating that writing to CDAYR2 and CDACR2 in a bundle write has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next HD input. This bit is set to 1 when data for the number of lines (number of bytes in data enable fetch) set in CBDSR has been captured and the last data transfer to the bus has completed. However, in image capture or data synchronous fetch, this interrupt does not occur when the last captured data in a bundle write is the last captured data of a frame (field).
12	CPBE1	0	R/W	This bit is used for an interrupt indicating that writing to CDAYR and CDACR in a bundle write has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next HD input. This bit is set to 1 when data for the number of lines (number of bytes in data enable fetch) set in CBDSR has been captured and the last data transfer to the bus has completed. However, in image capture or data synchronous fetch, this interrupt does not occur when the last captured data in a bundle write is the last captured data of a frame (field).
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	VD	Undefined	R/W	This bit is used for an interrupt indicating that VD (vertical sync signal) was input from an external module. In data enable fetch mode, this bit is set to 1 when a VD input from an external module is detected. In image capture mode and data synchronous fetch mode, this module generates a VD interrupt in response to the first detection of the active level of HD following detection of the active level of VD from an external module. Note that, when VD and HD are asserted and detected at the same time, this module generates a VD interrupt at that time. Immediately after the VDPOL bit in CAMCR is modified, a pseudo VD is input and this bit is set to 1. The VD interrupt after the VDPOL bit is modified should be ignored.
8	HD	Undefined	R/W	This bit is used for an interrupt indicating that HD (horizontal sync signal) was input from an external module. This bit is set to 1 when an HD input from an external module is detected. Immediately after the HDPOL bit in CAMCR is modified, a pseudo HD is input and this bit is set to 1. The HD interrupt after the HDPOL bit is modified should be ignored.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	IGRW	0	R/W	This bit is used for an interrupt indicating that during capturing, access was attempted to a register to which writing during operation is prohibited. Among the CEU registers, writing during capturing is prohibited for some registers. Table 46.10 shows which registers can/cannot be written to during capturing. This bit is set to 1 when a register to which writing during capturing is prohibited has been written to.
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	CFE	0	R/W	This bit is used for an interrupt indicating that capturing of one field from an external module has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next VD input (see Figure 46.46). This interrupt occurs only in both-field capture mode.
0	CPE	0	R/W	This bit is used for an interrupt indicating that capturing of one frame from an external module has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next VD input. This interrupt indicates that capturing of one frame has finished. This bit is set to 1 when the image of the size set in CAPWR is captured and the last data transfer to the bus finished (see Figure 46.47).

Table 46.10 Registers that Can/Cannot Be Modified during Capturing

Register Name	Modification during Capturing
CAPSR	Possible
CAPCR	Prohibited
CAMCR	Prohibited
CMCYR	Prohibited
CAMOR	Possible
CAPWR	Possible
CAIFR	Prohibited
CRCNTR	Possible
CRCMPR	Prohibited
CFLCR	Possible
CFSZR	Possible
CDWDR	Possible
CDAYR	Possible
CDACR	Possible
CDBYR	Possible
CDBCR	Possible
CBDSR	Possible
CFWCR	Possible
CLFCR	Possible
CDOCR	Possible
CEIER	Possible
CETCR	Possible
CSTSR	Prohibited
CDSSR	Prohibited
CDAYR2	Possible
CDACR2	Possible
CDBYR2	Possible
CDBCR2	Possible

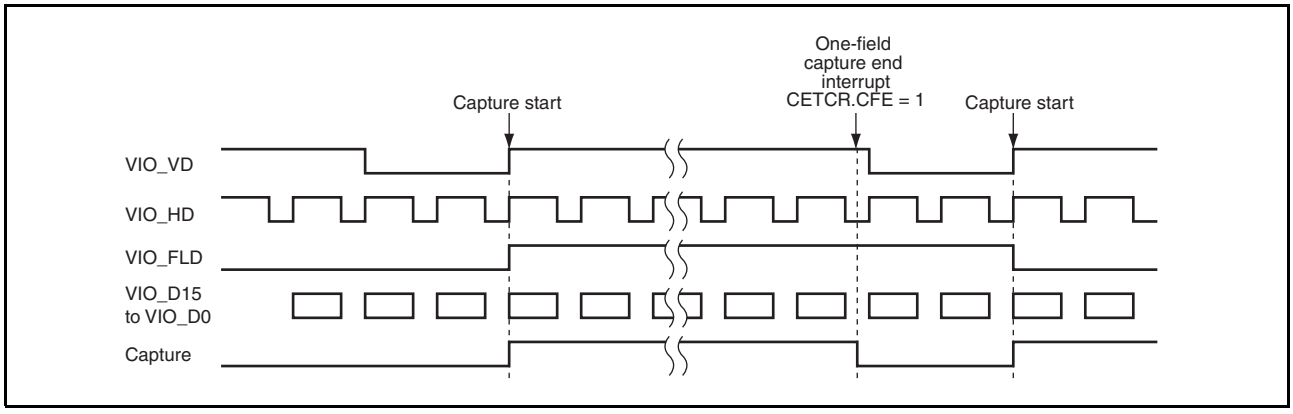


Figure 46.46 CFE Generation Timing

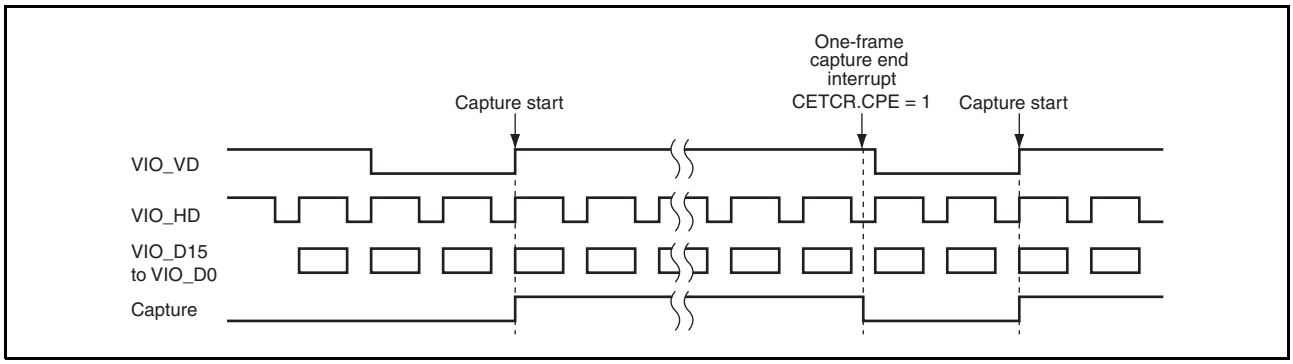


Figure 46.47 CPE Generation Timing

46.4.23 Capture Status Register (CSTSR)

CSTSR indicates the internal status of the CEU. CSTSR differs from CETCR in that no interrupt is generated for the events indicated in CSTSR.

The CEU operating/halt state can be confirmed using CSTSR. To confirm the halt state of the CEU, make sure that the status bit (bit 0) indicating that the CEU is operating is cleared to 0 for sure.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	CRST	—	—	—	—	—	—	—	CP FLD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CP TON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	CRST	0	R	Indicates which register plane is currently used. 0: Plane A of the register is being used 1: Plane B of the register is being used
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CPFLD	0	R	Indicates which field is being captured. 0: Bottom field is being captured 1: Top field is being captured
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	CPTON	0	R	Indicates that the CEU is operating. This bit retains 1 during the period that starts from the internal VD at capture start and ends when a one-frame capture end interrupt occurs. Figure 46.48 shows the CEU operating period.

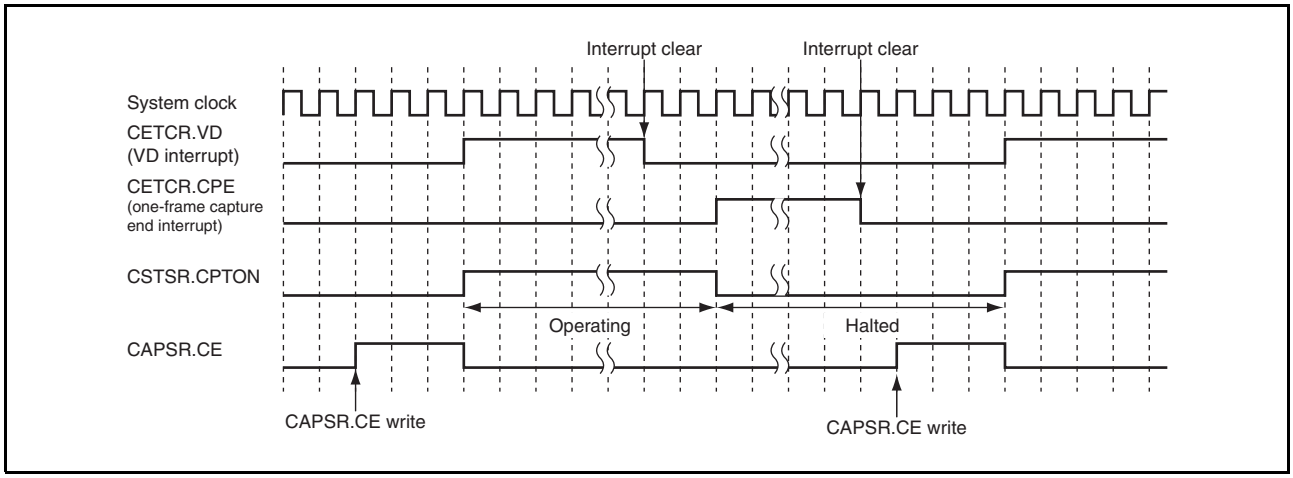


Figure 46.48 Operating Status during Capturing

46.4.24 Capture Data Size Register (CDSSR)

CDSSR indicates the size of data written to the memory in data enable fetch. As this register indicates a correct value at the end of capture, confirm this register when capture is completed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDSS[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDSS[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CDSS[31:0]	H'0000 0000	R	Indicate the size of data written to the memory in data enable fetch. In a bundle write, size of data written to the selected address at the end of one-frame capture is indicated. In a bundle write, as soon as the number of bytes specified by CBDSR is transferred to the bus, address to which data is written is switched. Therefore, if one-frame capture is completed at the same time as a bundle write is completed, this register indicates H'0000 0000. Figure 46.49 and Figure 46.50 show the overall timing of the CDSSR operation in a bundle write.

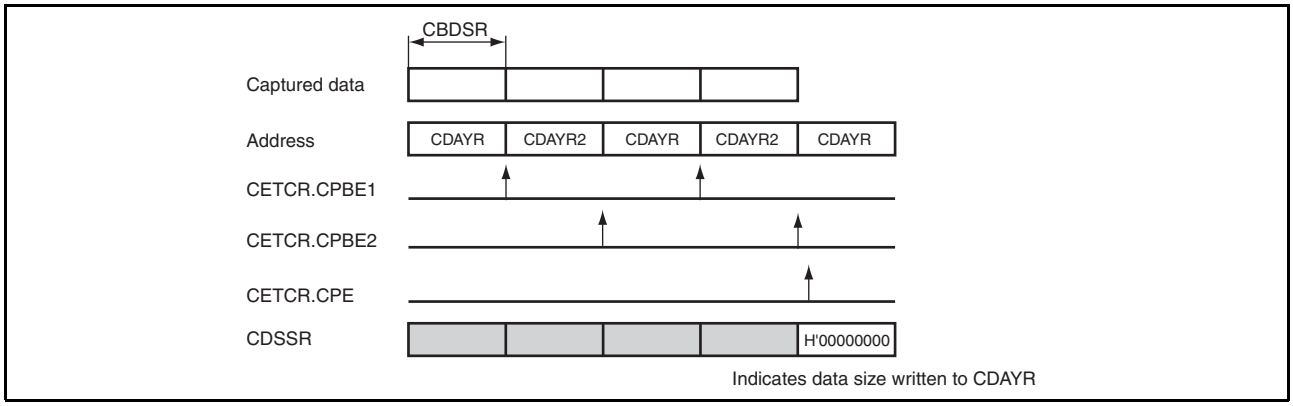


Figure 46.49 Overall Timing of CDSSR Operation in Bundle Write (When Bundle Write End and Capture End Coincide)

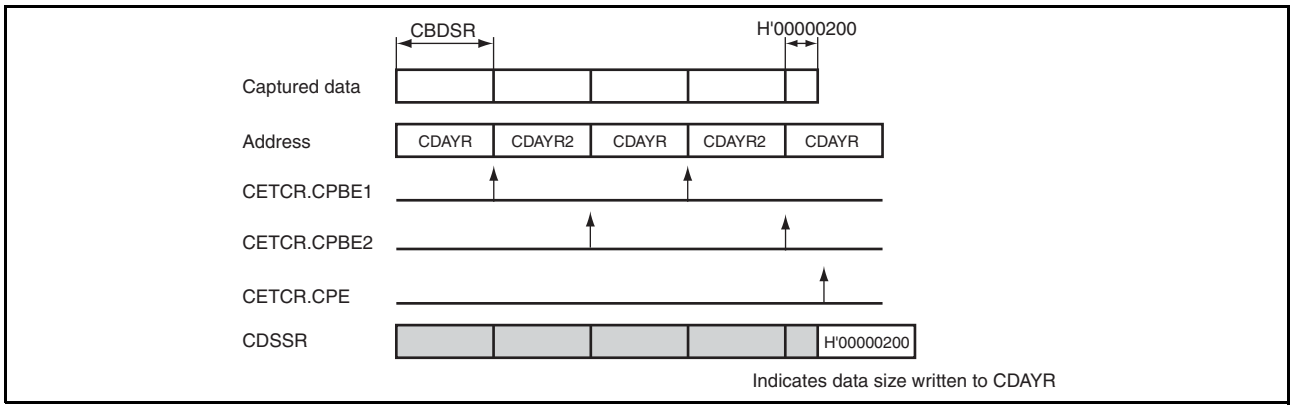


Figure 46.50 Overall Timing of CDSSR Operation in Bundle Write (When Bundle Write End and Capture End Do Not Coincide)

46.4.25 Capture Data Address Y Register 2 (CDAYR2)

CDAYR2 specifies the address for the luminance (Y) component used in a bundle write and the address for data storage in a bundle write in data fetch. CDAYR2 is used only in a bundle write.

CDAYR2 specifies the address where the Y component of the captured data is to be stored in frame image capture or one-field image capture, the address where the Y component of the captured top field is to be stored in both-field image capture, and the address where data is to be stored in data fetch. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. In frame image capture or one-field image capture, set the start address of the memory area where the Y component of the captured data is to be stored by CDAYR2. In both-field image capture, set the start address of the memory area where the Y component of the captured top-field image is to be stored by CDAYR2. In data fetch, set the start address of the memory area to be used for data storage.

The address specified by this register must be in 32-bit units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAYR2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAYR2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2 1, 0	CAYR2[31:2] CAYR2[1:0]	H'0000 0000	R/W R	• Frame image capture: These bits set the address for storing the Y component data of the captured data (4-pixel units). • One-field image capture: These bits set the address for storing the Y component data of the captured data (4-pixel units). • Both-field image capture: These bits set the address for storing the Y component data of the captured top-field data (4-pixel units). • Data synchronous fetch: These bits set the address for storing data (4-byte units). • Data enable fetch: These bits set the address for storing data (32-byte units).

Set the address of the starting point of the memory area where the Y component of the captured image is to be stored in this register, as shown in Figure 46.51.

- Frame image capture: Set the address of the starting point of the memory area where the Y component of the captured image is to be stored.
- One-field image capture: Set the address of the starting point of the memory area where the Y component of the captured image is to be stored.
- Both-field image capture: Set the address of the starting point of the memory area where the Y component of the captured top-field image is to be stored.
- Data synchronous fetch: Set the address of the starting point of the memory area where the captured data is to be stored.
- Data enable fetch: Set the address of the starting point of the memory area where the captured data is to be stored in 32-byte units.

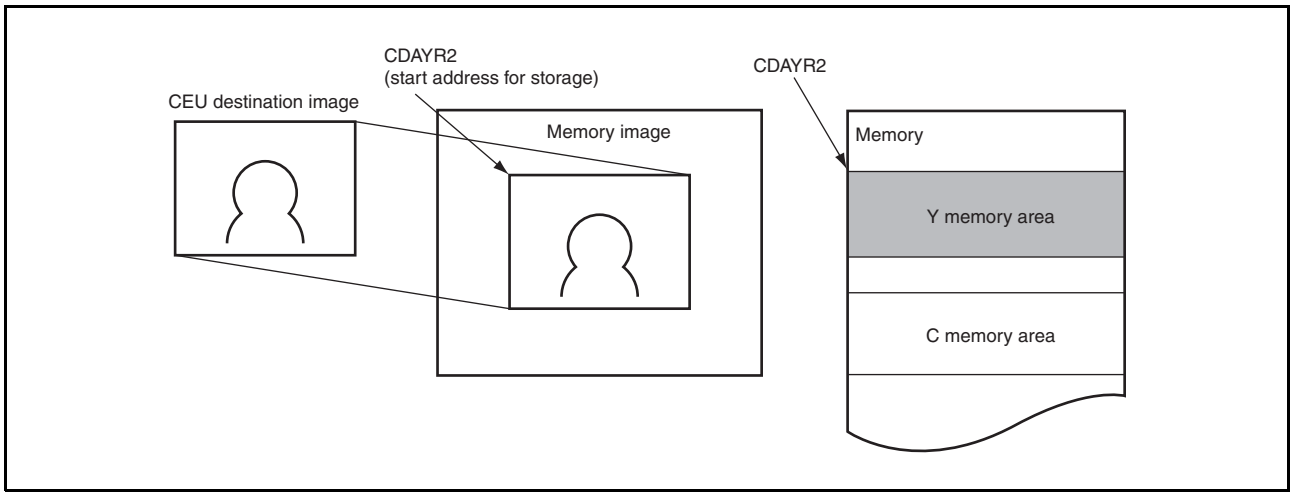


Figure 46.51 Relationship between Captured Image and Y Component Memory Area

46.4.26 Capture Data Address C Register 2 (CDACR2)

CDACR2 specifies the address for the chrominance (C) component used in a bundle write. CDACR2 is used only in a bundle write.

CDACR2 specifies the address where the C component of the captured data is to be stored in frame image capture or one-field image capture, and the address where the C component of the captured top field is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. In frame image capture or one-field image capture, set the start address of the memory area where the C component of the captured data is to be stored by CDACR2. In both-field image capture, set the start address of the memory area where the C component of the captured top-field image is to be stored by CDACR2. CDACR2 is not used in data fetch.

The address specified by this register must be in 32-bit units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CACR2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CACR2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CACR2[31:2]	H'0000 0000	R/W	• Frame image capture: These bits set the address for storing the C component data of the captured data (4-pixel units). • One-field image capture: These bits set the address for storing the C component data of the captured data (4-pixel units). • Both-field image capture: These bits set the address for storing the C component data of the captured top-field data (4-pixel units).
1, 0	CACR2[1:0]		R	

In this register, set the address of the starting point of the memory area where the captured data is to be stored in a bundle write, as shown in Figure 46.52.

- Frame image capture: Set the address of the starting point of the memory area where the C component of the captured image is to be stored.
- One-field image capture: Set the address of the starting point of the memory area where the C component of the captured image is to be stored.
- Both-field image capture: Set the address of the starting point of the memory area where the C component of the captured top-field image is to be stored.

The C component has an output data format as shown in Figure 46.53, and is saved in the memory in this format.

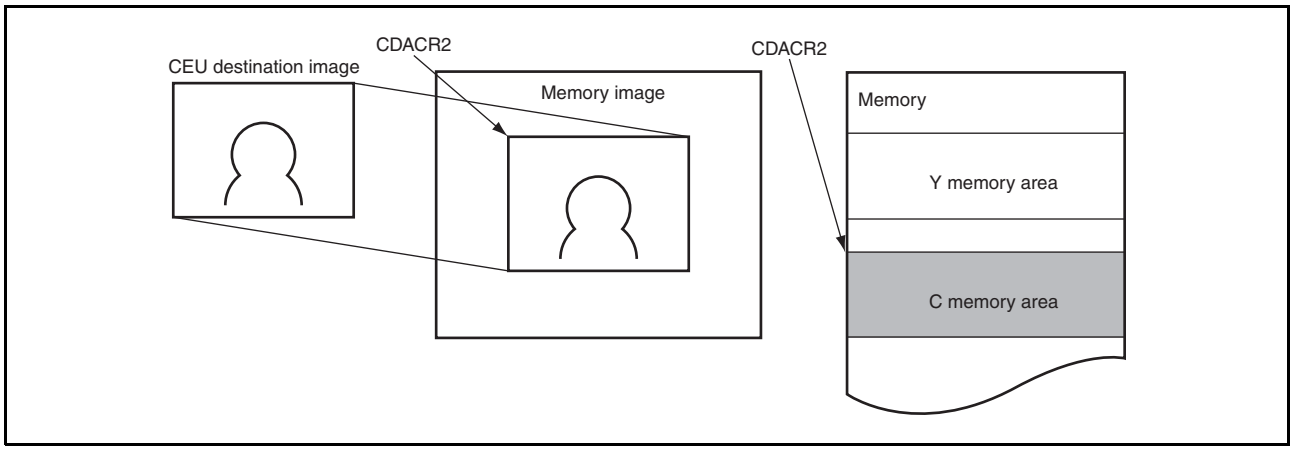


Figure 46.52 Relationship between Captured Image and C Component Memory Area

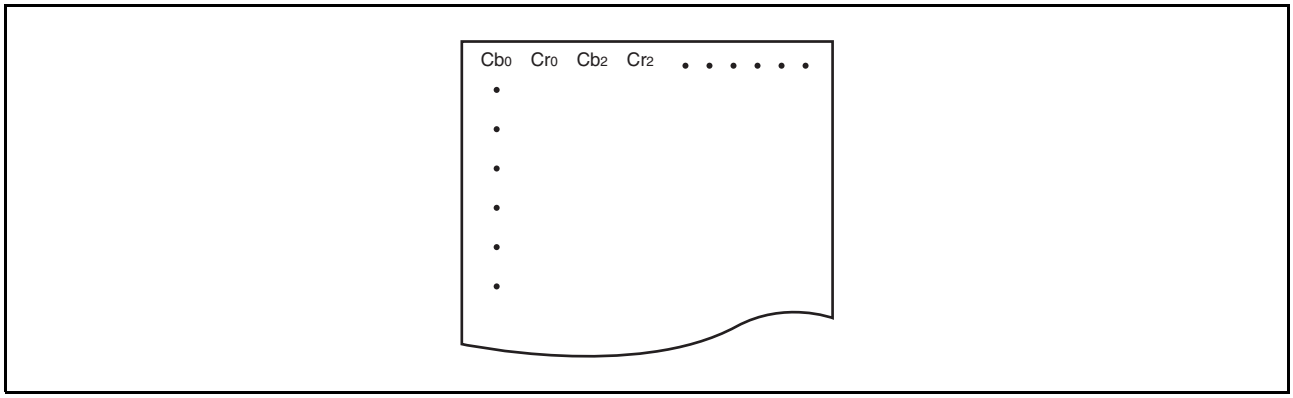


Figure 46.53 Image of Storing C Components in Memory

46.4.27 Capture Data Bottom-Field Address Y Register 2 (CDBYR2)

CDBYR2 specifies the address for the luminance (Y) component of the bottom field used in a bundle write. CDBYR2 is used only in a bundle write.

CDBYR2 specifies the address where the Y component of the captured bottom-field data is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. Set the start address of the memory area where the Y component of the bottom-field image captured in both-field image capture is to be stored by CDBYR2. CDBYR2 is not used in frame image capture, one-field image capture, or data fetch.

The address specified by this register must be in 32-bit units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CBYR2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CBYR2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CBYR2[31:2]	H'0000 0000	R/W	These bits set the address for storing the Y component data of the captured bottom-field data (4-pixel units).
1, 0	CBYR2[1:0]		R	

In this register, set the address of the starting point of the memory area where the Y component of the captured bottom-field image is to be stored in a bundle write, as shown in Figure 46.54.

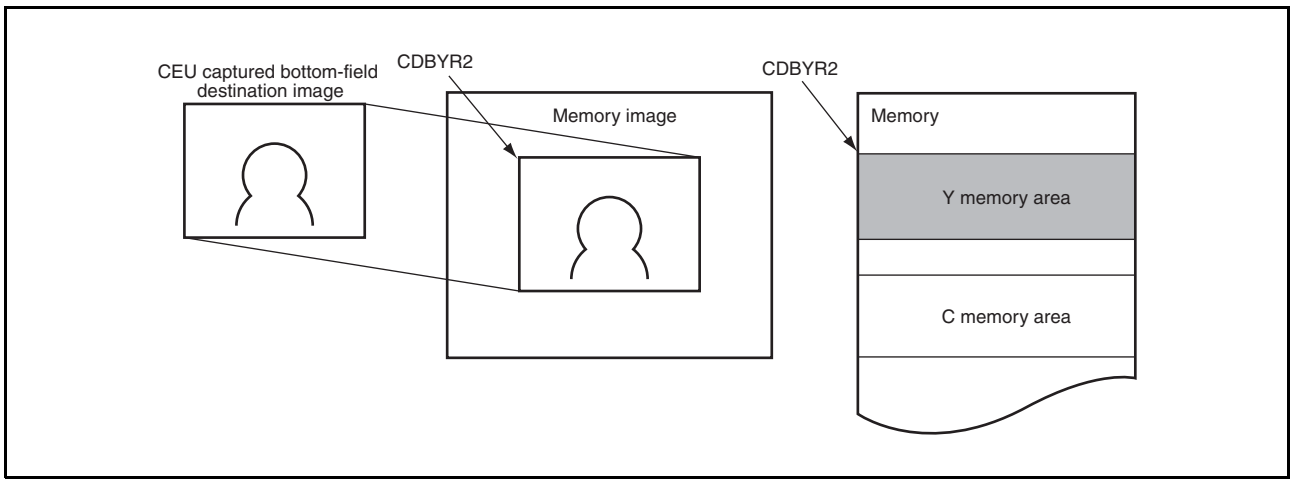


Figure 46.54 Relationship between Captured Bottom-Field Image and Y Component Memory Area

46.4.28 Capture Data Bottom-Field Address C Register 2 (CDBCR2)

CDBCR2 specifies the address for the chrominance (C) component of the bottom field used in a bundle write. CDBCR2 is used only in a bundle write.

CDBCR2 specifies the address where the C component of the captured bottom-field data is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. Set the start address of the memory area where the C component of the bottom-field image captured in both-field image capture is to be stored by CDBCR2. CDBCR2 is not used in frame image capture, one-field image capture, or data fetch.

The address specified by this register must be in 32-bit units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CBCR2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CBCR2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CBCR2[31:2]	H'0000 0000	R/W	These bits set the address for storing the C component data of the captured bottom-field data (4-pixel units).
1, 0	CBCR2[1:0]		R	

In this register, set the address of the starting point of the memory area where the C component of the captured bottom-field image is to be stored in a bundle write, as shown in Figure 46.55. The C component has an output data format as shown in Figure 46.56, and is saved in the memory in this format.

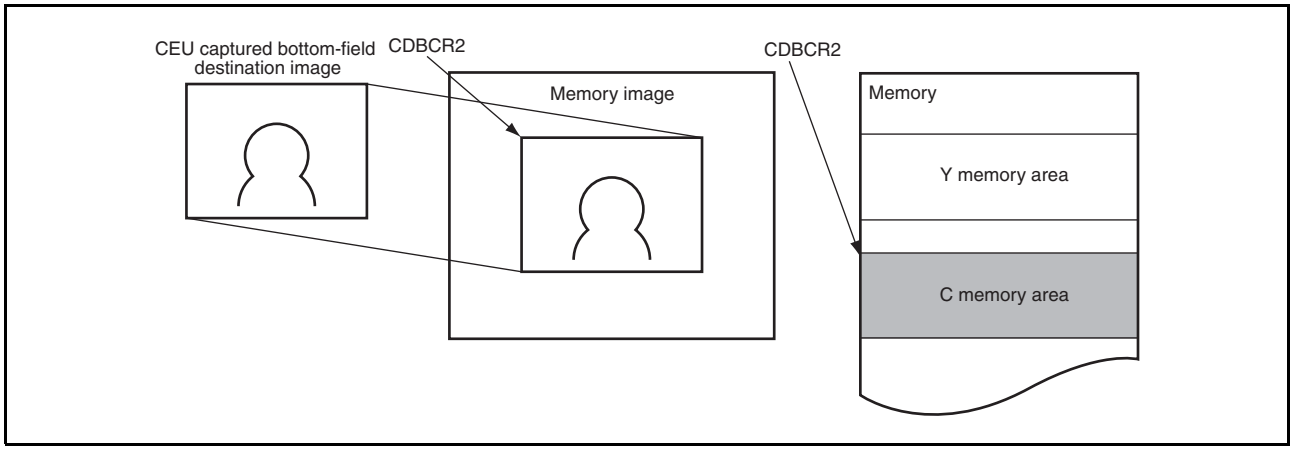


Figure 46.55 Relationship between Captured Bottom-Field Image and C Component Memory Area

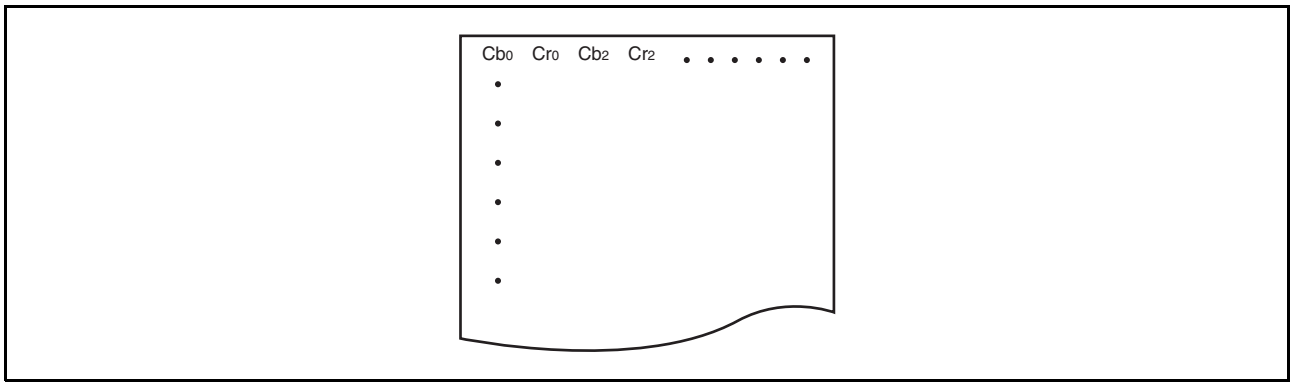


Figure 46.56 Image of Storing C Components in Memory

46.5 Usage Notes for CEU

46.5.1 Conditions for Connection to an External Module

(1) Clock Frequency

The external input clock should have a frequency at most the same as the CEU operating clock frequency (Bφ), with jitter on both sides included.

CEU operating clock frequency (Bφ) ≥ External input clock frequency

(2) Blanking Period

The period from the last valid pixel in each line to the next horizontal sync signal HD must be at least 20 cycles.

(3) Fixed Period of Field Identification Signal

The field identification signal FLD should be fixed for at least 1-HD period since a VD input

46.5.2 Restrictions on Input/Output Functions

Table 46.11 lists the restrictions regarding the CEU input/output functions.

Table 46.11 Restrictions on CEU Input/Output Functions

Item	Restrictions
External module interface	The operating clock of the external module (VIO_CLK) should always have a frequency at most the same as that of the CEU operating clock (Bφ), with jitter on both sides included.
	Selecting the interface, or modifying the frequency of the external module operating clock or HD/VD polarity must be done when capture operations are halted for sure.
	The capture horizontal size in image capture must be specified as follows: 8-bit interface: 8-cycle units 16-bit interface: 4-cycle units
	The capture horizontal size in data fetch must be specified as follows: 8-bit interface: 4-cycle units 16-bit interface: 2-cycle units
	The capture vertical size must be specified in 4-line units.
	The maximum number of cycles in the horizontal sync signal period should be 16,375 cycles of external input clock for 8-bit digital image input pins, or 16,379 cycles of external input clock for 16-bit digital image input pins.
	The maximum number of lines (HD count) in the vertical sync signal should be 16,382 lines.
	The minimum number of captured pixels should be sub-QCIF (128 × 96).
	The maximum number of captured pixels should be 5 megapixels (2,560 × 1,920).
	Captured size in data enable fetch Maximum: 6 Mbytes (2,048 × 1,536 × 2) Minimum: 16 bytes
Memory output	The output address must be specified in 32-bit units.
	The horizontal size of the destination image (memory) must be specified in 4-pixel units.
	The number of horizontal output pixels (= horizontal clipping size) must be specified in 4-pixel units.
	The number of vertical output lines (HD) (= vertical clipping size) must be specified in 4-line (HD) units.
	In data enable fetch bundle write, the output address must be specified in 32-byte units.
Internal processing	The filter clipping size must be specified as a value equal to or lower than the actual output size of the filter.

46.5.3 Display in the Video Display Controller 6

Since the input data are written to the memory separately as Y data and CbCr data in image capture mode, the captured data cannot be displayed in the video display controller 6.

46.5.4 Software Reset

For transitions to the software reset state by the CPKIL bit in the CAPSR register, see section 52.3.6, Software Reset. However, where the procedure refers to the SRST bit, read this as the CPKIL bit in the CAPSR register.

47. MIPI CSI2 Interface

47.1 Overview

The CSI2 module is a MIPI (Mobile Industry Processor Interface Alliance) CSI-2 (Camera Serial Interface 2) receiver module. It supports MIPI CSI-2 V1.1 and MIPI D-PHY V2.0. It includes the PHY and LINK blocks. The PHY block incorporates a DPHY. This LSI chip has one CSI2 set, which has two lanes.

The received signals are input to the PHY block; the LINK block receives the bytes of data output by the PHY block; and the PHY block extracts the image data, data type, and channel information before it outputs the data to the subsequent VIN module. Here, the LINK block generates the vertical sync signals (VD), field signals (FLD), horizontal sync signals (HD), data enable signals (PE), and byte enable signals (PEB) for each channel, and outputs them to the VIN module. The LINK registers are used to control the PHY block and monitor its state.

47.1.1 Features

The CSI2 module has the following features.

(1) Transfer rates from 80 Mbps to 1.0 Gbps for MIPI CSI-2 data (refer to section 47.3.1)
(2) ECC 1-bit error correction and 2-bit or more error detection for packet headers (refer to section 47.3.2)
(3) CRC error detection for payload data (refer to section 47.3.3)
(4) Generation of VD (vertical sync), HD (horizontal sync), and FLD (field) signals (refer to section 47.3.4)
(5) Interrupts (refer to section 47.3.6)
(6) Lane swapping (refer to section 47.3.7)
(7) Initial settings of PHY (refer to section 47.3.9)
(8) PHY control and monitoring through register settings (refer to section 47.3.10)
(9) Interrupt generation in response to the ultra-low-power (ULP) state and reception error state (refer to section 47.3.11)
(10) Supported data type (refer to section 48, Video Input Module (VIN))

47.1.2 Block Diagram

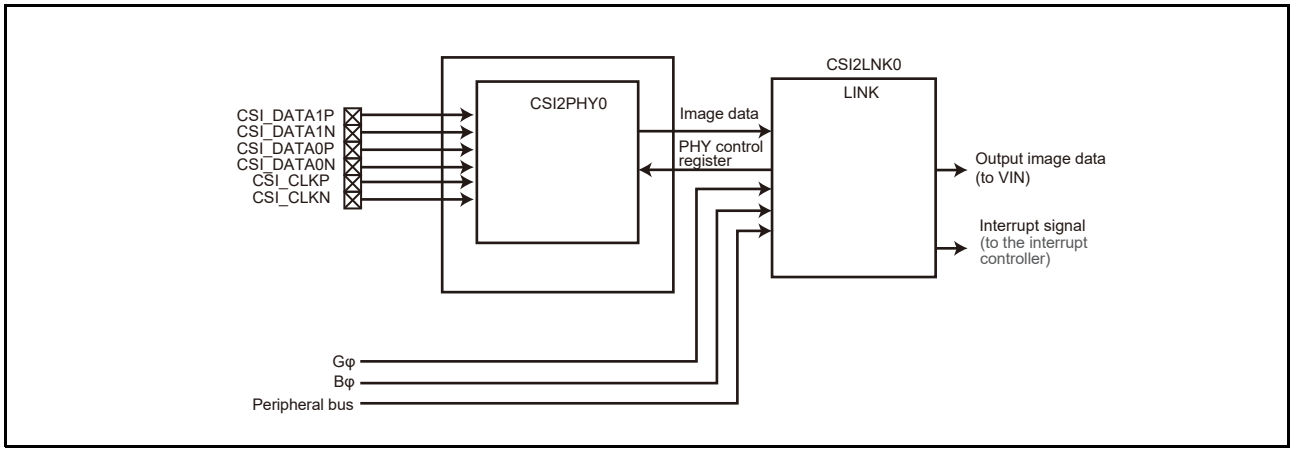


Figure 47.1 CSI2 Block Diagram

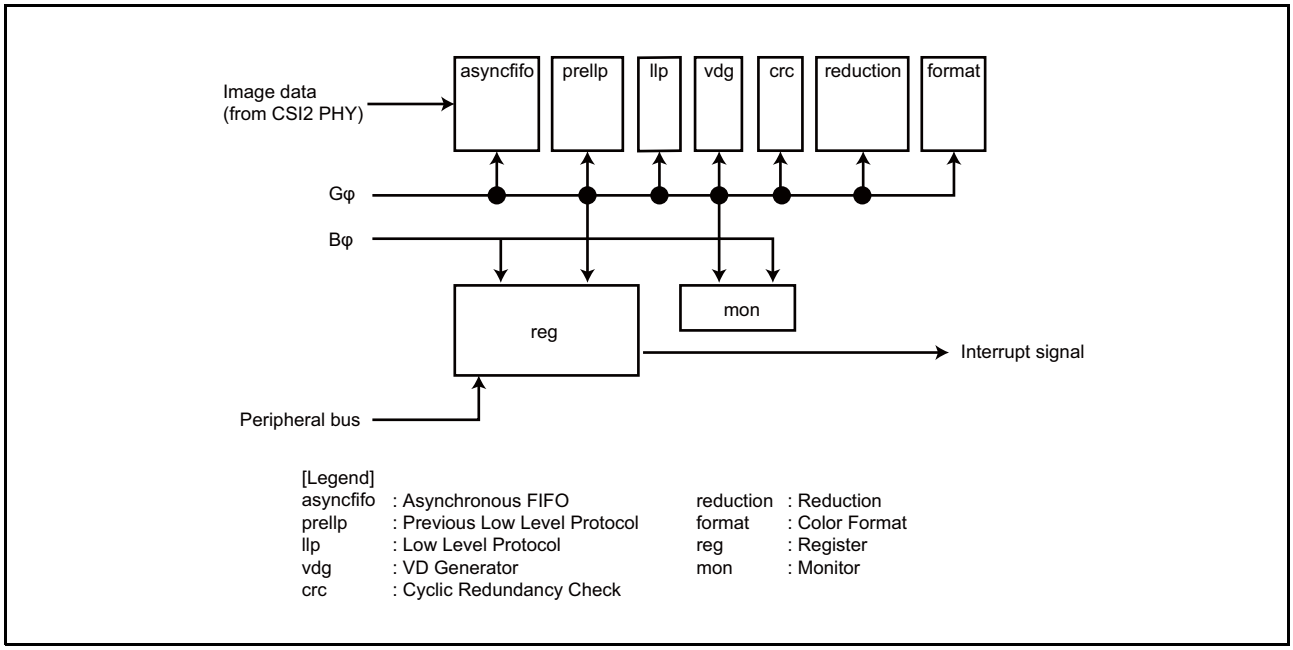


Figure 47.2 CSI2 LINK Block Diagram

47.1.3 External Pins

Table 47.1 External Pins

Pin Name	Abbreviation	I/O	Function
Lane 0 positive data pin	CSI_DATA0P	Input	Differential positive receive data input on CSI2 lane 0
Lane 0 negative data pin	CSI_DATA0N	Input	Differential negative receive data input on CSI2 lane 0
Lane 1 positive data pin	CSI_DATA1P	Input	Differential positive receive data input on CSI2 lane 1
Lane 1 negative data pin	CSI_DATA1N	Input	Differential negative receive data input on CSI2 lane 1
Clock lane positive data pin	CSI_CLKP	Input	Differential positive reception input on CSI2 clock lane
Clock lane negative data pin	CSI_CLKN	Input	Differential negative reception input on CSI2 clock lane

47.1.4 Register Configuration

The following lists the base address of this module.

Module Name	Address
CSI2LNK0	H'E820 9000

Table 47.2 Register Configuration

Register Name	Abbreviation	R/W	Address	Initial Value	Access Size
Control Timing Select Register	TREF	R/W	H'0000	H'0000 0001	32
Software Reset Register	SRST	R/W	H'0004	H'0000 0000	32
PHY Operation Control Register	PHYCNT	R/W	H'0008	H'0000 0000	32
Checksum Control Register	CHKSUM	R/W	H'000C	H'0000 0003	32
Channel Data Type Select Register	VCDT	R/W	H'0010	H'011E 801E	32
Frame Data Type Select Register	FRDT	R/W	H'0018	H'0001 0000	32
Field Detection Control Register	FLD	R/W	H'001C	H'0000 0000	32
Automatic Standby Control Register	ASTBY	R/W	H'0020	H'0000 3F21	32
Long Data Type Setting Register 0	LNGDT0	R/W	H'0028	H'FFFF 0000	32
Long Data Type Setting Register 1	LNGDT1	R/W	H'002C	H'FFFF FFFF	32
Interrupt Enable Register	INTEN	R/W	H'0030	H'0000 0000	32
Interrupt Source Mask Register	INTCLOSE	R/W	H'0034	H'0004 0000	32
Interrupt Status Monitor Register	INTSTATE	R/(W)*1	H'0038	H'0000 0000	32
Interrupt Error Status Monitor Register	INTERRSTATE	R/(W)*2	H'003C	H'0000 0000	32
Short Packet Data Register	SHPDAT	R	H'0040	H'0000 0000	32
Short Packet Count Register	SHPCNT	R	H'0044	H'0000 0000	32
LINK Operation Control Register	LINKCNT	R/W	H'0048	H'8000 0000	32
Lane Swap Register	LSWAP	R/W	H'004C	H'0000 00E4	32
PHY ESC Error Monitor Register	PHEERM	R	H'0074	H'0000 0000	32
PHY Clock Lane Monitor Register	PHCLM	R	H'0078	H'0000 000C	32
PHY Data Lane Monitor Register	PHDLM	R	H'007C	H'0000 3000	32
Packet Header 0 Monitor Register 0	PH0M0	R	H'00F0	H'0000 0000	32
Packet Header 0 Monitor Register 1	PH0M1	R	H'00F4	H'0000 0000	32
Packet Header 1 Monitor Register 0	PH1M0	R	H'00F8	H'0000 0000	32
Packet Header 1 Monitor Register 1	PH1M1	R	H'00FC	H'0000 0000	32
Packet Header 2 Monitor Register 0	PH2M0	R	H'0100	H'0000 0000	32

Table 47.2 Register Configuration

Register Name	Abbreviation	R/W	Address	Initial Value	Access Size
Packet Header 2 Monitor Register 1	PH2M1	R	H'0104	H'0000 0000	32
Packet Header 3 Monitor Register 0	PH3M0	R	H'0108	H'0000 0000	32
Packet Header 3 Monitor Register 1	PH3M1	R	H'010C	H'0000 0000	32
Packet Header R Monitor Register 0	PHRM0	R	H'0110	H'0000 0000	32
Packet Header R Monitor Register 1	PHRM1	R	H'0114	H'0000 0000	32
Packet Header R Monitor Register 2	PHRM2	R	H'0118	H'0000 0000	32
Packet Header C Monitor Register 0	PHCM0	R	H'0120	H'0000 0000	32
Packet Header C Monitor Register 1	PHCM1	R	H'0124	H'0000 0000	32
CRC Monitor Register 0	CRCM0	R	H'0128	H'0000 0000	32
CRC Monitor Register 1	CRCM1	R	H'012C	H'0000 0000	32
SOT Error Count Register	SERRCNT	R	H'0140	H'0000 0000	32
SOTSYNC Error Count Register	SSERRCNT	R	H'0144	H'0000 0000	32
ECC_CRCT Count Register	ECCCM	R	H'0148	H'0000 0000	32
ECC_ERR Count Register	ECECM	R	H'014C	H'0000 0000	32
CRC_ERR Count Register	CRCECM	R	H'0150	H'0000 0000	32
Line Register	LCNT	R	H'0160	H'0000 0000	32
Line Monitor Register	LCNTM	R	H'0168	H'0000 0000	32
Frame Count Monitor Register	FCNTM	R	H'0170	H'0000 0000	32
PHY Input Data Monitor Register	PHYDIM	R	H'0180	H'0000 0000	32
PHY Input Monitor Register	PHYIM	R	H'0184	H'0000 0000	32
VIN Data Monitor Register	VINDM	R	H'018C	H'0000 0000	32
VIN Signal Monitor Register 1	VINSM1	R	H'0190	H'0000 0000	32
VIN Signal Monitor Register 3	VINSM3	R	H'0198	H'0000 0000	32
PHY Output Monitor Register	PHYOM	R	H'019C	H'0000 0000	32
Packet Header Monitor Register 1	PHM1	R	H'01C0	H'0000 0000	32
Packet Header Monitor Register 2	PHM2	R	H'01C4	H'0000 0000	32
Packet Header Monitor Register 3	PHM3	R	H'01C8	H'0000 0000	32
Packet Header Monitor Register 4	PHM4	R	H'01CC	H'0000 0000	32
Packet Header Monitor Register 5	PHM5	R	H'01D0	H'0000 0000	32
Packet Header Monitor Register 6	PHM6	R	H'01D4	H'0000 0000	32
Packet Header Monitor Register 7	PHM7	R	H'01D8	H'0000 0000	32
Packet Header Monitor Register 8	PHM8	R	H'01DC	H'0000 0000	32
PHY Timing Register 1	PHYTIM1	R/W	H'0264	H'0000 0000	32
PHY Timing Register 2	PHYTIM2	R/W	H'0268	H'0000 0000	32
PHY Timing Register 3	PHYTIM3	R/W	H'026C	H'0000 0000	32

Note 1. Only 1 can be written to clear the flag; bits 27, 4, and 3 are read-only bits and any value cannot be written to.
Note 2. Only 1 can be written to clear the flag.
Note 3. Access to these registers is only possible in 32-bit units.

47.2 Register Description

[Legend]

Initial value: Register value after a reset
—: Undefined value
R/W: Readable/writable. The written value can be read.
R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.
R: Read-only. The write value should always be 0.
—/WB: Write-only. The read value is undefined.

47.2.1 Control Timing Select Register (TREF)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TREF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TREF	1	R/W	Register value reflection timing selection 1: Immediately Do not change this bit from its initial value.

47.2.2 Software Reset Register (SRST)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SRST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SRST	0	R/W	Software reset control 0: Normal 1: Reset This bit resets the control circuit in the CSI2 block. To reset the circuit, set this bit to 1. Be sure to set this bit to 0 to cancel a reset. A software reset does not initialize the CSI2 registers.

Note: The setting of the effective bit of this register becomes applicable the instant the register is set.

47.2.3 PHY Operation Control Register (PHYCNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SHUTDOWNZ	RSTZ
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	ENABLECLK	—	—	ENABLE_1	ENABLE_0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	SHUTDOWNZ	0	R/W	Setting this bit to 0 shuts down the PHY block.
16	RSTZ	0	R/W	Setting this bit to 0 initializes the PHY block.
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	ENABLECLK	0	R/W	Clock lane operation enable 0: Disables operation. 1: Enables operation. When any of the ENABLE_1 to ENABLE_0 bits is set to 1, be sure to set the ENABLECLK bit to 1.
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	ENABLE_1	0	R/W	Data lane 1 operation enable 0: Disables operation. 1: Enables operation. Note: When 1- lane PHY is connected, this setting has no effect.
0	ENABLE_0	0	R/W	Data lane 0 operation enable 0: Disables operation. 1: Enables operation. Always set this bit to 1.

Note: The setting of the effective bit of this register becomes applicable the instant the register is set. Set this register while data input is suspended (the sensor is halted); never change the settings of this register during operation.

47.2.4 Checksum Control Register (CHKSUM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ECC_EN	CRC_EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	ECC_EN	0	R/W	ECC process control 0: Disabled 1: Enabled This bit controls ECC process of the packet header (PH) for Low Level Protocol (LLP). Set this bit to 1 to use ECC. When set to 0, data is always processed assuming it has no errors and thus no errors are corrected.
0	CRC_EN	0	R/W	CRC checksum process control 0: Disabled 1: Enabled This bit controls checksum process of the long packet data for Low Level Protocol (LLP). Set this bit to 1 to use CRC checksum. When set to 0, data is always processed assuming it has no errors.

Note: The settings of the effective bits of this register become applicable the instant the register is set. Set this register while data input is suspended (the sensor is halted); never change the settings of this register during operations.

47.2.5 Channel Data Type Select Register (VCDT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VCDT_EN	—	—	—	—	—	SEL_VC		—	SEL_DT_ON	SEL_DT					
Initial value:	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
R/W:	R/W	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	—	H'1E	R	Reserved These bits are always read as H’1E. The write value should always be H’1E.
15	VCDT_EN	1	R/W	Channel enable 0: Disabled 1: Enabled This bit enables or disables channel. When set to 0, VD and FLD are fixed to low and TAG = 0 is not generated.
14 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	SEL_VC	H'0	R/W	Channel select channel H'0 to H'3 These bits specify the virtual channels for channel to capture. Channels can be selected from among four channels from H'0 to H'3. Data is output from the channel when the value of this register matches the received virtual channel.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	SEL_DT_ON	0	R/W	Channel select data type enable 0: Disabled 1: Enabled This bit enables or disables data type match function using SEL_DT. When set to 0, data is output from the channel even when the setting of SEL_DT does not match the received data type as long as it matches the virtual channel. When set to 1, data is output from the channel only when the setting of SEL_DT matches the received data type and the virtual channel.
5 to 0	SEL_DT	H'1E	R/W	Channel select data type H'00 to H'3F These bits specify the data type for channel to capture. The CSI2 LINK block outputs the received payload data as it is regardless of the data type when SEL_DT_ON is set to 0 whereas there are some restrictions on the corresponding data type (such as color format) for the following VIN module.

Note 1. This register must be set during the initialization sequence.
Note 2. In the table above, “virtual channel” refers to a channel through which data flows from the camera and “channel” refers to an output channel of the CSI2.

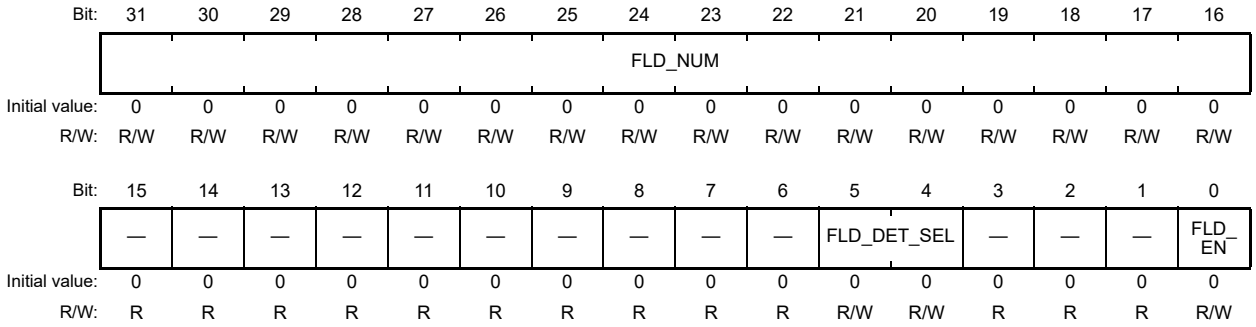
47.2.6 Frame Data Type Select Register (FRDT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	DT_FS						—	—	DT_FE					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 24	DT_FS	H'00	R/W	Frame start data type These bits specify the frame start data type. When the value of these bits matches the received data type, the data is determined as a frame start.
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21 to 16	DT_FE	H'01	R/W	Frame end data type These bits specify the frame end data type. When the value of these bits matches the received data type, the data is determined as a frame end.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

47.2.7 Field Detection Control Register (FLD)



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	FLD_NUM	H'0000	R/W	Even field number setting These bits specify the value for detecting the even field of the interlaced image. For detecting the even field, the WC value (16 bits) of the Frame Start packet is referred to. When the WC value matches the setting of these bits, the field is detected as the even field. During the frame period, the CSIR_FLD signal stays High.
15 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	FLD_DET_SEL	B'00	R/W	Even field detection condition select 00: The field is detected as the first field when FLD_NUM matches WC. 01: The field is detected as the first field when FLD_NUM[0] matches WC[0]. 10: The field is detected as the first field when FLD_NUM[0] matches WC[8]. 11: Setting is prohibited.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	FLD_EN	0	R/W	Channel even field detection control This bit enables or disables the detection of even fields in channel 1. Set this bit to 1 to enable detection. This bit must be set to 0 when input images are arranged for progressive (sequential) scanning.

Note: This register must be set during the initialization sequence.

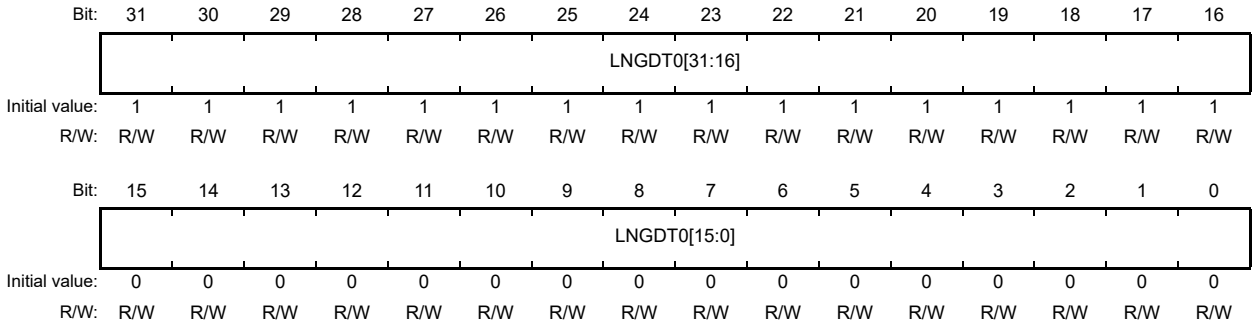
47.2.8 Automatic Standby Control Register (ASTBY)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	VD_MSK_CYCLE						—	—	VD_MSK_EN	AUTO_STANDBY_EN				
Initial value:	0	0	1	1	1	1	1	1	0	0	1	0	0	0	0	1
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

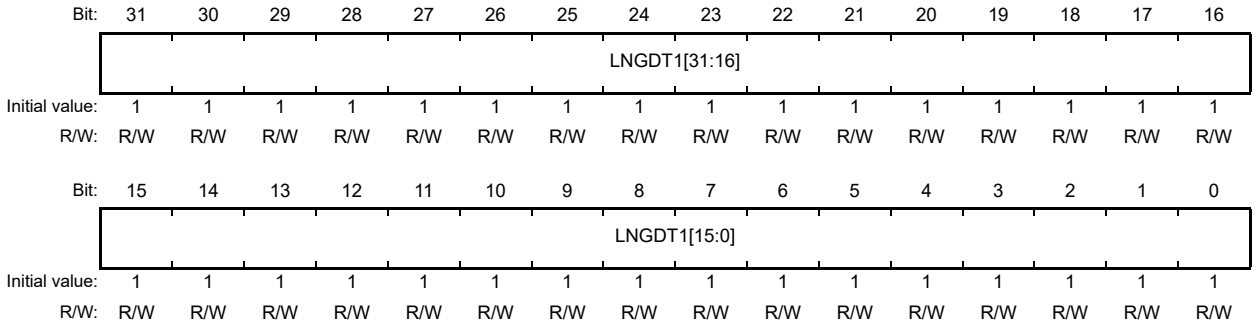
Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 8	VD_MSK_CYCLE	H'3F	R/W	VD masking cycle These bits specify the period for the masking of VD and FLD in response to frame-start errors. A frame-start error is the state in which FS is received and a further FS is received before FE is received. When a frame-start error occurs, the VD signal does not go low (Vsync is not generated); masking VD leads to the generation of the period over which VD is low. VD is masked for 4 × VD_MSK_CYCLE × input clock period. Maximum value: This bit must be set so that VD is masked for less than the period from an EOT short packet (frame start) to an SOT long packet. Minimum value: This bit must be at least 5, i.e. VD is masked for 20 × input clock period.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	VD_MSK_EN	1	R/W	VD masking enable 0: Disables masking VD. 1: Enables masking VD. This bit enables or disables VD and FLD masking function when a frame start error occurs. When this bit is set to enable the function, VD and FLD are masked for the period specified with the VD_MSK_CYCLE bits.
4 to 0	AUTO_STANDBY_EN	H'01	R/W	Automatic standby control 0: Neither VD nor FLD is initialized. 1: VD and FLD are initialized. [0]: ECC 2-bit error [1]: ECC 1-bit error on receiving FS [2]: ECC 1-bit error on receiving FE [3]: ECC 1-bit error on receiving data other than FS or FE [4]: CRC error Each bit enables or disables initialization of VD and FLD in response to the corresponding error. Set the corresponding bit to 1 to enable initialization of the VD and FLD signals.

47.2.9 Long Data Type Setting Register 0 (LNGDT0)



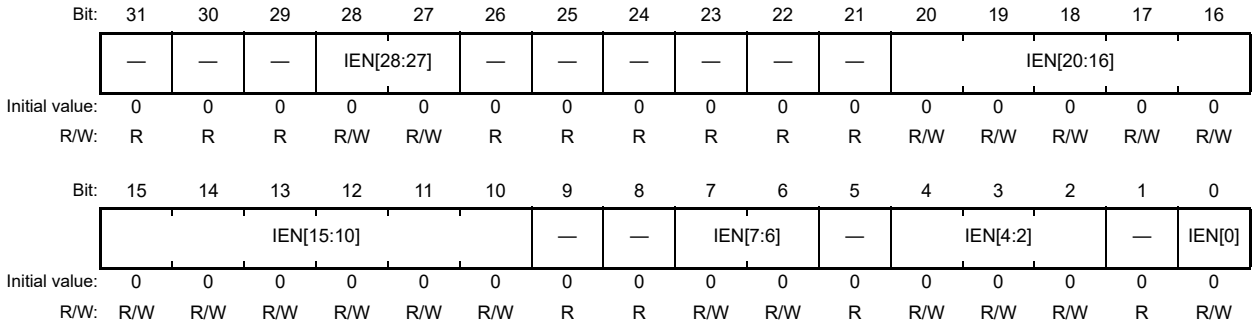
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LNGDT0	H'FFFF 0000	R/W	Long data type setting 0: Short packet 1: Long packet This register is combined with LNGDT1 register as in LNGDT = {LNGDT1, LNGDT0}. When LNGDT[received DT[5:0]] is 1, data is determined as a long packet and it is determined as a short packet when 0. According to the MIPI standard, DT = H'00 to H'0F is defined as a short packet and DT = H'10 to H'3F is defined as a long packet; to satisfy these conditions, the initial values of LNGDT are 0 in bits 0 to 15 and 1 in bits 16 to 63.

47.2.10 Long Data Type Setting Register 1 (LNGDT1)



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LNGDT1	H'FFFF FFFF	R/W	Long data type setting 1 0: Short packet 1: Long packet These bits set the long packet or short packet as the data type (assigns H'3F to H'20).

47.2.11 Interrupt Enable Register (INTEN)



Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	IEN[28]	0	R/W	Interrupt 28 (INT_LESS_THAN_WC) enable 0: Disables the interrupt. 1: Enables the interrupt.
27	IEN[27]	0	R/W	Interrupt 27 (INT_AFIFO_OF) enable 0: Disables the interrupt. 1: Enables the interrupt.
26 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	IEN[20]	0	R/W	Interrupt 20 (INT_VD_START) enable 0: Disables the interrupt. 1: Enables the interrupt.
19	IEN[19]	0	R/W	Interrupt 19 (INT_VD_END) enable 0: Disables the interrupt. 1: Enables the interrupt.
18	IEN[18]	0	R/W	Interrupt 18 (INT_SHP_STB) enable 0: Disables the interrupt. 1: Enables the interrupt.
17	IEN[17]	0	R/W	Interrupt 17 (INT_FSFE) enable 0: Disables the interrupt. 1: Enables the interrupt.
16	IEN[16]	0	R/W	Interrupt 16 (INT_LNP_STB) enable 0: Disables the interrupt. 1: Enables the interrupt.
15	IEN[15]	0	R/W	Interrupt 15 (INT_CRC_ERR) enable 0: Disables the interrupt. 1: Enables the interrupt.
14	IEN[14]	0	R/W	Interrupt 14 (INT_HD_WC_ZERO) enable 0: Disables the interrupt. 1: Enables the interrupt.
13	IEN[13]	0	R/W	Interrupt 13 (INT_FRM_SEQ_ERR1) enable 0: Disables the interrupt. 1: Enables the interrupt.
12	IEN[12]	0	R/W	Interrupt 12 (INT_FRM_SEQ_ERR0) enable 0: Disables the interrupt. 1: Enables the interrupt.
11	IEN[11]	0	R/W	Interrupt 11 (INT_ECC_ERR) enable 0: Disables the interrupt. 1: Enables the interrupt.
10	IEN[10]	0	R/W	Interrupt 10 (INT_ECC_CRCT_ERR) enable 0: Disables the interrupt. 1: Enables the interrupt.

Bit	Bit Name	Initial Value	R/W	Description
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	IEN[7]	0	R/W	Interrupt 7 (INT_ULPS_START) enable 0: Disables the interrupt. 1: Enables the interrupt.
6	IEN[6]	0	R/W	Interrupt 6 (INT_ULPS_END) enable 0: Disables the interrupt. 1: Enables the interrupt.
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4	IEN[4]	0	R/W	Interrupt 4 (INT_ERRSOTHS) enable 0: Disables the interrupt. 1: Enables the interrupt.
3	IEN[3]	0	R/W	Interrupt 3 (INT_ERRSOTSYNCHS) enable 0: Disables the interrupt. 1: Enables the interrupt.
2	IEN[2]	0	R/W	Interrupt 2 (INT_ERRESC) enable 0: Disables the interrupt. 1: Enables the interrupt.
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	IEN[0]	0	R/W	Interrupt 0 (INT_ERRCONTROL) enable 0: Disables the interrupt. 1: Enables the interrupt.

Note 1. The setting of the effective bit of this register becomes applicable the instant the register is set.
Note 2. Do not change the settings in this register while the CSI2 is operating.

47.2.12 Interrupt Source Mask Register (INTCLOSE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	ICL[28:27]		—	—	—	—	—	—	ICL[20:16]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

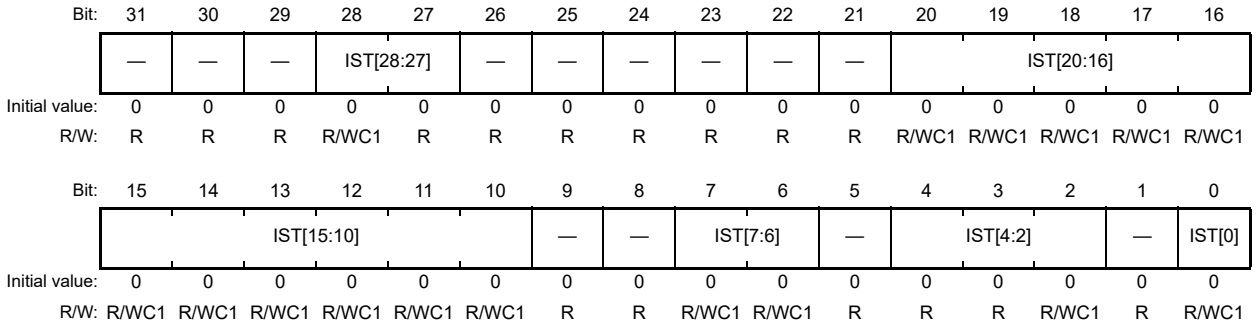
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ICL[15:10]						—	—	ICL[7:6]		—	ICL[4:2]			—	ICL[0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	ICL[28]	0	R/W	Interrupt 28 (INT_LESS_THAN_WC) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
27	ICL[27]	0	R/W	Interrupt 27 (INT_AFIFO_OF) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
26 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	ICL[20]	0	R/W	Interrupt 20 (INT_VD_START) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
19	ICL[19]	0	R/W	Interrupt 19 (INT_VD_END) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
18	ICL[18]	1	R/W	Interrupt 18 (INT_SHP_STB) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
17	ICL[17]	0	R/W	Interrupt 17 (INT_FSFE) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
16	ICL[16]	0	R/W	Interrupt 16 (INT_LNP_STB) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
15	ICL[15]	0	R/W	Interrupt 15 (INT_CRC_ERR) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
14	ICL[14]	0	R/W	Interrupt 14 (INT_HD_WC_ZERO) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
13	ICL[13]	0	R/W	Interrupt 13 (INT_FRM_SEQ_ERR1) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
12	ICL[12]	0	R/W	Interrupt 12 (INT_FRM_SEQ_ERR0) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
11	ICL[11]	0	R/W	Interrupt 11 (INT_ECC_ERR) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
10	ICL[10]	0	R/W	Interrupt 10 (INT_ECC_CRCT_ERR) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.

Bit	Bit Name	Initial Value	R/W	Description
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	ICL[7]	0	R/W	Interrupt 7 (INT_ULPS_START) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
6	ICL[6]	0	R/W	Interrupt 6 (INT_ULPS_END) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4	ICL[4]	0	R/W	Interrupt 4 (INT_ERRSOTHS) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
3	ICL[3]	0	R/W	Interrupt 3 (INT_ERRSOTSYNCHS) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
2	ICL[2]	0	R/W	Interrupt 2 (INT_ERRESC) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	ICL[0]	0	R/W	Interrupt 0 (INT_ERRCONTROL) source mask 0: Enables the interrupt source. 1: Masks the interrupt source.

Note: The setting of the effective bit of this register becomes applicable the instant the register is set.
Note: The settings of this register are optional. When INTCLOSE is set, the following sequence is recommended before “Start of PHY” operation.
1. set INTCLOSE
2. clear INTSTATE and INTERRSTATE
3. set INTEN

47.2.13 Interrupt Status Monitor Register (INTSTATE)



Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	IST[28]	0	R/WC1	Interrupt 28 (INT_LESS_THAN_WC) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 28 is an error interrupt that is generated when the length of payload data of a long packet is less than the WC value.
27	IST[27]	0	R	Interrupt 27 (INT_AFIFO_OF) monitor (read-only) 0: Normal 1: The interrupt has been generated. Interrupt 27 is generated in response to an overflow of the asynchronous FIFO, which holds the HS data sent from the PHY. IST[27] does not have the corresponding register (DFF); the OR of the lanes of the interrupt error status monitoring register (IEST[8]) generates the interrupt. Therefore, the interrupt is cleared using the interrupt error status monitoring register. Check the relevant registers such as the PHY data lane monitoring register since synchronization between lanes may not be taken or data may not be received in a specific lane.
26 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	IST[20]	0	R/WC1	Interrupt 20 (INT_VD_START) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 20 is generated in response to the start of VD output from the CSI2 (a frame start interrupt). It is issued in synchronization with the rising edge of the CSIR_VD signal.
19	IST[19]	0	R/WC1	Interrupt 19 (INT_VD_END) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 19 is generated in response to the end of VD output from the CSI2 (a frame end interrupt). It is issued in synchronization with the falling edge of the CSIR_VD signal.

Bit	Bit Name	Initial Value	R/W	Description
18	IST[18]	0	R/WC1	Interrupt 18 (INT_SHP_STB) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 18 is a short packet reception interrupt. It is issued upon reception of short packets such as frame start and frame end packets.
17	IST[17]	0	R/WC1	Interrupt 17 (INT_FSFE) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 17 is a frame packet reception interrupt. It is issued upon reception of a frame start or frame end packet among short packets.
16	IST[16]	0	R/WC1	Interrupt 16 (INT_LNP_STB) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 16 is a long packet reception interrupt. It is issued upon reception of long packets such as YUV data.
15	IST[15]	0	R/WC1	Interrupt 15 (INT_CRC_ERR) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 15 is a CRC error interrupt. It is issued if there is a data error in a long packet. To perform the CRC check, set the CRC_EN bit in the CHKSUM register to 1.
14	IST[14]	0	R/WC1	Interrupt 14 (INT_HD_WC_ZERO) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 14 is a WC (word count) zero interrupt. It is issued when the WC value in the packet header of a long packet is zero.
13	IST[13]	0	R/WC1	Interrupt 13 (INT_FRM_SEQ_ERR1) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 13 is a frame sequence error 1 interrupt. It is issued when an illegal Frame End packet is received. Specifically, it is issued in the following cases (assuming that the same channel is used): - A Frame End packet is received before a Frame Start packet is received. - Frame End packets are continuously received.
12	IST[12]	0	R/WC1	Interrupt 12 (INT_FRM_SEQ_ERR0) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 12 is a frame sequence error 0 interrupt. It is issued when an illegal Frame Start packet is received. Specifically, it is issued in the following case (assuming that the same channel is used): - Frame Start packets are continuously received.

Bit	Bit Name	Initial Value	R/W	Description
11	IST[11]	0	R/WC1	Interrupt 11 (INT_ECC_ERR) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 11 is an ECC error interrupt. It is issued when two or more bits of errors are found in a packet header. To perform the ECC processing, set the ECC_EN bit in the CHKSUM register to 1.
10	IST[10]	0	R/WC1	Interrupt 10 (INT_ECC_CRCT_ERR) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 10 is an ECC 1-bit correction interrupt. It is issued when 1-bit correction is performed for a packet header during ECC processing. To perform the ECC processing, set the ECC_EN bit in the CHKSUM register to 1.
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	IST[7]	0	R/WC1	Interrupt 7 (INT_ULPS_START) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 7 is an ultra-low power data transfer start interrupt. This bit is set to high when any of the lanes shifts to the ULP state. For the lane state, see the description of the PHY data lane monitoring register (PHDLM). It is need to write to 1’b1 after initial sequence of PHY.
6	IST[6]	0	R/WC1	Interrupt 6 (INT_ULPS_END) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 6 is an ultra-low power data transfer end interrupt. This bit is set to high when no lanes are in the ULP state. For the lane state, see the description of the PHY data lane monitoring register (PHDLM). It is need to write to 1’b1 after initial sequence of PHY.
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4	IST[4]	0	R	Interrupt 4 (INT_ERRSOTHS) monitor (read-only) 0: Normal 1: The interrupt has been generated. Interrupt 4 is a synchronized SOT (start of transfer) error interrupt during HS reception. This bit is set to high when an error is detected in any of the lanes. For the lane state, see the description of the interrupt error status monitoring register (INTERRSTATE). IST[4] does not have the corresponding register (DFF); the OR of the lanes of the interrupt error status monitoring register (IEST[5:4]) generates the interrupt. Since the ERRSOTHS_[3-0] input from the PHY are pulse signals, the state are not retained. Therefore, the interrupt error status register is provided as the interrupt status register for each lane. Therefore, the interrupt is cleared using the interrupt error status monitoring register.

Bit	Bit Name	Initial Value	R/W	Description
3	IST[3]	0	R	Interrupt 3 (INT_ERRSOTSYNCHS) monitor (read-only) 0: Normal 1: The interrupt has been generated. Interrupt 3 is a non-synchronized SOT (start of transfer) error interrupt during HS reception. This bit is set to high when an error is detected in any of the lanes. For the lane state, see the description of the interrupt error status monitoring register (INTERRSTATE). IST[3] does not have the corresponding register (DFF); the OR of the lanes of the interrupt error status monitoring register (IEST[5:4]) generates the interrupt. Since the ERRSOTSYNCHS [3-0] input from the PHY are pulse signals, the state are not retained. Therefore, the interrupt error status register is prepared as the interrupt status register for each lane. Therefore, the interrupt is cleared using the interrupt error status monitoring register.
2	IST[2]	0	R/WC1	Interrupt 2 (INT_ERRESC) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 2 is an escape mode entry error interrupt. It is issued when an unrecognizable escape entry command is received. This bit is set to high when an error is detected in any of the lanes. For the lane state, see the description of the PHY ESC error monitoring register (PHEERM). Even after the interrupt is cleared, the PHEERM register retains the high- level state until a stop state is received.
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	IST[0]	0	R/WC1	Interrupt 0 (INT_ERRCONTROL) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Interrupt 0 is a PHY control error interrupt. It is issued when a transition is made to an incorrect line state. This bit is set to high when an error is detected in any of the lanes. For the lane state, see the description of the PHY ESC error monitoring register (PHEERM). Even after the interrupt is cleared, the PHEERM register retains the high-level state until a stop state is received.

47.2.14 Interrupt Error Status Monitor Register (INTERRSTATE)

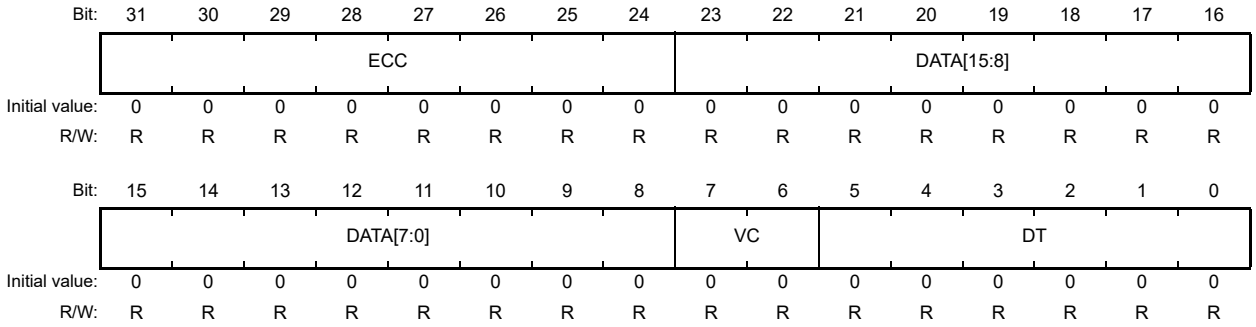
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	IEST [13]	IEST [12]	—	—	—	IEST[8]	—	—	IEST[5:4]		—	—	IEST[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/WC1	R/WC1	R	R	R	R/WC1	R	R	R/WC1	R/WC1	R	R	R/WC1	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	IEST[13]	0	R/WC1	Error interrupt 13 (INT_FIFO_OF_1) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Error interrupt 13 is generated in response to an overflow of the synchronous FIFO, which holds HS data sent from the PHY block. If the setting of ICL[27] in the interrupt source mask register is 1, INT_FIFO_OF_1 is cleared.
12	IEST[12]	0	R/WC1	Error interrupt 12 (INT_FIFO_OF_0) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Error interrupt 12 is generated in response to an overflow of the synchronous FIFO, which holds HS data sent from the PHY block. If the setting of ICL[27] in the interrupt source mask register is 1, INT_FIFO_OF_0 is cleared.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	IEST[8]	0	R/WC1	Error interrupt 8 (INT_AFIFO_OF_0) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Error interrupt 8 is generated in response to an overflow of the asynchronous FIFO, which holds HS data sent from the PHY block. If the setting of ICL[27] in the interrupt source mask register is 1, INT_AFIFO_OF_0 is cleared to 0.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5	IEST[5]	0	R/WC1	Error interrupt 5 (INT_ERRSOTHS_1) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Note: When the connected PHY is single-lane, the interrupt is not generated. Error interrupt 5 is generated in response to a synchronized start-of-transmission (SoT) error during HS reception. This bit is set to 1 when an error is detected in lane 1. If the setting of ICL[4] in the interrupt source mask register is 1, INT_ERRSOTHS_1 is cleared to 0.
4	IEST[4]	0	R/WC1	Error interrupt 4 (INT_ERRSOTHS_0) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Error interrupt 4 is generated in response to a synchronized synchronized start-of-transmission (SoT) error during HS reception. This bit is set to 1 when an error is detected in lane 0. If the setting of ICL[4] in the interrupt source mask register is 1, INT_ERRSOTHS_0 is cleared to 0.
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	IEST[1]	0	R/WC1	Error interrupt 1 (INT_ERRSOTSYNCHS_1) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Note: When the connected PHY is single-lane, the interrupt is not generated. Error interrupt 1 is generated in response to a non-synchronized start-of-transmission (SoT) during HS reception. This bit is set to 1 when an error is detected in lane 1. If the setting of ICL[3] in the interrupt source mask register is 1, INT_ERRSOTSYNCHS_1 is cleared to 0.
0	IEST[0]	0	R/WC1	Error interrupt 0 (INT_ERRSOTSYNCHS_0) monitor Writing to this bit leads to clearing of the interrupt. 1: Clears the interrupt. When read, the bit reflects the state. 0: Normal 1: The interrupt has been generated. Error interrupt 0 is generated in response to a non-synchronized start-of-transmission (SoT) during HS reception. This bit is set to 1 when an error is detected in lane 0. If the setting of ICL[3] in the interrupt source mask register is 1, INT_ERRSOTSYNCHS_0 is cleared to 0.

47.2.15 Short Packet Data Register (SHPDAT)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ECC	H'00	R	ECC data received
23 to 8	DATA	H'0000	R	Short packet data received
7, 6	VC	B'00	R	Virtual channel data received
5 to 0	DT	H'00	R	Data type received

SHPDAT is used to read the received short packet data. The value of this register is stored in the FIFO and the number of packets stored can be checked by reading the short packet count register (SHPCNT). Short packets with the data types H'00 (frame start), H'01 (frame end), and H'08 (Generic Short Packet Code 1, DSI End of Transmission) are not stored.

47.2.16 Short Packet Count Register (SHPCNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OVF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	NUM			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	OVF	0	R	FIFO overflow bit FIFO has overflowed. The received data may have been lost.
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	NUM	H'0	R	Stored short packet count The number of short packets stored in the FIFO can be read. Up to nine packets can be stored.

47.2.17 LINK Operation Control Register (LINKCNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MONITOR_EN	—	—	—	—	—	REG_MONI_PACT_EN	—	—	—	—	—	—	—	—	—
Initial value:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	MONITOR_EN	1	R/W	Monitoring function control 0: Disable 1: Enable This bit updates the monitor state to be read by the registers.
30 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	REG_MONI_PACT_EN	0	R/W	Monitoring packet counter enable 0: Disable 1: Enable This bit controls the packet counter for monitoring.
24 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

47.2.18 Lane Swap Register (LSWAP)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	L1SEL		L0SEL	
Initial value:	0	0	0	0	0	0	0	0	1	1	1	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 5	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3, 2	L1SEL	H'1	R/W	LINK lane 1 select bits H'0: Uses PHY lane 0 H'1: Uses PHY lane 1 H'2: Setting prohibited H'3: Setting prohibited Note: When the connected PHY is single-lane, this setting has no effect. These bits select a PHY lane to be assigned to LINK lane 1. These bits should be set before the start of PHY operation and should not be modified during operation.
1, 0	L0SEL	H'0	R/W	LINK lane 0 select bits H'0: Uses PHY lane 0 H'1: Uses PHY lane 1 H'2: Setting prohibited H'3: Setting prohibited These bits select a PHY lane to be assigned to LINK lane 0. These bits should be set before the start of PHY operation and should not be modified during operation.

Note: The setting of the effective bit of this register becomes applicable the instant the register is set. Set the ENABLE_0 bit in PHYCNT to 1 to enable modifying the value of this register.

47.2.19 PHY ESC Error Monitor Register (PHEERM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CL_ERR CONTROL	—	—	ERRESC_1	ERRESC_0	—	—	—	—	—	—	ERRCONTROL_1	ERRCONTROL_0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	CL_ERR CONTROL	0	R	Clock lane status control error bit If an incorrect line state is entered during entry to the HS or escape mode, this bit is set to the high level, and the current state is retained until a transition is made to the LP-00, LP-01, LP-10, or LP-11 state.
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	ERRESC_1	0	R	Lane 1 escape mode error bit Note: When 1-lane PHY is connected, the error is not generated. If an unrecognizable escape entry command is received, this bit is set to the high level, and the current state is retained until a transition is made to the LP-00, LP-01, LP-10, or LP-11 state.
8	ERRESC_0	0	R	Lane 0 escape mode error bit If an unrecognizable escape entry command is received, this bit is set to the high level, and the current state is retained until a transition is made to the LP-00, LP-01, LP-10, or LP-11 state.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	ERRCONTROL_1	0	R	Lane 1 status control error bit Note: When 1-lane PHY is connected, the error is not generated. If an incorrect line state is entered during entry to HS or escape mode, this bit is set to the high level, and the current state is retained until a transition is made to the LP-00, LP-01, LP-10, or LP-11 state.
0	ERRCONTROL_0	0	R	Lane 0 status control error bit If an incorrect line state is entered during entry to HS or escape mode, this bit is set to the high level, and the current state is retained until a transition is made to the LP-00, LP-01, LP-10, or LP-11 state.

47.2.20 PHY Clock Lane Monitor Register (PHCLM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	ULPSACTIVENOTCLK	RXULPSCLKNOT	RXCLKACTIVEHS	STOPSTATECLK
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	ULPSACTIVENOTCLK	1	R	This active-low signal is asserted to indicate that the clock lane is in the ultra low power (ULP) state. This bit is set to 0 to when the clock lane is placed in the ULP state, and is set to 1 to when the Mark-1 state (LP-10) is observed.
2	RXULPSCLKNOT	1	R	This active-low signal is asserted to indicate that the clock lane is in the ULP state. This bit is set to 0 when the clock lane is placed in the ULP state, and is set to 1 when the stop state (LP-11) is observed.
1	RXCLKACTIVEHS	0	R	This bit being 1 indicates that the clock lane is receiving the HS DRR clock.
0	STOPSTATECLK	0	R	This bit being 1 indicates that the clock lane is in the stop state.

47.2.21 PHY Data Lane Monitor Register (PHDLM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	ULPSACTIVENOT_1	ULPSACTIVENOT_0	—	—	RXULPSESC_1	RXULPSESC_0	—	—	—	—	—	—	STOPSTATEDATA_1	STOPSTATEDATA_0
Initial value:	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	ULPSACTIVENOT_1	1	R	This active-low signal is asserted to indicate that lane 1 is in the ultra low power (ULP) state. Note: When the connected PHY is single-lane, this bit is fixed to 1. This bit is set to 0 when lane 1 is placed in the ULP state, and is set to 1 when the Mark-1 state (LP-10) is observed.
12	ULPSACTIVENOT_0	1	R	This active-low signal is asserted to indicate that lane 0 is in the ULP state. This bit is set to 0 when lane 0 is placed in the ULP state, and is set to 1 when the Mark-1 state (LP-10) is observed.
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	RXULPSESC_1	0	R	This bit indicates that lane 1 is in the ULP state. Note: When the connected PHY is single-lane, this bit is fixed to 0. This bit is set to 1 when lane 1 is placed in the ULP state, and is set to 0 to when the stop state (LP-11) is observed.
8	RXULPSESC_0	0	R	This bit indicates that lane 0 is in the ULP state. This bit is set to 1 to indicate that lane 0 is placed in the ULP state, and is set to 0 to indicate that the stop state (LP-11) is observed.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	STOPSTATEDATA_1	0	R	This bit being 1 indicates that lane 1 is in the stop state. Note: When the connected PHY is single-lane, this bit is fixed to 0.
0	STOPSTATEDATA_0	0	R	This bit being 1 indicates that lane 0 is in the stop state.

47.2.22 Packet Header 0 Monitor Register 0 (PH0M0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	WC[15:8]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WC[7:0]								VC		DT					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 8	WC	H'0000	R	Word count data received
7, 6	VC	B'00	R	Virtual channel data received
5 to 0	DT	H'00	R	Data type data received

PH0M0 is used to monitor the received packet header.

47.2.23 Packet Header 0 Monitor Register 1 (PH0M1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PH_CNT															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	PH_CNT	H'0000	R	PH0M0 register consecutive reception count These bits monitor the number of times of consecutive receptions for the PH0M0 register.

47.2.24 Packet Header 1 Monitor Register 0 (PH1M0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	WC[15:8]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WC[7:0]								VC		DT					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 8	WC	H'0000	R	Word count data received
7, 6	VC	B'00	R	Virtual channel data received
5 to 0	DT	H'00	R	Data type data received

PH1M0 is used to monitor the received packet header (previous data of PH0M0).

47.2.25 Packet Header 1 Monitor Register 1 (PH1M1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PH_CNT															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	PH_CNT	H'0000	R	PH1M0 register consecutive reception count These bits monitor the number of times of consecutive receptions for the PH1M0 register.

47.2.26 Packet Header 2 Monitor Register 0 (PH2M0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	WC[15:8]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WC[7:0]								VC		DT					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 8	WC	H'0000	R	Word count data received
7, 6	VC	B'00	R	Virtual channel data received
5 to 0	DT	H'00	R	Data type data received

PH2M0 is used to monitor the received packet header (previous data of PH1M0).

47.2.27 Packet Header 2 Monitor Register 1 (PH2M1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PH_CNT															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	PH_CNT	H'0000	R	PH2M0 register consecutive reception count These bits monitor the number of times of consecutive receptions for the PH2M0 register.

47.2.28 Packet Header 3 Monitor Register 0 (PH3M0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	WC[15:8]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WC[7:0]								VC		DT					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 8	WC	H'0000	R	Word count data received
7, 6	VC	B'00	R	Virtual channel data received
5 to 0	DT	H'00	R	Data type data received

PH3M0 is used to monitor the received packet header (previous data of PH2M0).

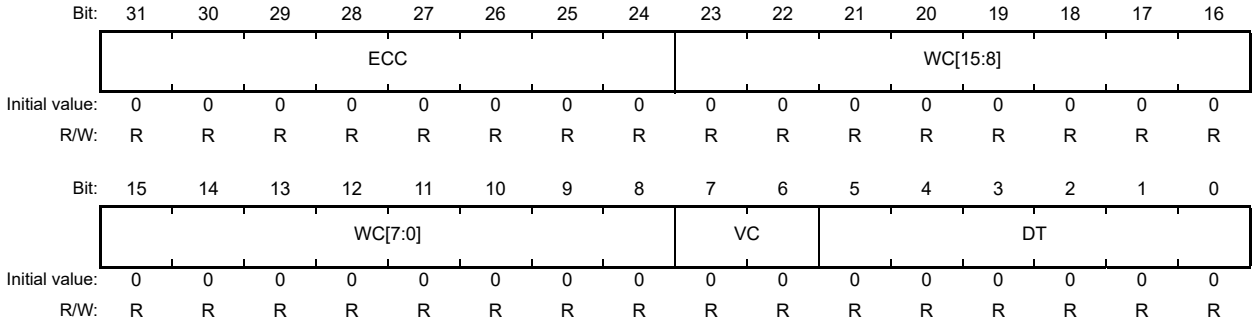
47.2.29 Packet Header 3 Monitor Register 1 (PH3M1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PH_CNT															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	PH_CNT	H'0000	R	PH3M0 register consecutive reception count These bits monitor the number of times of consecutive receptions for the PH3M0 register.

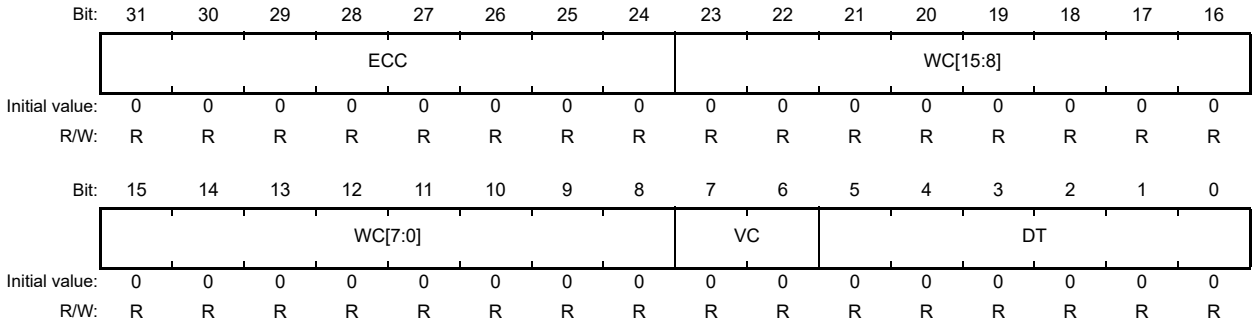
47.2.30 Packet Header R Monitor Register 0 (PHRM0)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ECC	H'00	R	ECC data received
23 to 8	WC	H'0000	R	Word count data received
7, 6	VC	B'00	R	Virtual channel data received
5 to 0	DT	H'00	R	Data type data received

PHRM0 is used to monitor the currently received packet header.

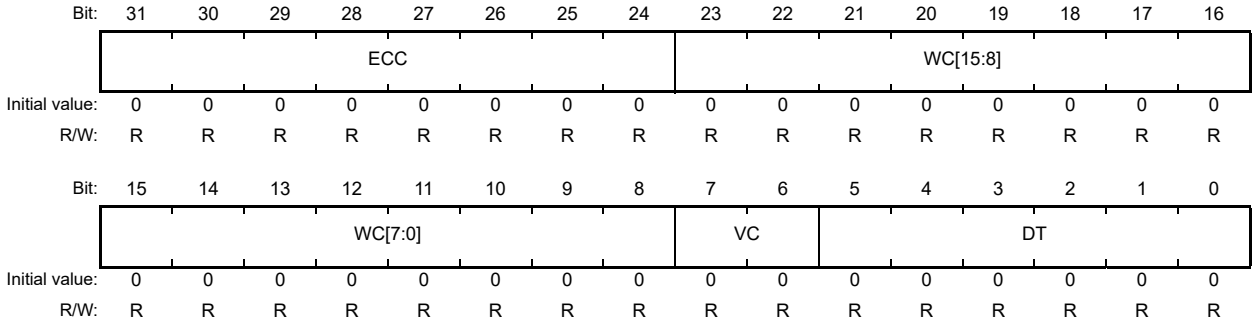
47.2.31 Packet Header R Monitor Register 1 (PHRM1)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ECC	H'00	R	ECC data received
23 to 8	WC	H'0000	R	Word count data received
7, 6	VC	B'00	R	Virtual channel data received
5 to 0	DT	H'00	R	Data type data received

PHRM1 is used to monitor the received packet header (previous data of PHRM0).

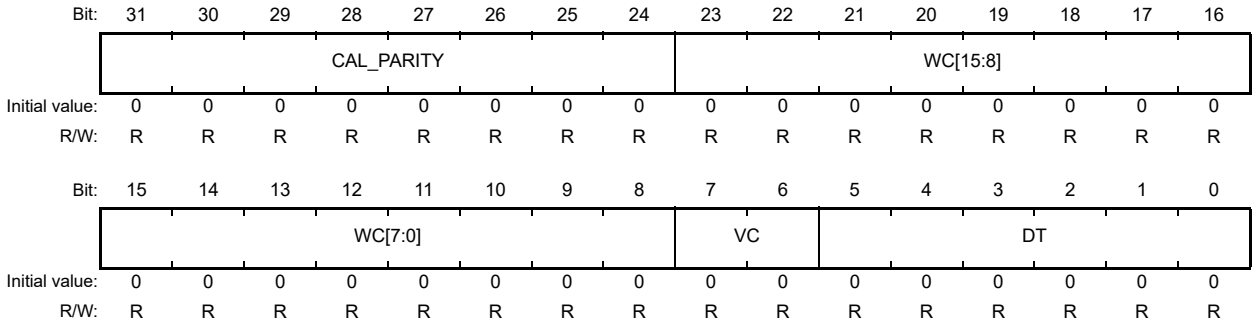
47.2.32 Packet Header R Monitor Register 2 (PHRM2)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ECC	H'00	R	ECC data received
23 to 8	WC	H'0000	R	Word count data received
7, 6	VC	B'00	R	Virtual channel data received
5 to 0	DT	H'00	R	Data type data received

PHRM2 is used to monitor the received packet header (previous data of PHRM1).

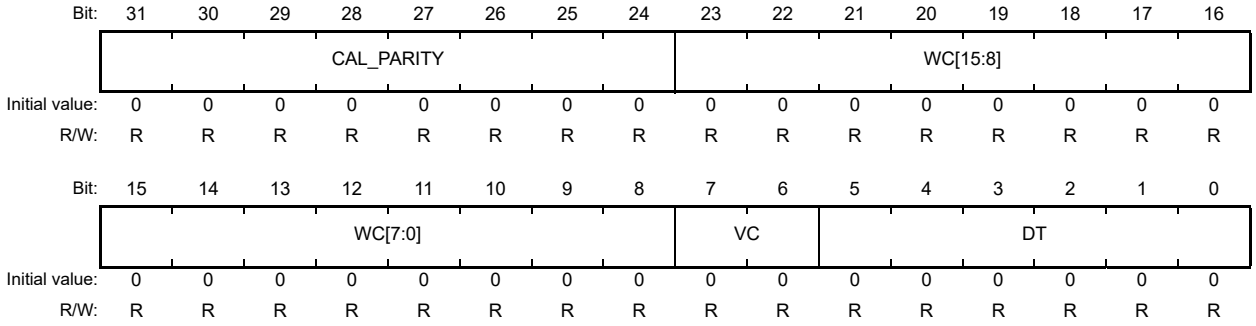
47.2.33 Packet Header C Monitor Register 0 (PHCM0)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CAL_PARITY	H'00	R	Parity calculated from the header
23 to 8	WC	H'0000	R	Received word count data after ECC correction
7, 6	VC	B'00	R	Received virtual channel data after ECC correction
5 to 0	DT	H'00	R	Received data type data after ECC correction

PHCM0 is used to monitor the currently received ECC-corrected packet header and calculated parity.

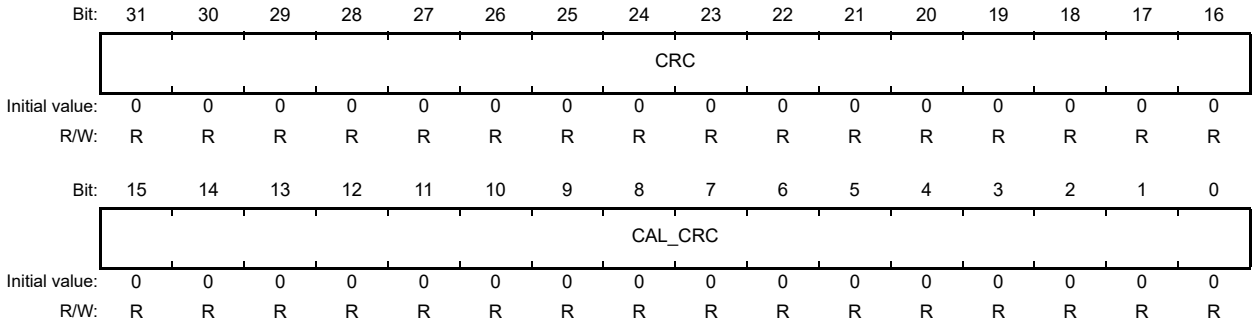
47.2.34 Packet Header C Monitor Register 1 (PHCM1)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CAL_PARITY	H'00	R	Parity calculated from the header
23 to 8	WC	H'0000	R	Received word count data after ECC correction
7, 6	VC	B'00	R	Received virtual channel data after ECC correction
5 to 0	DT	H'00	R	Received data type data after ECC correction

PHCM1 is used to monitor the currently received ECC-corrected packet header and calculated parity (previous data of PHCM0).

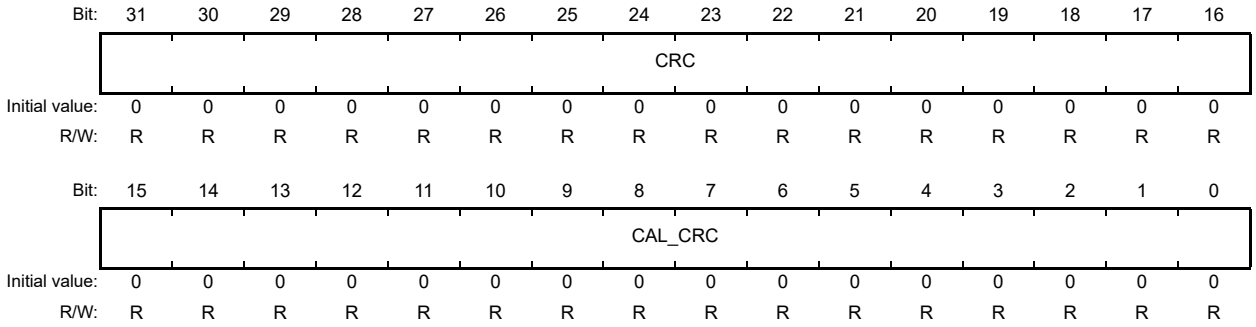
47.2.35 CRC Monitor Register 0 (CRCM0)



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	CRC	H'0000	R	Received CRC
15 to 0	CAL_CRC	H'0000	R	Received data type data after ECC correction

CRCM0 is used to monitor the CRC currently received and CRC calculated from the received data.

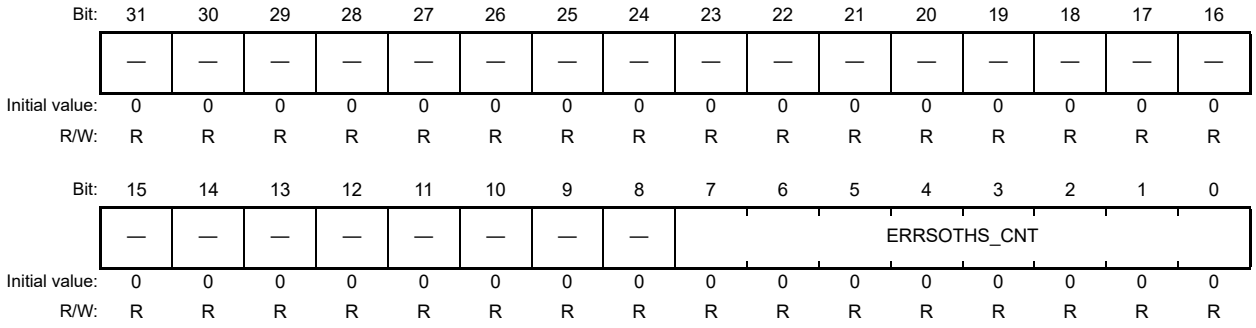
47.2.36 CRC Monitor Register 1 (CRCM1)



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	CRC	H'0000	R	Received CRC
15 to 0	CAL_CRC	H'0000	R	CRC result calculated from the received data

CRCM1 is used to monitor the CRC currently received and CRC calculated from the received data (previous data of CRCM0).

47.2.37 SOT Error Count Register (SERRCNT)



Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	ERRSOTHS_CNT	H'00	R	Synchronized SOT error count These bits monitor the number of synchronized SOT errors. The count is the sum of the errors on each lane.

47.2.38 SOTSYNC Error Count Register (SSERRCNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	ERRSOTSYNCHS			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	ERRSOTSYNCHS	H'0	R	Non-synchronized SOT error count These bits monitor the number of non-synchronized SOT errors. The count is the sum of the errors on each lane. After this error is input, no synchronization signal is generated. If the asynchronous FIFO for storing data from the PHY overflows here, the internal registers shift to the abnormal state. In this case, the state can only be restored by resetting the CSI2 control circuit using the software reset register (within this module) or inputting a software reset from an external module after resetting the PHY and stopping the normal lane data from the PHY.

47.2.39 ECC_CRCT Count Register (ECCCM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ECC_CRCT_CNT							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	ECC_CRCT_CNT	H'00	R	ECC 1-bit correction count These bits monitor the number of ECC 1-bit corrections.

47.2.40 ECC_ERR Count Register (ECECM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ECC_ERR_CNT							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	ECC_ERR_CNT	H'00	R	ECC error (two bits or more) count These bits monitor the number of ECC errors (of two bits or more).

47.2.41 CRC_ERR Count Register (CRCECM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CRC_ERR_CNT							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	CRC_ERR_CNT	H'00	R	CRC error count These bits monitor the number of CRC errors.

47.2.42 Line Register (LCNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LINE_CNT															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	H'0000	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	LINE_CNT	H'0000	R	Channel reception line These bits monitor the line currently received on channel.

47.2.43 Line Monitor Register (LCNTM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MONI_LINECNT															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	H'0000	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MONI_ LINECNT	H'0000	R	Channel received line count These bits monitor the number of lines of the most recently received frame on channel.

47.2.44 Frame Count Monitor Register (FCNTM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MONI_FCOUNT															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	H'0000	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MONI_ FCOUNT	H'0000	R	Channel received frame count These bits monitor the number of frames which have been received so far on channel.

47.2.45 PHY Data IN Monitor Register (PHYDIM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RXDATAHS_1								RXDATAHS_0							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	H'0000	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	RXDATAHS_1	H'00	R	Byte data received on data lane1 When 1-lane PHY is connected, these bits are fixed to 0.
7 to 0	RXDATAHS_0	H'00	R	Byte data received on data lane 0

PHYDIM is used to monitor byte data input from the PHY block.

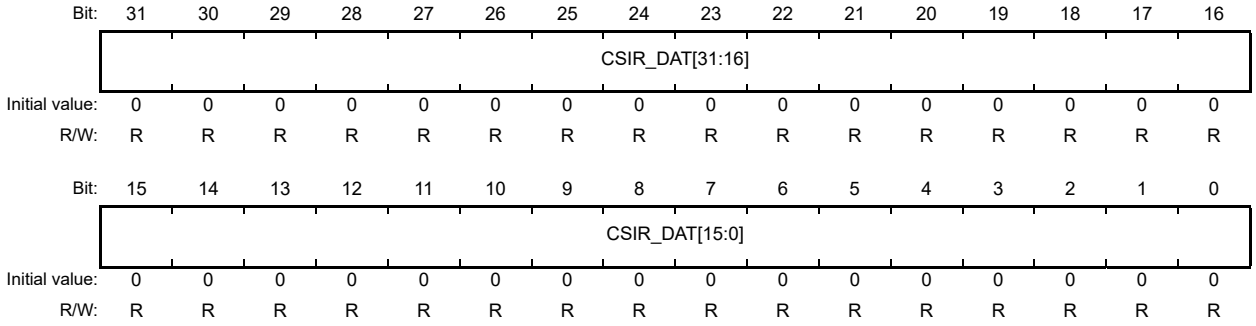
47.2.46 PHY Input Monitor Register (PHYIM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RXCLK_CNT	—	—	—	—	—	—	—	—	—	RXVALIDHS_1	RXVALIDHS_0	—	—	RXACTIVEHS_1	RXACTIVEHS_0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RXSYNCHS_1_CNT				RXSYNCHS_0_CNT			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

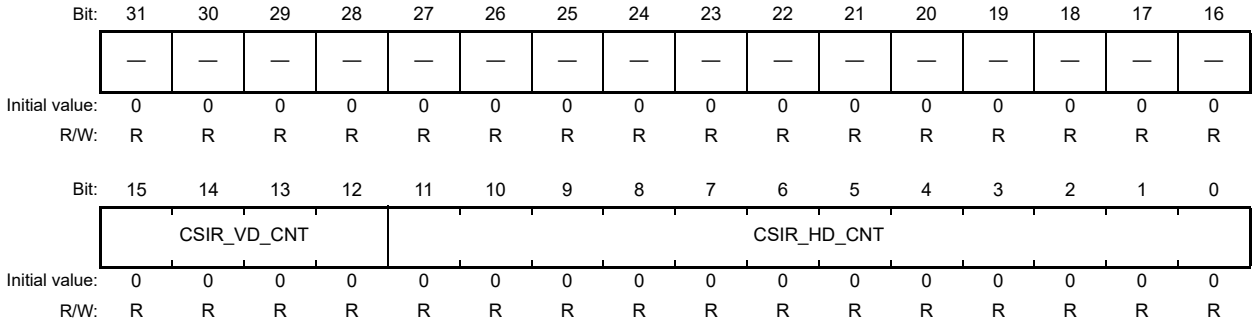
Bit	Bit Name	Initial Value	R/W	Description
31	RXCLK_CNT	0	R	RXCLK counter 0 This bit monitors signals toggled at the rising edge of RXCLK.
30 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	RXVALIDHS_1	0	R	Data lane 1 HS valid signal Note: When 1-lane PHY is connected, this bit is fixed to 0. This bit monitors the valid signal on lane 1 input from the PHY.
20	RXVALIDHS_0	0	R	Data lane 0 HS valid signal This bit monitors the valid signal on lane 0 input from the PHY.
19,18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	RXACTIVEHS_1	0	R	Data lane 1 HS active signal Note: When 1-lane PHY is connected, this bit is fixed to 0. This bit monitors the active signal on lane 1 input from the PHY.
16	RXACTIVEHS_0	0	R	Data lane 0 HS active signal This bit monitors the active signal on lane 0 input from the PHY.
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 4	RXSYNCHS_1_CNT	H'0	R	Data lane 1 HS synchronization signal count Note: When 1-lane PHY is connected, these bits are fixed to 0. These bits monitor the count of RSYNCHS_1 input from the PHY.
3 to 0	RXSYNCHS_0_CNT	H'0	R	Data lane 0 HS synchronization signal count These bits monitor the count of RSYNCHS_0 input from the PHY.

47.2.47 VIN Data Monitor Register (VINDM)



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CSIR_DAT [31:0]	H'0000_0000	R	Data output from CSI2 These bits monitor the CSIR_DAT signal output from the CSI2.

47.2.48 VIN Signal Monitor Register 1 (VINSM1)



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 12	CSIR_VD_ CNT	H'0	R	Count of VD output from CSI2 These bits count and monitor the vertical synchronization signals for channel output from CSI2.
11 to 0	CSIR_HD_ CNT	H'000	R	Count of HD output from CSI2 These bits count and monitor the HD signals for channel output from CSI2.

47.2.49 VIN Signal Monitor Register 3 (VINSM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIR_ERRE	CSIR_ERRC	CSIR_TAG		CSIR_FLD				CSIR_PEB				—	—	—	CSIR_PE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	CSIR_ERRE	0	R	ECC error output from CSI2 This bit monitors the ECC error signal output from CSI2.
14	CSIR_ERRC	0	R	CRC error output from CSI2 This bit monitors the CRC error signal output from CSI2.
13, 12	CSIR_TAG	B'00	R	TAG output from CSI2 These bits monitor the channel information output from CSI2.
11 to 8	CSIR_FLD	H'0	R	FLD output from CSI2 These bits monitor the field information output from CSI2.
7 to 4	CSIR_PEB	H'0	R	PEB output from CSI2 These bits monitor the byte enable signal output from CSI2.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	CSIR_PE	0	R	PE output from CSI2 This bit monitors the packet enable signal output from CSI2.

47.2.50 PHY Output Monitor Register (PHYOM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	ENABLE CLK	—	—	ENABLE_1	ENABLE_0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	ENABLECLK	0	R	Lane enable signal to the clock lane
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	ENABLE_1	0	R	Lane enable signal to data lane 1
0	ENABLE_0	0	R	Lane enable signal to data lane 0

PHYOM is used to monitor the lane enable signal output from the CSI2 to the PHY block.

47.2.51 Packet Header Monitor Registers 1 to 8 (PHM1 to PHM8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECC								WC[15:8]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WC[7:0]								VC		DT					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ECC	H'00	R	ECC data received
23 to 8	WC	H'0000	R	Word count data received
7, 6	VC	B'00	R	Virtual channel data received
5 to 0	DT	H'00	R	Data type data received

PHM1 to PHM8 are used to monitor received packet headers.
The first data are stored in PHM1, and then are shifted to PHM2 when the received packet header changes. After data have been stored in PHM8, data are stored in PHM1 again.

47.2.52 PHY Timing Register 1 (PHYTIM1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	T_INIT_SLAVE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	T_INIT_SLAVE	H'0000	R/W	MIPI D-PHY T_{INIT} parameter setting These bits specify the minimum duration of the INIT state.

47.2.53 PHY Timing Register 2 (PHYTIM2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	TCLK_MISS[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TCLK_SETTLE[5:0]						—	—	—	TCLK_PREPARE[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	TCLK_MISS	H'00	R/W	MIPI D-PHY T_{CLK_MISS} parameter setting These bits specify the period from detection of the absence of a clock until disabling of the HS-RX in the clock lane.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 8	TCLK_SETTLE	H'00	R/W	MIPI D-PHY T_{CLK_SETTLE} parameter setting These bits specify the period over which a transition to the HS state is to be ignored after the $T_{CLK\_PREPARE}$ period has begun in the clock lane.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	TCLK_PREPARE	H'00	R/W	MIPI D-PHY $T_{CLK\_PREPARE}$ parameter setting These bits specify the duration of the LP-00 state (immediately before entry to the HS-0 state) in the clock lane.

47.2.54 PHY Timing Register 3 (PHYTIM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	THS_SETTLE[5:0]						—	—	THS_PREPARE[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 8	THS_ SETTLE	H'00	R/W	MIPI D-PHY T_{HS_SETTLE} parameter setting These bits specify the period over which a transition to the HS state is to be ignored after the $T_{CLK\_PREPARE}$ period has begun in the data lane.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	THS_ PREPARE	H'00	R/W	MIPI D-PHY $T_{HS\_PREPARE}$ parameter setting These bits specify the duration of the LP-00 state (immediately before entry to the HS-0 state) in the data lane.

47.3 Operation

47.3.1 Transfer Rate

This module supports transfer rates from 80 Mbps to 1.0 Gbps.
Figure 47.3 shows a transfer at 1.0 Gbps.

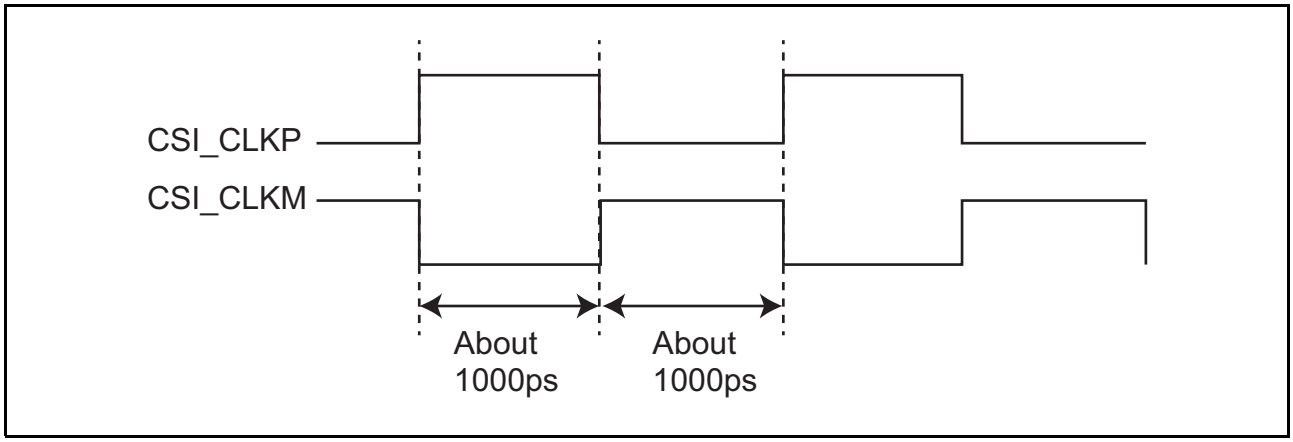


Figure 47.3 CSI-2 Transfer Rate

47.3.2 ECC 1-Bit Error Correction and 2-Bit or More Error Detection for Packet Headers

An interrupt can be generated upon ECC 1-bit error correction and 2-bit or more error detection.

Interrupt Source	Register Bit Name	Bit
ECC error in 2 or more bits	ECC_ERR	[11]
ECC 1-bit error correction	ECC_CRCT_ERR	[10]

The ECC_EN bit in the checksum control register (CHKSUM) is used to enable or disable this function.

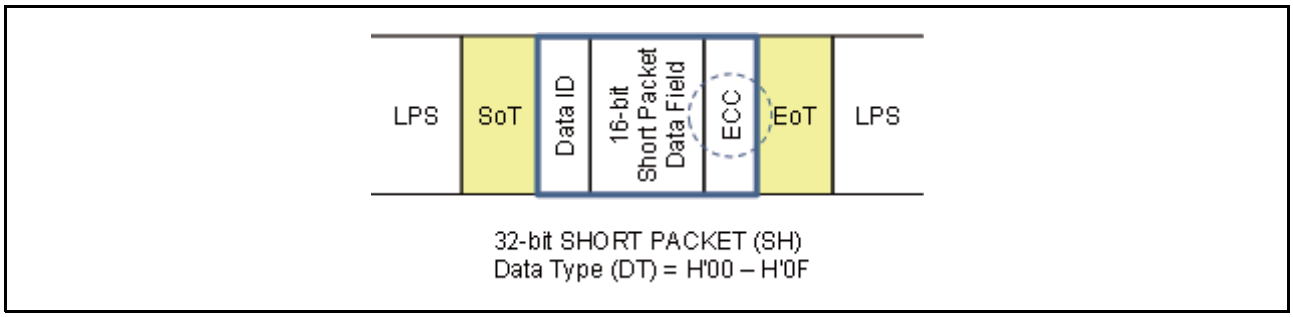


Figure 47.4 MIPI CSI-2 Short Packet

47.3.3 CRC Error Detection for the Payload Data

This module generates an interrupt in response to detection of a CRC error.

Interrupt Source	Register Bit Name	Bit
CRC error	CRC_ERR	[15]

The CRC_EN bit in the checksum control register (CHKSUM) is used to enable or disable this function.

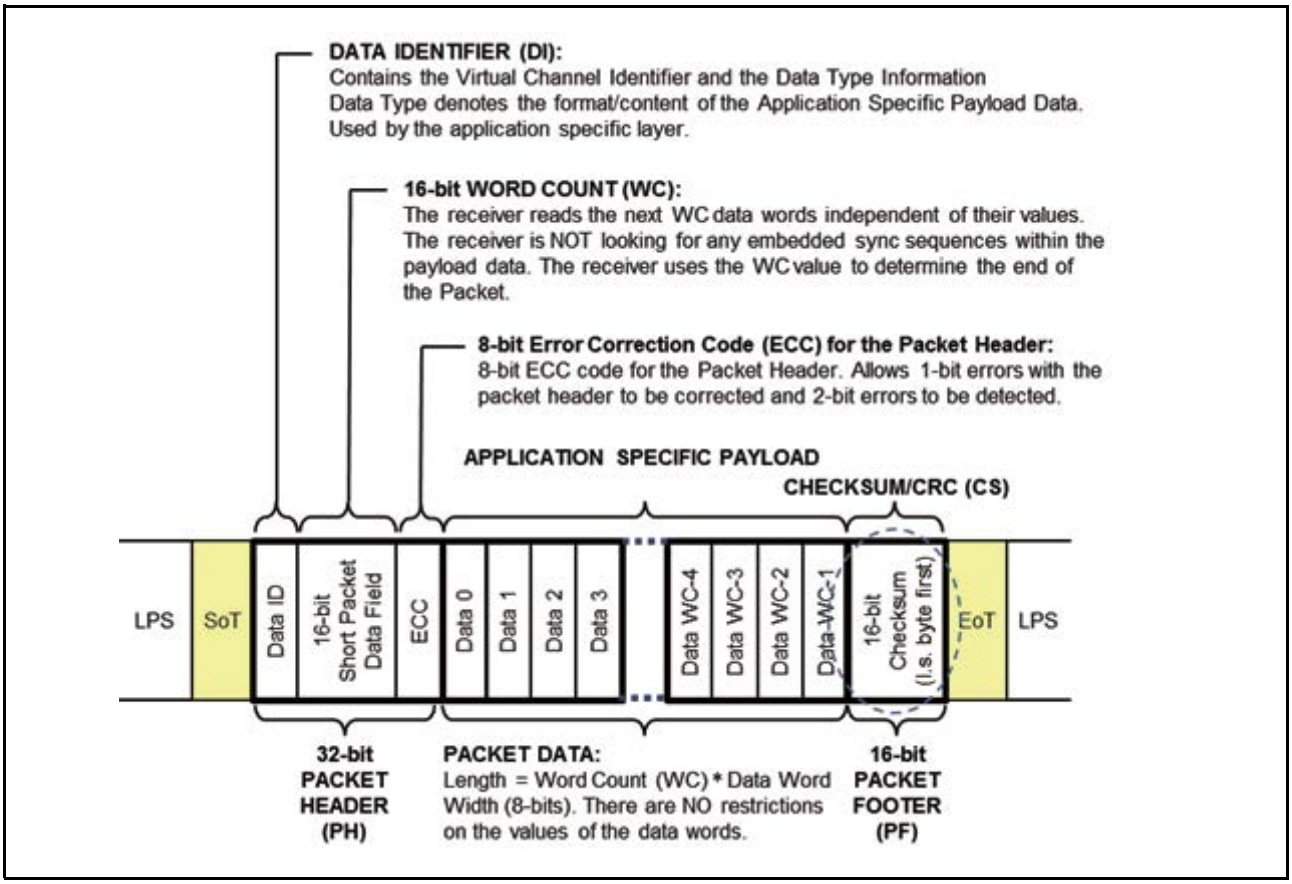


Figure 47.5 MIPI CSI-2 Long Packet

47.3.4 Generation of Vertical Sync (VD), Horizontal Sync (HD), and Field (FLD) Signals

(1) VD

When a frame start short packet is received, VD is driven high. When a frame end short packet is received, VD is driven low. (This applies to the channel specified with the VC value in the packet header.)

(2) HD

When the sync signal of a long packet (SOT) is received, HD is driven high. When the number of the received long packets reaches the WC value in the packet header, HD is driven low. The frame start short packet of the reception channel should be received in advance.

(3) FLD

When a frame start short packet is received, if the field value of the short packet data satisfies the condition specified with the field detection control register (FLD), FLD is driven high. When a frame end short packet is received, FLD is driven low. (This applies to the channel specified with the VC value in the packet header.)

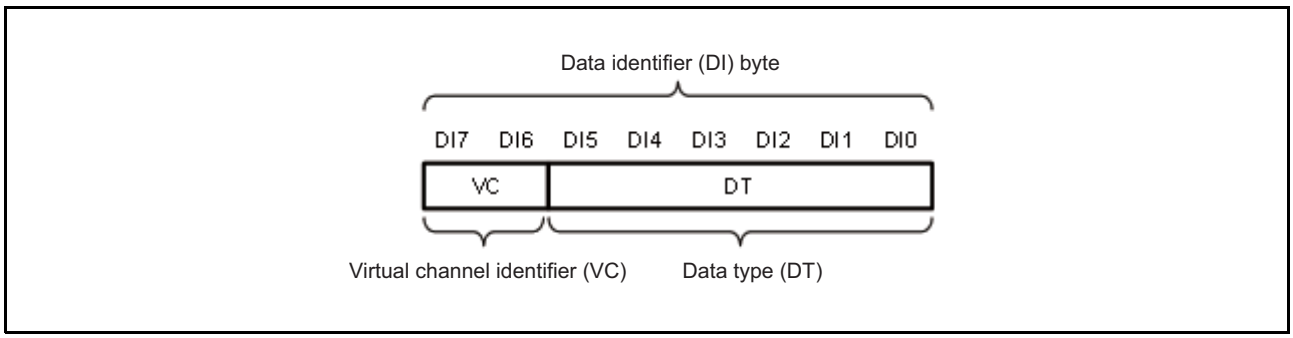


Figure 47.6 MIPI CSI-2 DI

Table 47.3 MIPI CSI-2 Short Packet Data Type Codes

Data Type	Description
H'00	Frame Start Code
H'01	Frame End Code
H'02	Line Start Code (Optional)
H'03	Line End Code (Optional)
H'04 to H'07	Reserved

47.3.5 Single-Channel Output

As shown in Figure 47.7 and Figure 47.8, when an interleaved transfer of virtual channels is received, the output virtual channel can be selected. The output virtual channel can be selected among the four virtual channels between 0 and 3. The receiving virtual channel can be changed with the register setting (SEL_VC bit of the channel data type selection register (VCDT)).

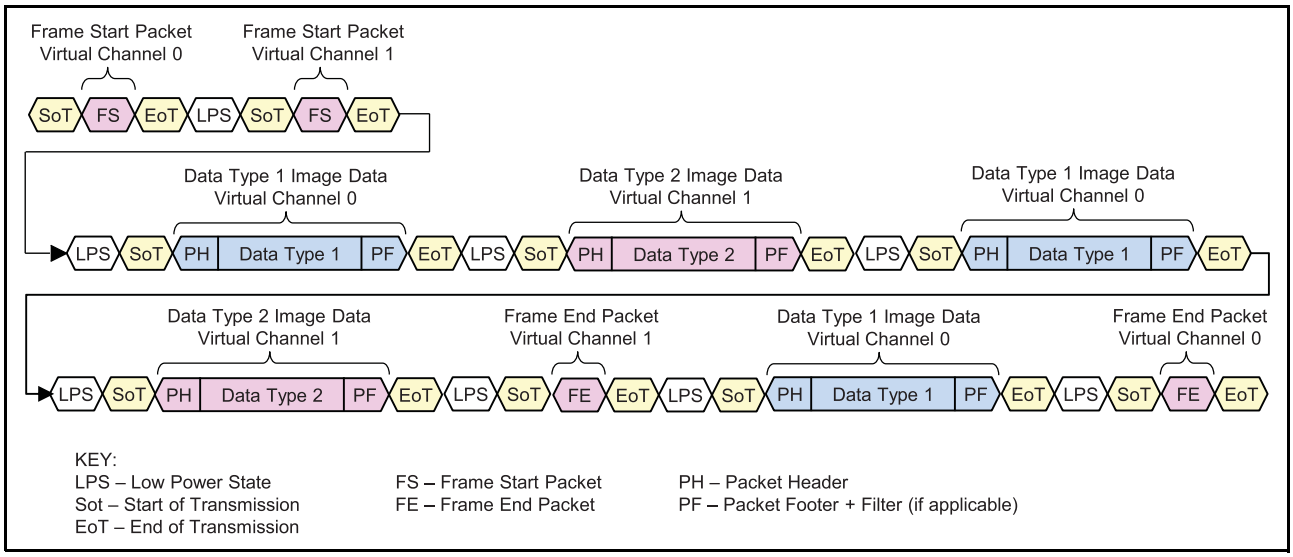


Figure 47.7 (1) MIPI CSI-2 Interleaved Transmission 1

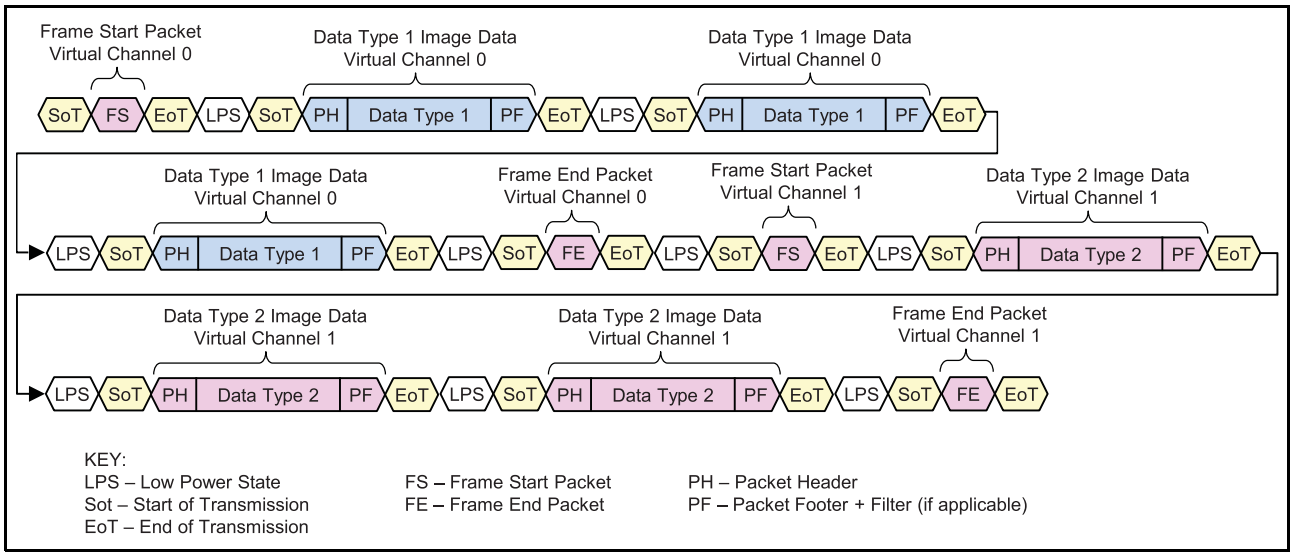


Figure 47.8 (2) MIPI CSI-2 Interleaved Transmission 2

47.3.6 Interrupts

Table 47.4 Interrupt Registers

Function	Register Name
Monitors interrupt state.	INTSTATE
Monitors interrupt error state.	INTERRSTATE
Enables interrupts.	INTEN
Masks interrupt sources.	INTCLOSE

As shown in Figure 47.9, the AFIFO (asynchronous FIFO) overflow and ERRSOT* (HS synchronization error) have corresponding interrupt error status monitoring register flag bits for each channel. To clear a status flag, write 1 to the corresponding bit of the INTERRSTATE register.

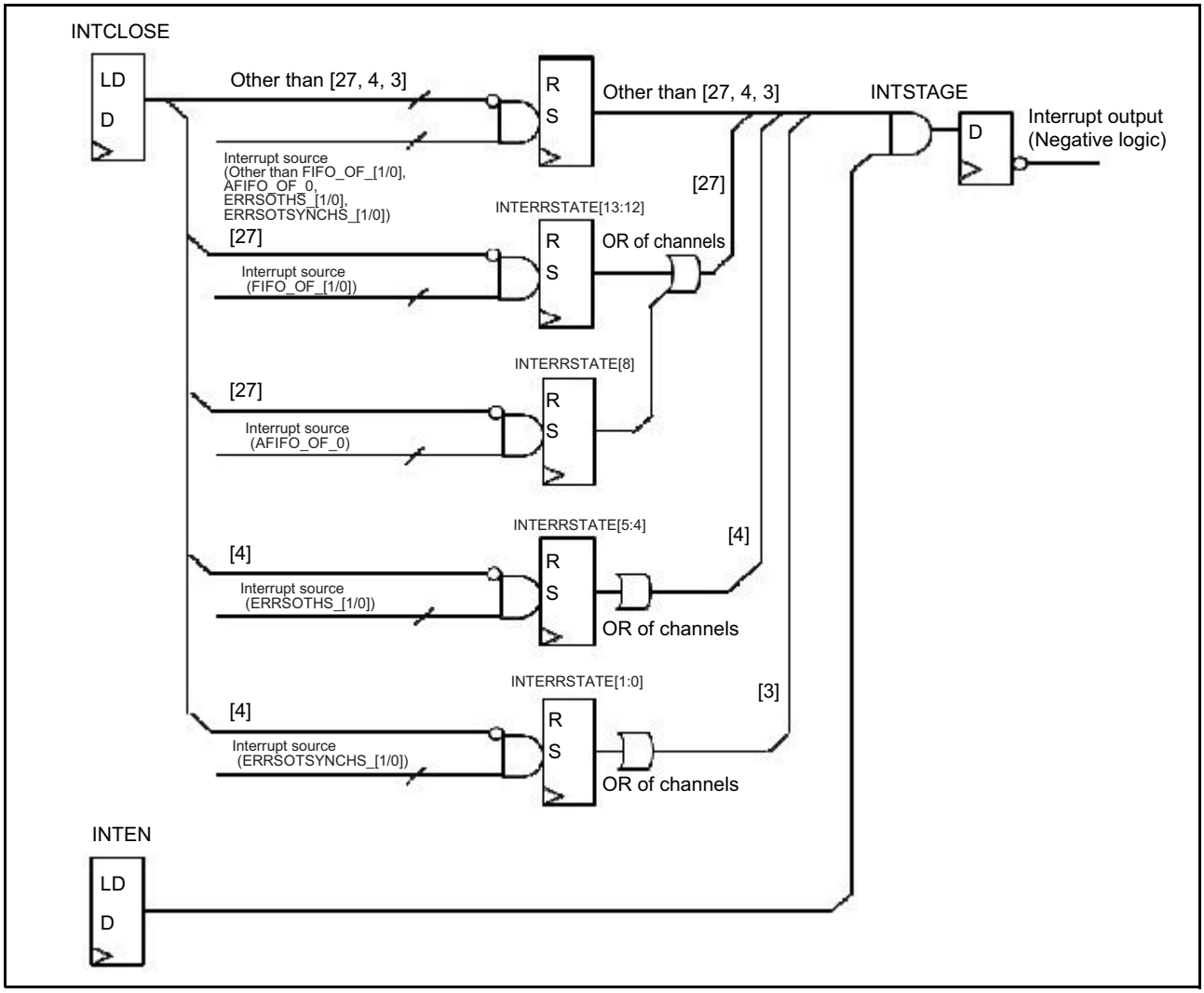


Figure 47.9 Interrupt Structure

Table 47.5 Interrupt Bit Assignments

Interrupt Source	Register Bit Name	Bit
Long packet payload data count less than WC value	LESS_THAN_WC	[28]
Overflow of asynchronous FIFO for storing HS data	AFIFO_OF	[27]
-	-	[26]
-	-	[25]
-	-	[24]
-	-	[23]
-	-	[22]
-	-	[21]
Start of frame on output channel (rising edge of CSIR_VD signal)	VD_START	[20]
End of frame on output channel (falling edge of CSIR_VD signal)	VD_END	[19]
Reception of a short packet (reception of short packets such as Frame Start and Frame End)	SHP	[18]
Reception of a frame packet (reception of short packets such as Frame Start and Frame End)	FSFE	[17]
Reception of a long packet	LNP	[16]
CRC error	CRC_ERR	[15]
Reception of WC = 0	HD_WC_ZERO	[14]
Reception of an illegal frame end packet (Frame End is received without receiving Frame Start.)	FRM_SEQ_ERR1	[13]
Reception of an illegal frame start packet (After reception of Frame Start, Frame Start is received again without receiving Frame End)	FRM_SEQ_ERR0	[12]
ECC 2-bit or more error	ECC_ERR	[11]
ECC 1-bit correction	ECC_CRCT_ERR	[10]
-	-	[9]
-	-	[8]
Start of ultra-low power state	ULPS_START	[7]
End of ultra-low power state	ULPS_END	[6]
-	-	[5]
Synchronized SOT (start of transfer) error during HS reception	ERRSOTHS	[4]
Non-synchronized SOT (start of transfer) error during HS reception	ERRSOTSYNCHS	[3]
Escape mode entry error	ERRESC	[2]
-	-	[1]
PHY control error	ERRCONTROL	[0]

Table 47.6 Bit Assignments in Error Status Monitor Register

Interrupt Source	Register Bit Name	Bit
-	-	[15]
-	-	[14]
-	-	[13]
-	-	[12]
-	-	[11]
-	-	[10]
-	-	[9]
Overflow of asynchronous FIFO for storing HS data on lane 0	AFIFO_OF_0	[8]
-	-	[7]
-	-	[6]
Synchronized SOT (start of transfer) error during HS reception on lane 1	ERRSOTHS_1	[5]
Synchronized SOT (start of transfer) error during HS reception on lane 0	ERRSOTHS_0	[4]
-	-	[3]
-	-	[2]
Non-synchronized SOT (start of transfer) error during HS reception on lane 1	ERRSOTSYNCHS_1	[1]
Non-synchronized SOT (start of transfer) error during HS reception on lane 0	ERRSOTSYNCHS_0	[0]

Table 47.7 MIPI CSI-2 PHY/ Annex A.1 Signal Description

Error Signals	Description
ERRSOTHS	Start-of-Transmission (SoT) Error. If the High-Speed Sot leader sequence is corrupted, but in such a way that proper synchronization can still be achieved, this active high signal is asserted for one cycle of RXBYTECLKHS. This is considered to be a "soft error" in the leader sequence and confidence in the payload data is reduced.
ERRSOTSYNCHS	Start-of-Transmission Synchronization Error. If the High-Speed Sot leader sequence is corrupted in a way that proper synchronization cannot be expected, this active high signal is asserted for one cycle of RXBYTECLKHS.
ERRESC	Escape Entry Error. If an unrecognized escape entry command is received, this active high signal is asserted and remains asserted until the next change in line state.
ERRCONTROL	Control Error. This active high signal is asserted when an incorrect line state sequence is detected. For example, if a turn-around request or escape mode request is immediately followed by a Stop state instead of the required Bridge state, this signal is asserted and remains asserted until the next change in line state.

47.3.7 Lane Swapping

The lane assignment for data received from the PHY block by the LINK block can be modified by setting the relevant register (see section 47.2.18, Lane Swap Register (LSWAP)). This capability provides for flexible pin assignment to compensate for the lack of a connector specification for the MIPI CSI-2.

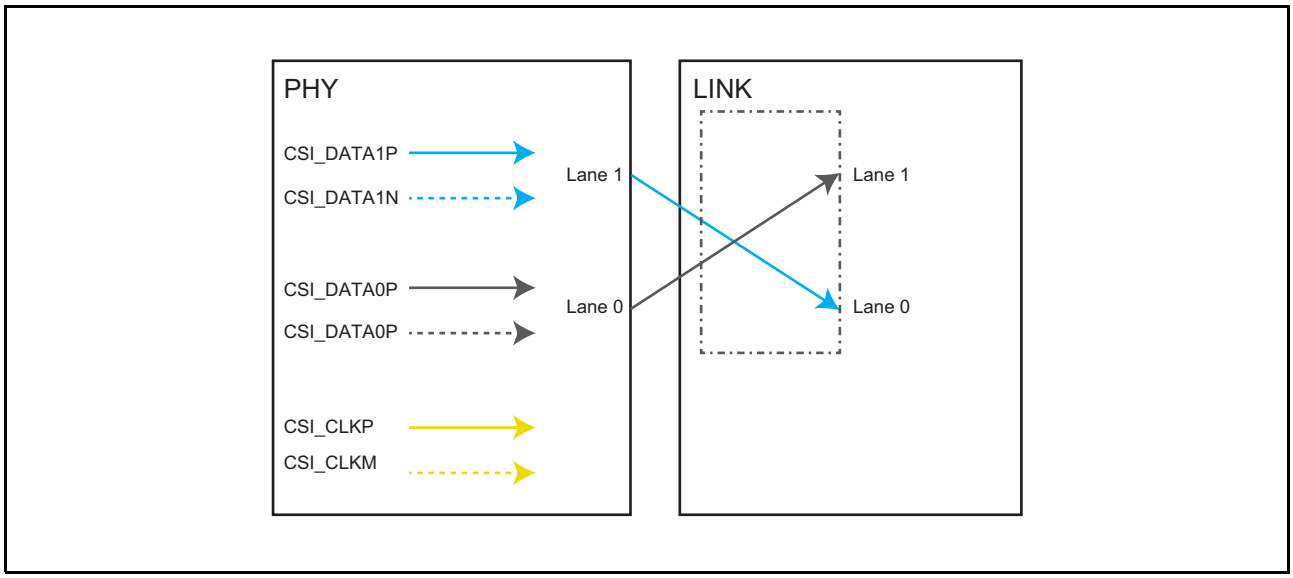


Figure 47.10 Example of Lane Swapping

47.3.8 PHY Timing Setting

The operation timing parameters defined in the MIPI D-PHY standard can be specified by setting PHY timing register 1 (PHYTIM1) in section 47.2.52, PHY timing register 2 (PHYTIM2) in section 47.2.53, and PHY timing register 3 (PHYTIM3) in section 47.2.54.

(1) T_INIT_SLAVE

These bits specify the PHY initialization period.
The PHY initialization period is the period calculated by the formula below after the SHUTDOWNZ and RSTZ bits of the PHY operation control register (PHYCNT) have been set to 1.

(T_INIT_SLAVE setting + 1 (PHY internal delay)) × (1/internal bus clock (Bφ))

When supporting the minimum value (100 µs) of T_{INIT} that is defined in the MIPI D-PHY standard, set the T_INIT_SLAVE bits to satisfy the following formula.

T_INIT_SLAVE setting + 1 (PHY internal delay) > 100 µs/(1/internal bus clock (Bφ))

(2) TCLK_MISS

These bits specify the period until a reception timeout of the HS clock is detected.
A reception timeout is detected when the period calculated by the formula below has elapsed after the receive clock has been stopped while receiving an HS clock.

TCLK_MISS setting × (1/internal bus clock (Bφ))

Note that the setting should have a margin because the difference in the clock accuracy with the device on the other end may cause a timeout to be erroneously detected.
When detecting a reception timeout of the HS clock in 80-Mbps transfer, set the TCLK_MISS bits to satisfy the following formula.

TCLK_MISS setting > (1/80 Mbps)/(1/internal bus clock (Bφ)) + 1 (margin)

(3) TCLK_PREPARE

These bits specify the period until HS IO of the clock lane is enabled.
HS IO is enabled when the period calculated by the formula below has elapsed after LP00 was input.

(TCLK_PREPARE setting × (1/internal bus clock (Bφ))) + (3 × (1/internal bus clock (Bφ))) (PHY internal delay)

Set the TCLK_PREPARE bits so that HS IO is enabled within the HS0 period.
When supporting the maximum value (95 ns) of $T_{CLK\text{-}PREPARE}$ that is defined in the MIPI D-PHY standard, set the TCLK_PREPARE bits to satisfy the following formula.

(TCLK_PREPARE setting × (1/internal bus clock (Bφ))) + (3 × (1/internal bus clock (Bφ))) (PHY internal delay) > 95 ns

(4) TCLK_SETTLE

These bits specify the period until HS clock reception of the clock lane is enabled.
HS clock reception is enabled when the period calculated by the formula below has elapsed after LP00 was input.

TCLK_SETTLE setting × (1/internal bus clock (Bφ))

Set the TCLK_SETTLE bits so that HS clock reception is enabled when at least 50 ns has elapsed after HS IO of the clock lane was enabled.
When supporting the maximum value (300 ns) of $T_{CLK\text{-}SETTLE}$ that is defined in the MIPI D-PHY standard, set the TCLK_SETTLE bits to satisfy the following formula.

300 ns > TCLK_SETTLE setting × (1/internal bus clock (Bφ)) > (TCLK_PREPARE setting × (1/internal bus clock (Bφ))) + (3 × (1/internal bus clock (Bφ))) (PHY internal delay) + 50 ns

Figure 47.11 shows the relationship between TCLK_PREPARE and TCLK_SETTLE.

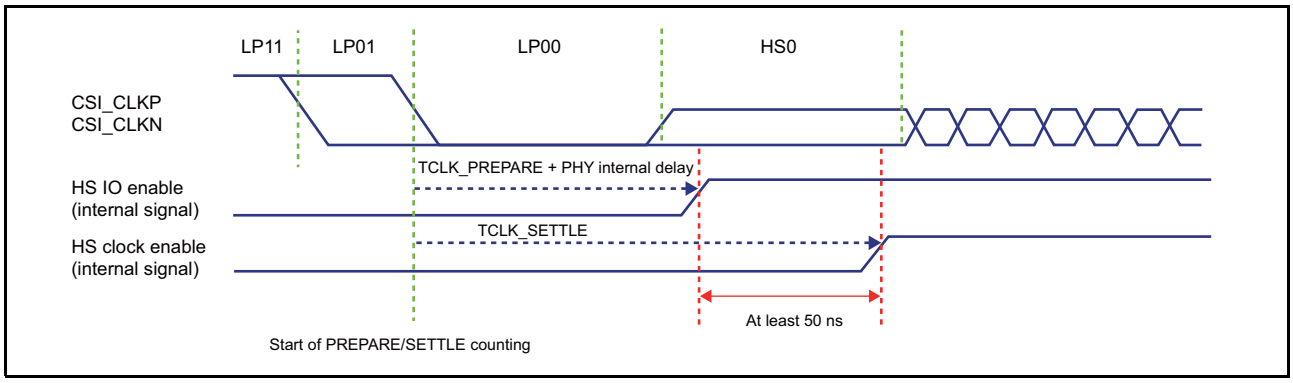


Figure 47.11 Timing of TCLK_PREPARE and TCLK_SETTLE

(5) THS_PREPARE

These bits specify the period until HS IO of the data lane is enabled.
HS IO is enabled when the period calculated by the formula below has elapsed after LP00 was input.

(THS_PREPARE setting × (1/internal bus clock (Bφ))) + (3 × (1/internal bus clock (Bφ)) + 6 UI_{INST}) (PHY internal delay)

Set the THS_PREPARE bits so that HS IO is enabled within the HS0 period.
When supporting the maximum value (85 ns + 6 UI_{INST}) of $T_{HS\text{-}PREPARE}$ that is defined in the MIPI D-PHY standard, set the THS_PREPARE bits to satisfy the following formula.

(THS_PREPARE setting × (1/internal bus clock (Bφ))) + (3 × (1/internal bus clock (Bφ)) + 6 UI_{INST}) (PHY internal delay) > 85 ns + 6 UI_{INST}

(6) THS_SETTLE

These bits specify the period until HS data reception of the data lane is enabled.
HS data reception is enabled when the period calculated by the formula below has elapsed after LP00 was input.

THS_SETTLE setting × (1/internal bus clock (Bφ))

Set the THS_SETTLE bits so that HS data reception is enabled when at least 50 ns has elapsed after HS IO of the data lane was enabled.
When supporting the maximum value (145 ns + 10 UI_{INST}) of $T_{HS-SETTLE}$ that is defined in the MIPI D-PHY standard, set the THS_SETTLE bits to satisfy the following formula.

145 ns + 10 UI_{INST} > THS_SETTLE setting × (1/internal bus clock (Bφ)) > (THS_PREPARE setting × (1/internal bus clock (Bφ))) + (3 × (1/internal bus clock (Bφ)) + 6 UI_{INST}) (PHY internal delay) + 50 ns

Figure 47.12 shows the relationship between THS_PREPARE and THS_SETTLE.

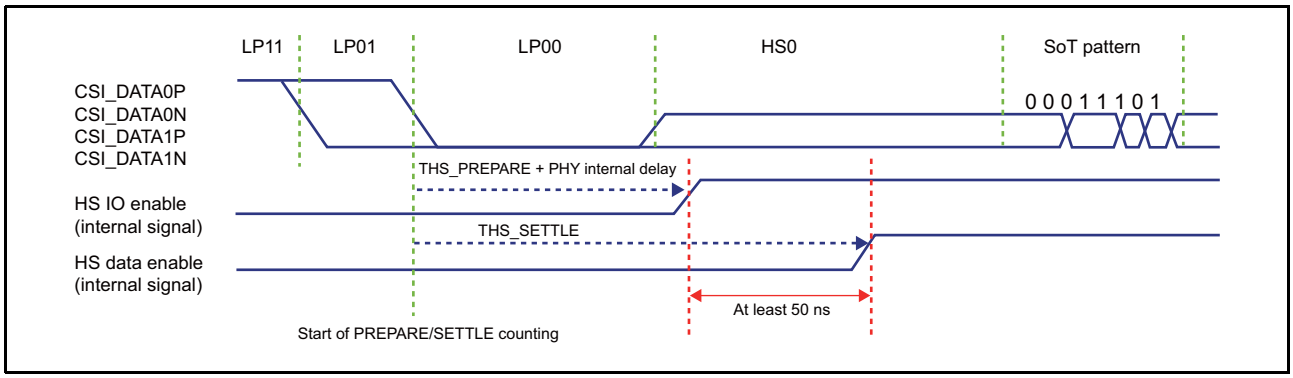


Figure 47.12 Timing of THS_PREPARE and THS_SETTLE

(7) Setting examples

Table 47.8 shows the setting examples that were given in the description of each parameter.

Table 47.8 Setting Examples of PHY Timing Registers (when Bφ = 132 MHz, 1 UI_{INST} = 1 ns)

Bit Name	Example of Calculation Formula	Setting Example
T_INIT_SLAVE	T_INIT_SLAVE + 1 > 100 µs/(1/Bφ)	16'h33F3
TCLK_MISS	TCLK_MISS > (1/80 Mbps)/(1/Bφ) + 1	5'h03
TCLK_PREPARE	(TCLK_PREPARE × (1/Bφ)) + (3 × (1/Bφ)) > 95 ns	5'h0A
TCLK_SETTLE	300 ns > TCLK_SETTLE × (1/Bφ) > (TCLK_PREPARE × (1/Bφ)) + (3 × (1/Bφ)) + 50 ns	6'h14
THS_PREPARE	(THS_PREPARE setting × (1/Bφ)) + (3 × (1/Bφ) + 6 UI_{INST}) > 85 ns + 6 UI_{INST}	6'h09
THS_SETTLE	145 ns + 10 UI_{INST} > THS_SETTLE × (1/Bφ) > (THS_PREPARE × (1/Bφ)) + (3 × (1/Bφ) + 6 UI_{INST}) (PHY internal delay) + 50 ns	6'h14

47.3.9 Initial Setting of the PHY Block

The initial sequence of the VIN module can be executed either before or after that of the CSI2 module.

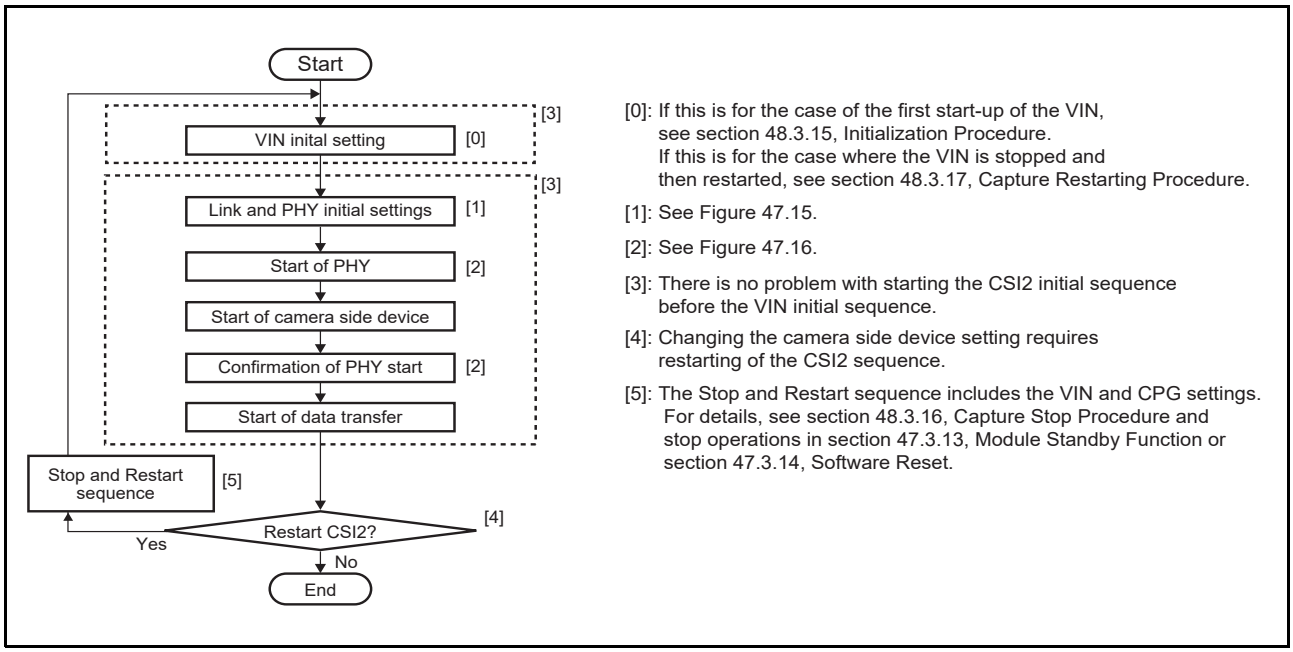


Figure 47.13 Example 1 of Initial Setting Procedure for the Video Modules

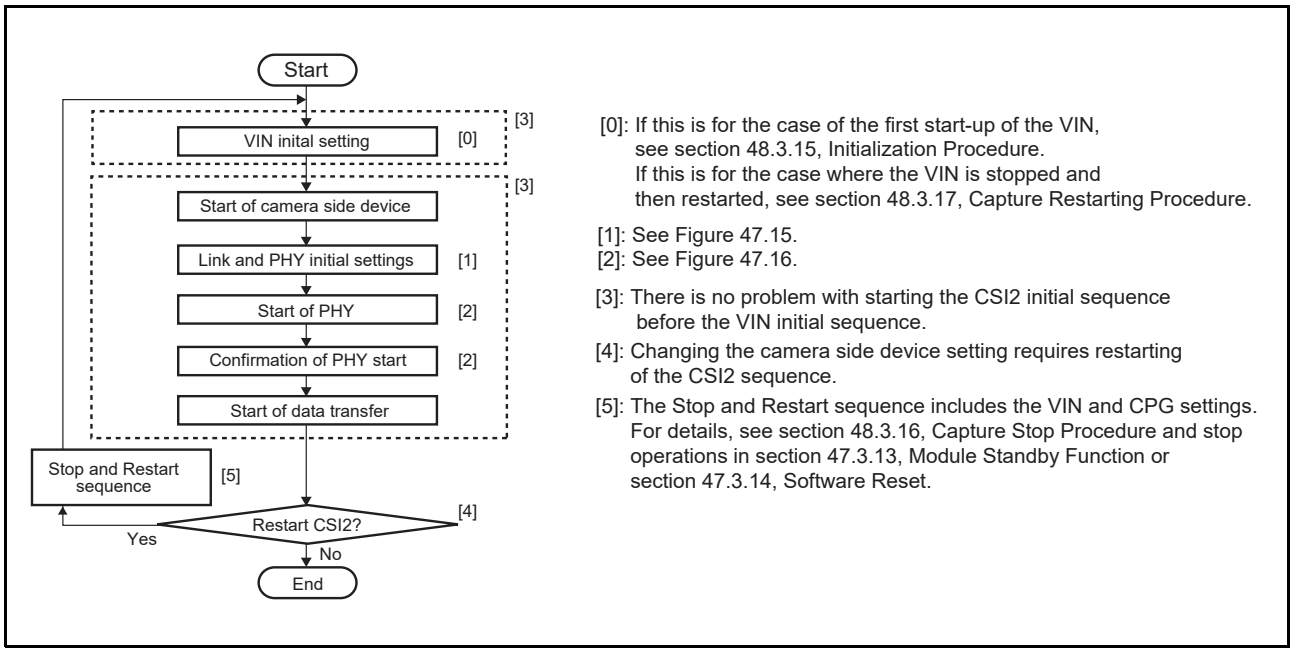


Figure 47.14 Example 2 of Initial Setting Procedure for the Video Modules

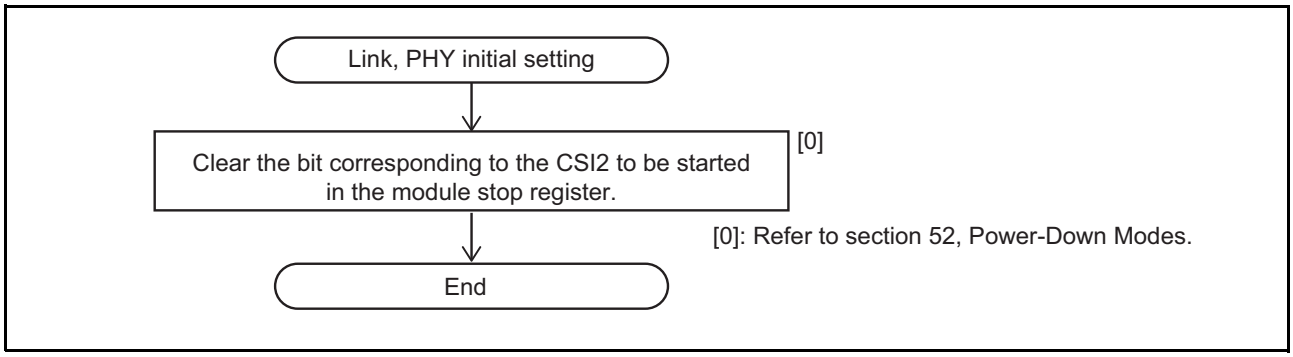


Figure 47.15 Example of PHY Initialization

47.3.9.1 Terminology

This section explains the terminology for CSI2.

- Start of PHY
  Initialization of CSI-2 receiver module. After initialization, CSI-2 receiver module expects that CSI-2 transmitter device stays StopState (LP-11).
- Start of camera side device
  Initialization of CSI-2 transmitter device. After initialization, CSI-2 transmitter device keeps StopState (LP-11).
- Confirmation of PHY start
  CSI-2 receiver module confirms that CSI-2 transmitter device stays StopState (LP-11) after initialization.
- Start data transfer
  CSI-2 transmitter device starts to send High-Speed data to CSI-2 receiver module.

47.3.10 PHY Control and Monitoring through Register Setting

The PHY IP can be controlled by setting the relevant LINK registers. For register functions, see sections 47.2.3 PHY Operation Control Register (PHYCNT), 47.2.52 PHY Timing Register 1 (PHYTIM1), 47.2.53 PHY Timing Register 2 (PHYTIM2), and 47.2.54 PHY Timing Register 3 (PHYTIM3).
The PHY IP status can be monitored through the relevant LINK registers. For register functions, see sections 47.2.19 PHY ESC Error Monitor Register (PHEERM), 47.2.20 PHY Clock Lane Monitor Register (PHCLM), and 47.2.21 PHY Data Lane Monitor Register (PHDLM).

(1) Case 1: Procedure for starting data transfer

1. Start the CSI-2 module of the camera without starting data transfer.
2. Start the CSI2 PHY module.
3. Start data transfer from the camera.

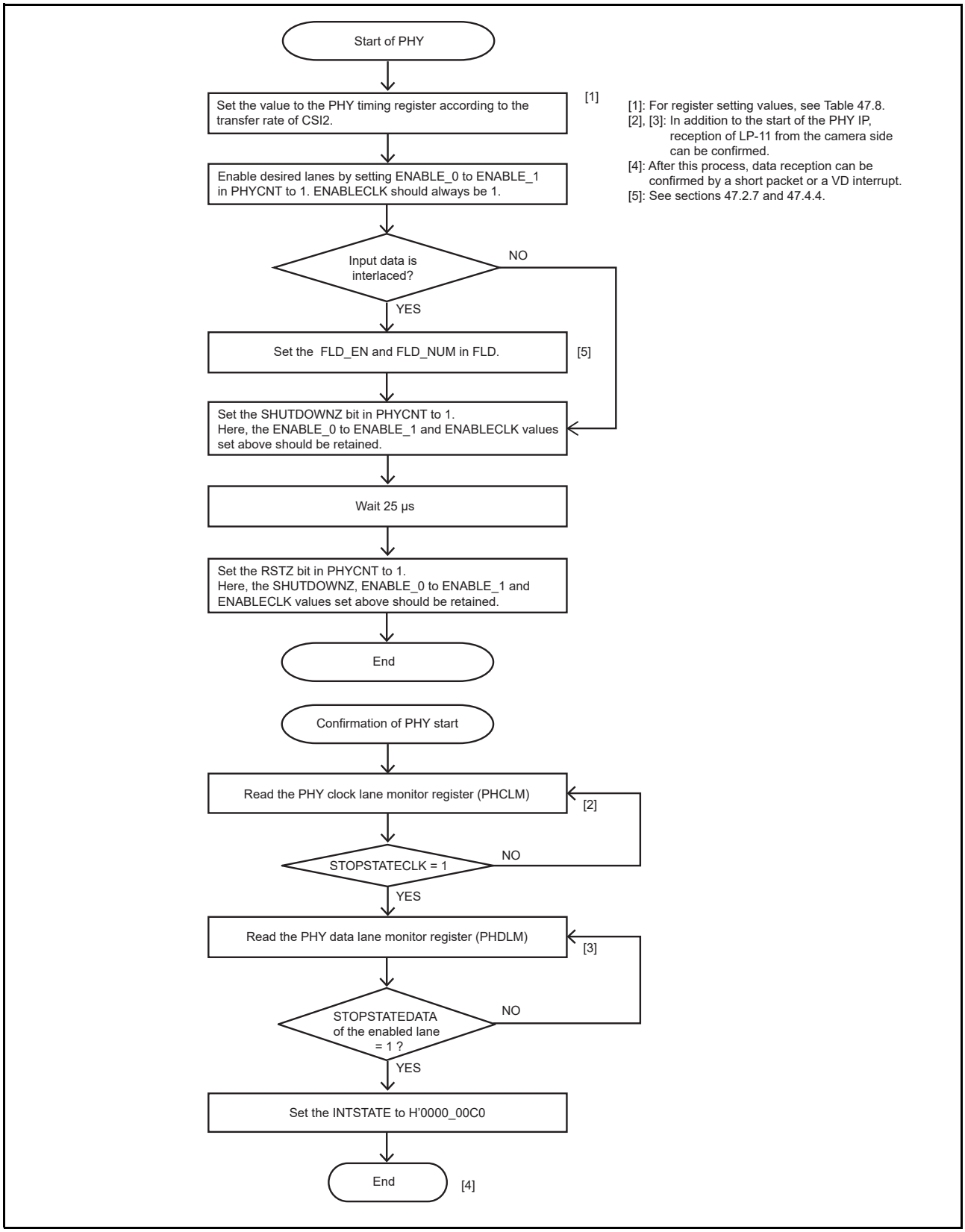


Figure 47.16 Example of PHY Starting Procedure

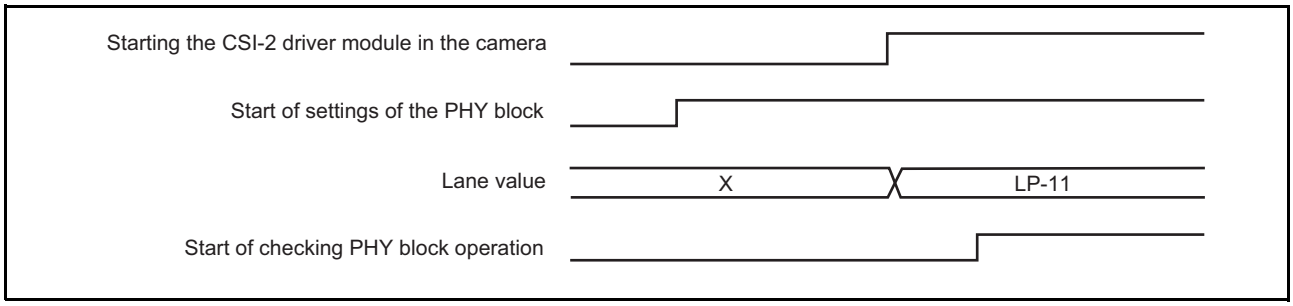


Figure 47.17 Procedure for Starting PHY Block 1

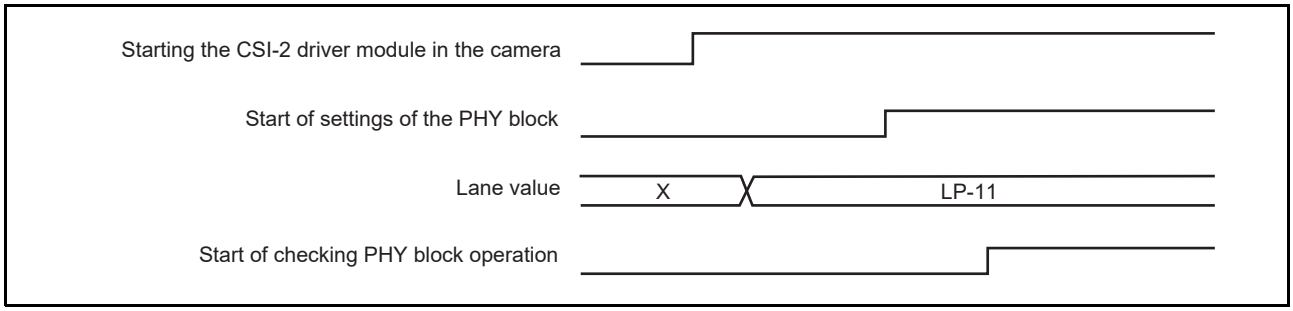


Figure 47.18 Procedure for Starting PHY Block 2

Table 47.9 Causes for Abnormal Reception

Symptom	Condition	State	Possible Causes	
PHY will not start.	Data not being transferred from the camera.	STOPSTATE = low	Internal	The CSI2 channel is different from the output channel on the camera side.
				CLK has been stopped.
				Module standby mode
				Software reset state (refer to section 47.3.14, Software Reset).
				RSTZ = 0
				SHUTDOWNZ = 0
				ENABLE = 0, ENABLECLK = 0 on reception lane
			External	No lane connected
				CSI2 on the camera side has not been started.
				The CSI2 module on the camera side has not output LP -11.
	Data being transferred from the camera.	STOPSTATE is always low.	External	The camera started up before this module and continued the output of data without stopping the signal in the clock lane (LP-11 was never received).
Reception failure	Data being transferred from the camera.	ERRSOTHS or ERRSOTSYNCHS is generated.	Internal	Different phases between the clock and data. Measures: Adjust the phase physically. Reduce the transfer rate.
		No interrupts are generated though ERRSOTHS or ERRSOTSYNCHS is not generated.		Abnormal voltage level Measures: Adjust the voltage level physically. Reduce the transfer rate.
		ECC error, CRC error, or asynchronous FIFO overflow occurs.		No ENABLE bits of the lanes during transmission are 1.
				Incorrect lane connection Measures: Change the connections. Perform lane swapping
		Clock lane is placed in ULP state.		Incorrect P/N connection of clock lane Measures: Change the connections.
		No output or abnormal output though no error occurs.	External	Incorrect transmission protocol Measures: Transfer data in the order: Frame Start → Payload data → Frame End.
			Internal	Incorrect LINK register setting (VCDT_EN = 0)

47.3.11 Interrupts Generated in Response to Changes in the ULP State and to Reception Errors

Table 47.10 MIPI CSI-2 Escape Entry Code

Escape Mode Action	Command Type	Entry Command Pattern (from the last to the first bit to have been transmitted)
Ultra-Low Power State	mode	B'0001 1110
Undefined-1	mode	B'1001 1111
Undefined-2	mode	B'1101 1110

This module can generate an interrupt signal on entry to and exit from the ultra-low-power (ULP) state.

Interrupt Source	Register Bit Name	Bit
Start of ultra-low power state	ULPS_START	[7]
End of ultra-low power state	ULPS_END	[6]

The state of the lanes can be monitored through the PHY clock lane monitoring register (PHCLM) and PHY data lane monitoring register (PHDLM).

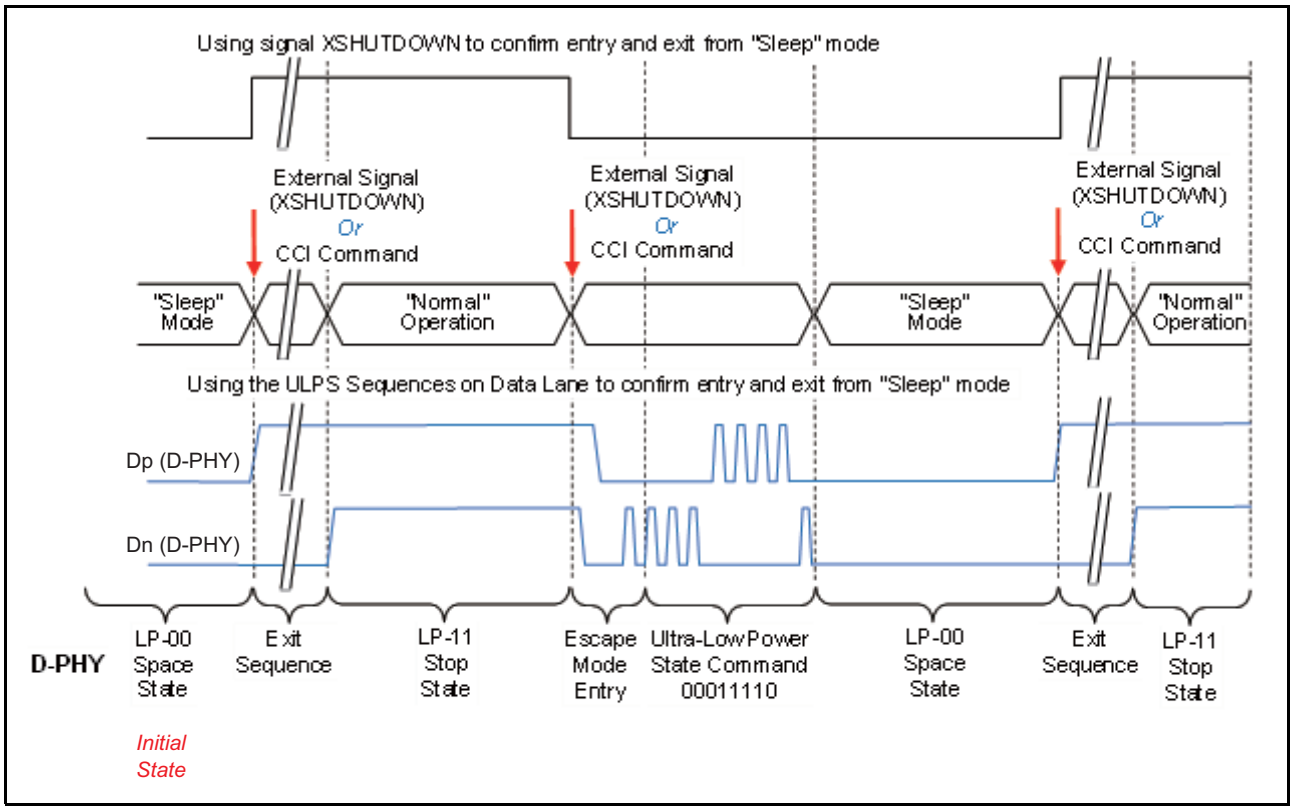


Figure 47.19 MIPI CSI-2 ULPS

This module can also generate an interrupt signal in response to reception errors

Interrupt Source	Register Bit Name	Bit
Synchronized SOT (start of transfer) error during HS reception	ERRSOTHS	[4]
Non-synchronized SOT (start of transfer) error during HS reception	ERRSOTSYNCHS	[3]
Escape mode entry error	ERRESC	[2]
PHY control error	ERRCONTROL	[0]

The error lanes related to ERRSOTHS and ERRSOTSYNCHS can be monitored through the interrupt error status monitoring register (INTERRSTATE) and the other error lanes can be monitored through the PHY ESC error monitoring register (PHEERM).

47.3.12 Monitoring Function

For monitoring of the packet header, see sections 47.2.22 to 47.2.34 and 47.2.51; for monitoring of CRC reception data, see sections 47.2.35 and 47.2.36; and for monitoring of the module internal signals used for debugging, see sections 47.2.37 to 47.2.50

47.3.13 Module Standby Function

Follow the steps below to place the CSI2 module in the module standby state. For the module standby function, refer to section 52, Power-Down Modes.

1. Set the SHUTDOWNZ bit to 0 in the PHY operation control register (PHYCNT).
2. Wait for 0.1 µs.
3. Set the SRST bit to 1 in the software reset register (SRST) of the CSI2 LINK.
4. Wait for 100 µs.
5. Set the MSTP bit assigned to this module to 1 (for details, refer to section 52, Power-Down Modes).

47.3.14 Software Reset

A software reset can be applied by controlling the SRST bit assigned to this module. The bit is stated in section 52, Power-Down Modes.
Resetting of the VIN module or stoppage of capture is required before the CSI2 module is reset.

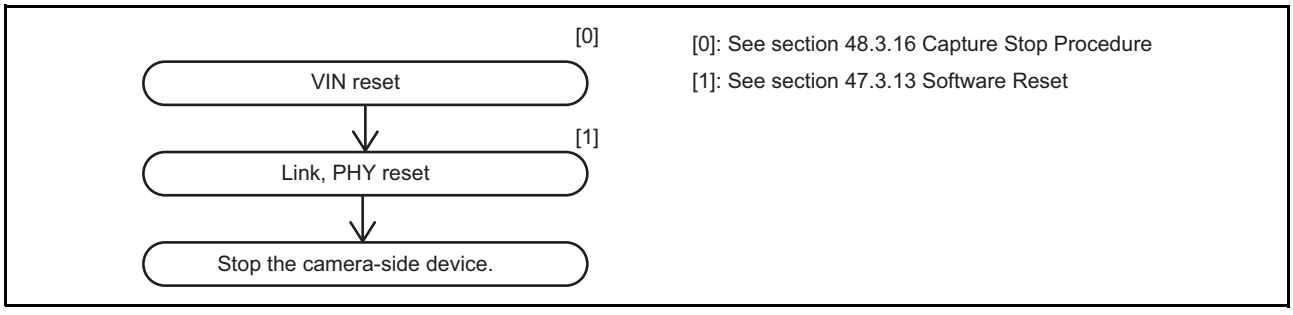


Figure 47.20 Example of Software Reset Procedure

47.4 Usage Notes

47.4.1 MIPI CSI-2 Transfer Rate

As described in section 47.3.1, Transfer Rate must not exceed 1.0 Gbps. Data being received at a rate exceeding 1.0 Gbps will result in the loss of the data for reception.

47.4.2 PHY Operation Control Register (PHYCNT)

When setting any of the ENABLE_1 to ENABLE_0 bits to 1, be sure to set the ENABLECLK bit to 1. The ENABLE_1, ENABLE_0, and ENABLECLK bits all have the initial value 0.

47.4.3 Synchronization Errors during HS (High-Speed) Reception

If HS reception fails to proceed normally, the PHY block outputs a synchronization error signal to the LINK block. When a synchronization error occurs, the PHY block may not output HS received data to the LINK block. If this occurs only in a specific lane, the lane synchronization function prevents reading of the FIFO buffer in the LINK block, thus causing an overflow. Synchronization errors and overflows can be monitored through the interrupt error status monitoring register (INTERRSTATE) by setting the interrupt source mask register (INTCLOSE) to 0.
After an overflow has occurred, the FIFO should be initialized by a software reset. For the software reset, refer to section 47.3.14, Software Reset.

47.4.4 FLD Signal

Since a field (FLD) signal is not generated in the initial state, appropriately set the field detection control register (FLD register) to select generation of an FLD signal.
This is shown in Figure 47.18, Example Interlaced Video.
In this case (the frame number of the even field is H’0002), set FLD_NUM to the same value as the frame number of the even field (H’0002) in the field detection control register (FLD) and set FLD_EN to 1 in the field detection control register (FLD register).

Note: The frame number is included in FS and FE. The frame number is incremented by 1 with every FS packet in the same virtual channel and is periodically reset to one, e.g. 1, 2, 1, 2, 1, 2.
If frame number of even field is H’0001, set FLD_NUM to H’0001.

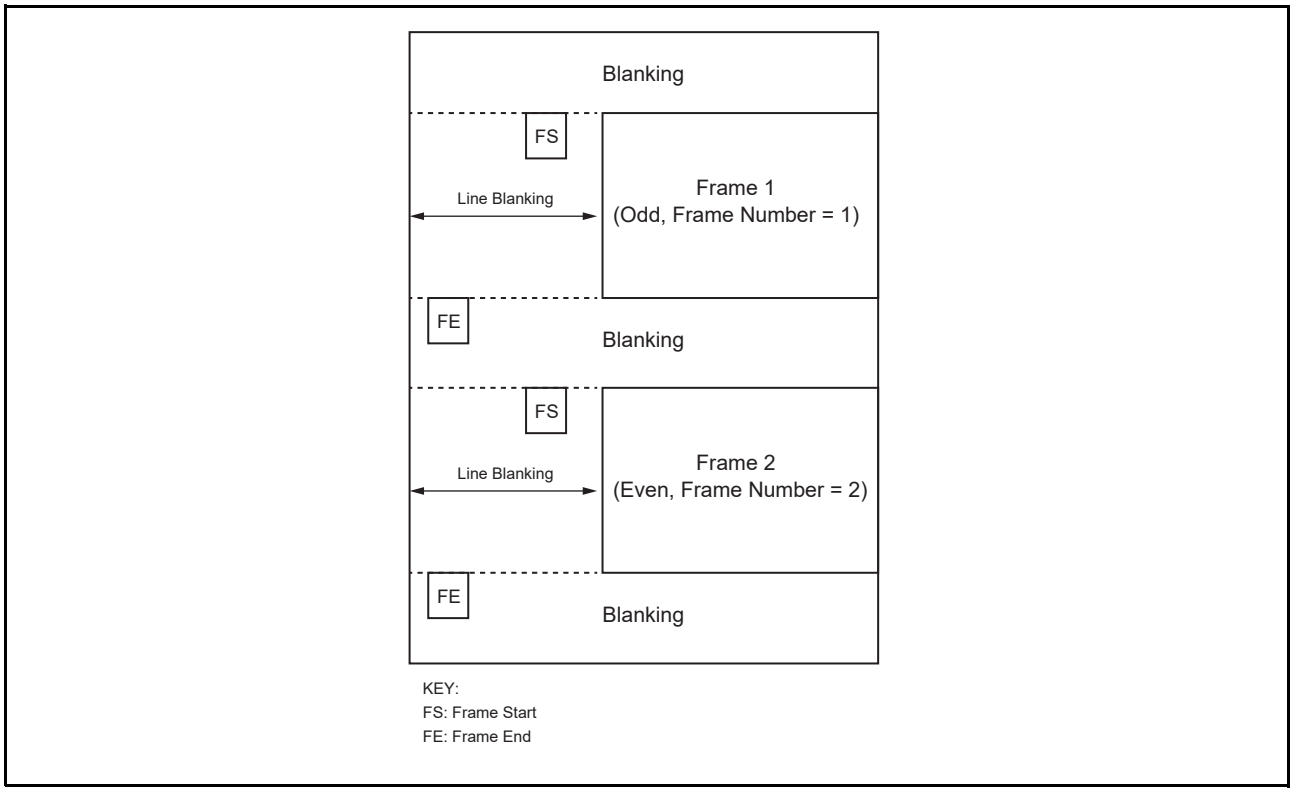


Figure 47.21 Example of Interlaced Video

47.4.5 INT_LESS_THAN_WC

When debugging INT_LESS_THAN_WC, only use a single virtual channel.

47.4.6 Transfer Rate for Interleaved Signal Transmission

When more than one virtual channel is in use, the transfer rate of the input to PHY block should be same in each of the virtual channels. When retention of the frame rate of an individual virtual channel is required, the transfer rate for input should be equivalent to the sum of the frame rates for each of the virtual channels.

47.4.7 V-Blanking Period

In order to receive data correctly, the minimum v-blanking period required is at least three lines.

Note: The V-blanking period is that from the frame end packet to the frame start packet.
The minimum and maximum values of VD_MSK_CYCLE do not define the supported minimum and maximum v-blanking periods.

47.4.8 Notes on Data

The input of data not compliant with the MIPI CSI-2 standard might cause the CSI2 module to hang or to behave in an unexpected manner.

47.4.9 Setting the Clock-Pulse Generator

Set the clock-pulse generator so that Gφ is 264 MHz when this module is to be used. For details on setting the clock-pulse generator, refer to section 6, Clock Pulse Generator.

48. Video Input Module

48.1 Overview

The video input module (hereinafter abbreviated as VIN) is a video capture module that stores in external memory YCbCr-422 data and RGB data through the MIPI CSI-2 interface.
The module has one video channel that can control the capture of data into a capture area of up to 2048 × 2048 pixels. It can also provide vertical and horizontal scaling of the data by up to three and two times, respectively.
For captured video data, the VIN provides a color space conversion function from YCbCr-422 to RGB and a format conversion function from RGB to ARGB.
As the VIN internally generates a field signal, it can capture progressive data.
Input of data through the MIPI Alliance CSI-2 interfaces is possible.

Note 1. For detailed specifications of the MIPI CSI-2 interface, see section 47, MIPI CSI2 Interface.
Note 2. The upper limits on size clipping depend on the use case.

48.1.1 Features

(1) Input interface

The following input interfaces can be selected.

Table 48.1 Input Interface for VIN

Interface (CSI2)	Data Width	Data Type
Interface (MIPI CSI-2)	—	YCbCr-422 8-bit data *1*2
Interface (MIPI CSI-2)	—	YCbCr-422 10-bit data *1*2
Interface (MIPI CSI-2)	—	RGB-888 data *1
Interface (MIPI CSI-2)	—	8-bit user defined data (RAW8) *1*2

Note 1. Inputs are in the formats which conform to the standard for the MIPI CSI-2 interface.
Note 2. Disable the XY scaling setting (only the 100% scaling is enabled).

(2) Internal Sync Signal Generation

The field signal can be internally generated.

(3) Capture Mode

The following three modes can be selected to capture the interlace images. In addition, single frame capture or continuous frame capture mode can be selected.
Triple-buffering control is provided in accordance with the captured field image and frame image to coordinate with the video capture mode of the display module.

- Odd-field capture mode
- Even-/odd-field capture mode
- Even-field capture mode

(4) Vertical and Horizontal Scaling

Factors for scaling up or down can be set to values in the range from one sixteenth to three in the vertical direction and from one sixteenth to two in the horizontal direction.

Note 1. When scaling-up is used, horizontal size that can be input is up to 2048 pixel.
Note 2. When scaling-down is used, horizontal size that can be output is up to 2048 pixel.

(5) Size Clipping

The VIN module has two clipping circuits, which independently handle images with up to 2048 × 2048 pixels. Any capture size within this limit can be specified before or after scaling.

Note 1. The upper limits on size clipping depend on the use case.

(6) Color Space Conversion

Color space conversion can be performed from YC to RGB or from RGB to YC. Desired conversion coefficients can be specified through registers to adjust colors.

(7) Lookup Table (LUT) Precision Conversion

The lookup table (LUT) precision can be converted from 10 bits to 8 bits for each pixel according to the color space conversion result.

Note: When the precision conversion function is not used, the upper 8 bits of data are output.

(8) Image Data Format Conversion

To the precision converted YCbCr422 or RGB888 image data, the following data format conversions are applicable.

- YCbCr image data
  - Y/Cb/Cr 8-bit multiplex conversion
  - YCbCr422 → YC separation (separated into Y and CbCr components.)
  - YCbCr422 → Y component extraction

- RGB image data
  - RGB-888 → 32-bit/pixel conversion
  - RGB-888 → RGB-565 (16-bit/pixel) conversion
  - RGB-888 → ARGB-1555 (16-bit/pixel) conversion
  - RGB-888 → ARGB-8888 (32-bit/pixel) conversion

Note: The lookup table is common to the YCbCr and RGB formats.

(9) Memory Output Data Format

Format-converted image data can be transferred to the memory. The following shows the available formats of the data stored.

- YCbCr image data
  - Y/Cb/Cr422, 8-bit multiplexed
  - Y/Cb/Cr422, 10-bit multiplexed
  - YC separation, YCbCr422, Y, 8-bit Cb/Cr, 8-bit multiplexed
  - YC separation, YCbCr422, Y, 10-bit Cb/Cr, 8-bit multiplexed
  - YC separation, YCbCr422, Y, 10-bit Cb/Cr, 10-bit multiplexed
  - YC separation, Y data, 8-bit
  - YC separation, Y data, 10-bit

- RGB image data
  - RGB-565 (16 bits/pixel)
  - ARGB-1555 (16 bits/pixel)
  - RGB-888 (32 bits/pixel)
  - ARGB-8888 (32 bits/pixel)

48.1.2 Block Diagram

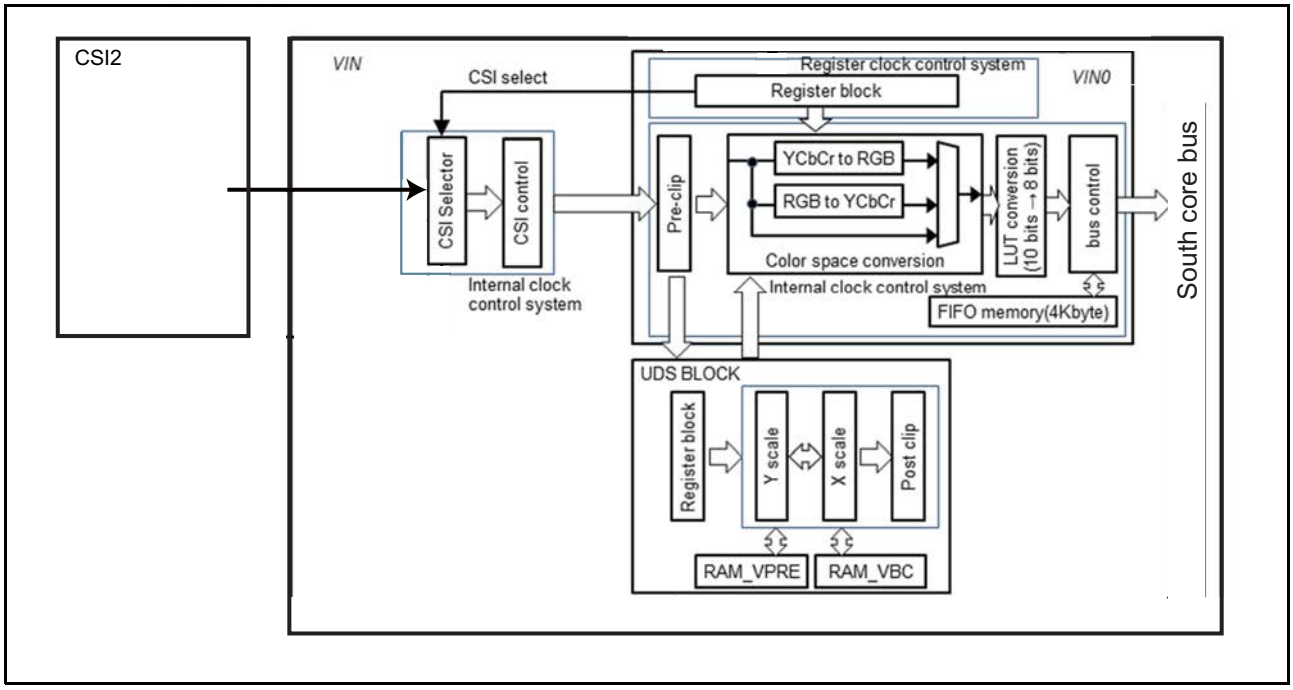


Figure 48.1 Functional Block Diagram of VIN

48.1.3 Register Configuration

Table 48.2 shows the VIN register configuration.

Note: Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined. The registers can only be accessed in 32 bits.
*: A register that supports the internal update mode is updated immediately after the register is written to.
Δ: The internal update mode is supported for the VnMC.CLP[1:0] and VnMC.LUTE bits only.

Table 48.2 VIN Registers

Name	Symbol	R/W	Address	Initial Value	Access Size	Internal Update Mode Support*
Video 0 main control register	V0MC	R/W	H'E803 F000	H'0000 0000	32	Δ
Video 0 module status register	V0MS	R	H'E803 F004	H'0000 0018	32	—
Video 0 frame capture register	V0FC	R/W	H'E803 F008	H'0000 0000	32	—
Video 0 start line pre-clip register	V0SLPrC	R/W	H'E803 F00C	H'0000 0000	32	Supported
Video 0 end line pre-clip register	V0ELPrC	R/W	H'E803 F010	H'0000 0000	32	Supported
Video 0 start pixel pre-clip register	V0SPPrC	R/W	H'E803 F014	H'0000 0000	32	Supported
Video 0 end pixel pre-clip register	V0EPPrC	R/W	H'E803 F018	H'0000 0000	32	Supported
Video 0 CSI2 interface mode register	V0CSI_IFMD	R/W	H'E803 F020	H'0000 0000	32	Supported
Video 0 image stride register	V0IS	R/W	H'E803 F02C	H'0000 0000	32	Supported
Video 0 memory base 1 register	V0MB1	R/W	H'E803 F030	H'0000 0000	32	Supported
Video 0 memory base 2 register	V0MB2	R/W	H'E803 F034	H'0000 0000	32	Supported
Video 0 memory base 3 register	V0MB3	R/W	H'E803 F038	H'0000 0000	32	Supported
Video 0 line count register	V0LC	R	H'E803 F03C	H'0000 0000	32	—
Video 0 interrupt enable register	V0IE	R/W	H'E803 F040	H'0000 0000	32	—
Video 0 interrupt status register	V0INTS	R/W	H'E803 F044	H'0000 0000	32	—
Video 0 scanline interrupt register	V0SI	R/W	H'E803 F048	H'0000 0000	32	Supported
Video 0 data mode register	V0DMR	R/W	H'E803 F058	H'0000 0000	32	Supported
Video 0 data mode register 2	V0DMR2	R/W	H'E803 F05C	H'0000 0000	32	—
Video 0 UV address offset register	V0UVAOF	R/W	H'E803 F060	H'0000 0000	32	Supported
Video 0 color space conversion coefficient 1 register	V0CSCC1	R/W	H'E803 F064	H'0129 1080	32	Supported
Video 0 color space conversion coefficient 2 register	V0CSCC2	R/W	H'E803 F068	H'0198 00D0	32	Supported
Video 0 color space conversion coefficient 3 register	V0CSCC3	R/W	H'E803 F06C	H'0064 0204	32	Supported
Video 0 scaling control register	V0UDS_CTRL	R/W	H'E803 F080	H'0000 0000	32	—
Video 0 scaling factor register	V0UDS_SCALE	R/W	H'E803 F084	H'0000 0000	32	—
Video 0 passband register	V0UDS_PASS_BWIDTH	R/W	H'E803 F090	H'0000 0000	32	—
Video 0 UDS output size clipping register	V0UDS_CLIP_SIZE	R/W	H'E803 F0A4	H'0000 0000	32	—
Video 0 lookup table pointer	V0LUTP	R/W	H'E803 F100	H'0000 0000	32	—
Video 0 lookup table data register	V0LUTD	R/W	H'E803 F104	H'00xx xxxx	32	—
Video 0 RGB → Y calculation setting register 1	V0YCCR1	R/W	H'E803 F228	H'0000 0107	32	Supported
Video 0 RGB → Y calculation setting register 2	V0YCCR2	R/W	H'E803 F22C	H'0064 0204	32	Supported
Video 0 RGB → Y calculation setting register 3	V0YCCR3	R/W	H'E803 F230	H'0A00 0010	32	Supported

Table 48.2 VIN Registers

Name	Symbol	R/W	Address	Initial Value	Access Size	Internal Update Mode Support*
Video 0 RGB → Cb calculation setting register 1	V0CBCCR1	R/W	H'E803 F234	H'0000 1F68	32	Supported
Video 0 RGB → Cb calculation setting register 2	V0CBCCR2	R/W	H'E803 F238	H'01C2 1ED6	32	Supported
Video 0 RGB → Cb calculation setting register 3	V0CBCCR3	R/W	H'E803 F23C	H'0A00 0080	32	Supported
Video 0 RGB → Cr calculation setting register 1	V0CRCCR1	R/W	H'E803 F240	H'0000 01C2	32	Supported
Video 0 RGB → Cr calculation setting register 2	V0CRCCR2	R/W	H'E803 F244	H'1FB7 1E87	32	Supported
Video 0 RGB → Cr calculation setting register 3	V0CRCCR3	R/W	H'E803 F248	H'0A00 0080	32	Supported
Video 0 YC → RGB calculation setting register 1	V0CSCE1	R/W	H'E803 F300	H'0000 129F	32	Supported
Video 0 YC → RGB calculation setting register 2	V0CSCE2	R/W	H'E803 F304	H'0100 0800	32	Supported
Video 0 YC → RGB calculation setting register 3	V0CSCE3	R/W	H'E803 F308	H'1989 0D02	32	Supported
Video 0 YC → RGB calculation setting register 4	V0CSCE4	R/W	H'E803 F30C	H'0645 2045	32	Supported

48.2 Register Description

[Legend]

—: Reserved. The write value should always be 0.
Initial value: Register value after a reset
R/W: Readable/writable. The written value can be read.
R: Read-only. The write value should always be 0.
Vn: Video channel n

48.2.1 Video n Main Control Register (VnMC)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CLP[1:0]		—	SCLE	—	—	—	—	—	LUTE	YCAL	INF[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DC[1:0]		—	—	—	—	—	—	—	EN	—	IM[1:0]		—	BPS	ME
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29, 28	CLP[1:0]	00	R/W	Pixel Data Clipping When the input image data is in the YCbCr format, these bits specify the data clip value for clipping the YCbCr-RGB color conversion input data to the nominal range prescribed in the ITU-R BT.601 standard.
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26	SCLE	0	R/W	This bit is used to enable or disable scaling by the UDS. 0: Disables scaling by the UDS. 1: Enables scaling by the UDS.
25 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

CLP	Luminance	Color Difference	
00	No clipping	16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240.	Initial condition
01	16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240.	16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240.	
10	No clipping	16 or a smaller value is clipped to 128. 240 or a greater value is clipped to 128.	
11	No clipping	No clipping	

Note: These bits support the internal update mode.

Bit	Bit Name	Initial Value	R/W	Description
20	LUTE	0	R/W	Lookup Table Enable This bit enables conversion from 10 bits to 8 bits through the lookup table (LUT). 0: LUT is not used. 1: LUT is used. Note 1. To perform LUT conversion, the conversion table should be set with VnLUTP and VnLUTD. Note 2. This bit supports the internal update mode.
19	YCAL	0	R/W	YCbCr-422 Input Data Alignment This bit controls data alignment for YCbCr-422 input. 0: When the multiplexed CbCr interface is set, capturing is performed with Y in the upper bits and CbCr in the lower bits. 1: When the multiplexed CbCr interface is set, capturing is performed with CbCr in the upper bits and Y in the lower bits.
18 to 16	INF[2:0]	000	R/W	Input Interface Format These bits specify the format of images input to the VIN. 000: Setting prohibited 001: 8-bit YCbCr-422*1 010: Setting prohibited 011: 10-bit YCbCr-422*1 100: 8-bit user defined data (RAW8) *1 101: Setting prohibited 110: 24-bit RGB-888*1 111: Setting prohibited
15, 14	DC[1:0]	00	R/W	Dithering Mode Control These bits select the dithering mode for conversion from RGB888 to RGB565/ ARGB1555. 00: Dithering with cumulative addition 01: Ordered dithering 10: Setting prohibited 11: Setting prohibited
13 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	EN	0	R/W	Endian Type This bit specifies the endian type for data to be output to external memory. 0: Image data is packed and allocated in little endian. 1: Image data is packed and allocated in big endian. Note: When allocating the YCbCr422 (UYVY format) data in big endian, be sure to set the BPSM bit in VnDMR to 1.
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4, 3	IM[1:0]	00	R/W	Interlace Mode These bits specify the capture mode. Do not modify this setting during capture operation. 00: Odd-field (field 1) capture mode Handles only odd fields as frames and stores them in external memory. 01: Odd-/even-field capture mode Handles odd and even fields as separate frames and stores them in external memory. This mode is available only in continuous frame capture mode. 10: Even-field (field 2) capture mode Handles only even fields as frames and stores them in external memory. 11: Setting prohibited
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1	BPS	0	R/W	Color Space Conversion Bypass Mode 0: The input YCbCr data is converted into the RGB color space and RGB data is converted into the YCbCr color space*2. 1: Color space conversion is not performed. Note: YCbCr → RGB or RGB → YCbCr conversion is performed with the coefficients specified by the YC-RGB conversion coefficient register or RGB-YC coefficient register.

Bit	Bit Name	Initial Value	R/W	Description
0	ME	0	R/W	Module Enable This is the enable bit for the VIN. Set this bit before setting the video n Frame Capture register (VnFC). 0: The module operation is stopped. 1: The module operation is enabled. *3

Note 1. Table 48.3 shows the combinations of interfaces that can be set by the input interface format (the INF bits in VnMC). Do not make any setting that is not listed in the table.
Note 2. Table 48.4 shows the image data that can be converted according to the color space conversion bypass mode settings (set by the BPS bit in VnMC).
Note 3. To stop capturing operation, only set the ME bit to 0. Do not change the other bit settings.

Table 48.3 Capture Interface Settings

Interface	VnMC/INF
YCbCr-422 8-bit data (MIPI CSI-2)	001
YCbCr-422 10-bit data (MIPI CSI-2)	011
RGB-888 data (MIPI CSI-2)	110
8-bit user defined data (RAW8) (MIPI CSI-2)	100

Table 48.4 Captured Data Formats

Input Data Format	VnMC/INF	VnMC/BPS	VnMC/IM	Captured Data Format
YCbCr-422 (MIPI CSI-2)	001/011	0	—	RGB format
		1	—	YCbCr format
RGB-888 data (MIPI CSI-2)	110	0	—	YCbCr format
		1	—	RGB format
8-bit user defined data (RAW8)	100	1	01	User defined format

48.2.2 Video n Module Status Register (VnMS)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	FBS[1:0]		FS	AV	CA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4, 3	FBS[1:0]	11	R	Frame Buffer Status These bits show the frame buffer status. 00: The latest valid frame buffer has the base address defined by the memory base 1 register. 01: The latest valid frame buffer has the base address defined by the memory base 2 register. 10: The latest valid frame buffer has the base address defined by the memory base 3 register. 11: There is no valid frame buffer. Note: When video capture is operating, these bits should be read after the FIS bit in VnINTS is set to 1.
2	FS	0	R	Field Status This bit shows the type of the current capture field. 0: The current field is an odd field (field 1). 1: The current field is an even field (field 2). Note: When video capture is operating, this bit should be read after the FIS bit in VnINTS is set to 1.
1	AV	0	R	Active Video Status This bit shows whether the current field is in the active video area defined by the pre-clipping register. 0: The current field is not in the active video area. 1: The current field is in the active video area. Note: This bit will be 0 if no input data is captured.
0	CA	0	R	Video Capture Active Status This bit shows the current video capture operation status. This bit is updated by the captured field signal. 0: Video capture is not operating. 1: Video capture is operating. Note: In field capture mode, this bit is set to 1 even for the field that does not capture data.

48.2.3 Video n Frame Capture Register (VnFC)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CC	SC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	CC	0	R/W	Continuous Frame Capture Mode This bit specifies the continuous frame capture mode. In this mode, the first capture frame is written into the memory address specified by the video n memory base 1 register (VnMB1). After that, the capture operation is repeated in the order of MB2, MB3, MB1, and MB2. Writing 0 into this bit during continuous capture operation will immediately terminate the capture operation if the current frame is completed or it has not been captured. 0: The continuous frame capture mode is not set. 1: The continuous frame capture mode is set. After the capture operation is resumed by writing 1 into this bit, the memory address where the first capture frame is written depends on the value indicated by the frame buffer status (FBS[1:0]) bits in the video n module status register (VnMS). Note: When the capture is started with the SC bit set to 1 before setting this bit to 1, the first capture frame is written into the memory address specified by the video n memory base 2 register (VnMB2).
0	SC	0	R/W	Single Frame Capture Mode This bit specifies the single frame capture mode. In this mode, the capture frame is written into the memory address specified by the video n memory base 1 register (VnMB1). Immediately after this bit is set to 1, the frame buffer status bits (FBS) in VnMS are initialized and the SC bit is also cleared to 0. 0: The single frame capture mode is not set. 1: The single frame capture mode is set. Note: Do not set this bit to 1 when the interlace mode bits (IM) in the video n main control register (VnMC) are set to 01 (odd-/even-field capture mode).

Note: Do not specify the single frame capture mode and continuous frame capture mode at the same time.

48.2.4 Video n Start Line Pre-Clip Register (VnSLPrC)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLPrC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SLPrC[10:0]	H'000	R/W	Start Line Pre-Clip These bits specify the (pre-clipping start line - 1) value in line units. This value is used before scaling. Specify a value in the range from 0 to 2046 so that the number of lines after pre-clipping will be 2 or more. (The value of 0 indicates the first valid line.) Note 1. When using vertical or horizontal scaling, specify a value in the range from 0 to 2044 so that the number of lines after pre-clipping will be 4 or more. Note 2. In case of using Scaler, set the value of SLPrC within input l ines.

48.2.5 Video n End Line Pre-Clip Register (VnELPrC)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ELPrC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ELPrC[10:0]	H'000	R/W	End Line Pre-Clip These bits specify the (pre-clipping end line - 1) value in line units. This value is used before scaling. Specify a value in the range from 1 to 2047 so that the number of lines after pre-clipping will be 2 or more. Note 1. When using vertical or horizontal scaling, specify a value in the range from 3 to 2047 so that the number of lines after pre-clipping will be 4 or more. Note 2. In case of using Scaler, set the value of ELPrC within input lines. Note 3. If the interlace images are input, the value here should be such that ELPrC - SLPrC is half of the value of the number of the original line.

48.2.6 Video n Start Pixel Pre-Clip Register (VnSPPrC)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SPPrC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SPPrC[10:0]	H'000	R/W	Start Pixel Pre-Clip These bits specify the (pre-clipping start pixel - 1) value in pixel units. This value is used before scaling. Specify a value in the range from 0 to 2040 so that the number of pixels after pre-clipping will be an even number than 6. Note 1. When SPPrC is set to 0x0, the first valid pixel is specified. Note 2. Specify an even number. Note 3. In case of using Scaler, set the value of SPPrC within input pixels.

48.2.7 Video n End Pixel Pre-Clip Register (VnEPPrC)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	EPPrC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	EPPrC[10:0]	H'000	R/W	End Pixel Pre-Clip These bits specify the (pre-clipping end pixel - 1) value in pixel units. This value is used before scaling. Specify a value in the range from 7 to 2047 so that the number of pixels after pre-clipping will be an even number than 6. Note 1. Set these bits so that the (EPPrC - SPPrC) value is an odd number. Note 2. In case of using Scaler, set the value of EPPrC within input pixels.

48.2.8 Video n CSI2 Interface Mode Register (VnCSI_IFMD)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	DES0	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	DES0	0	R/W	Data Extension Select This bit is used to select how data are expanded to 12 bits within the VIN module. 0: Empty bits in the input data are repeatedly expanded from the highest-order bit. 1: Empty bits will be padded with zeros. This bit must be set to 1 when the YCbCr-422 interface is in use.
24 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

48.2.9 Video n Image Stride Register (VnIS)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IS[8:0]									—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

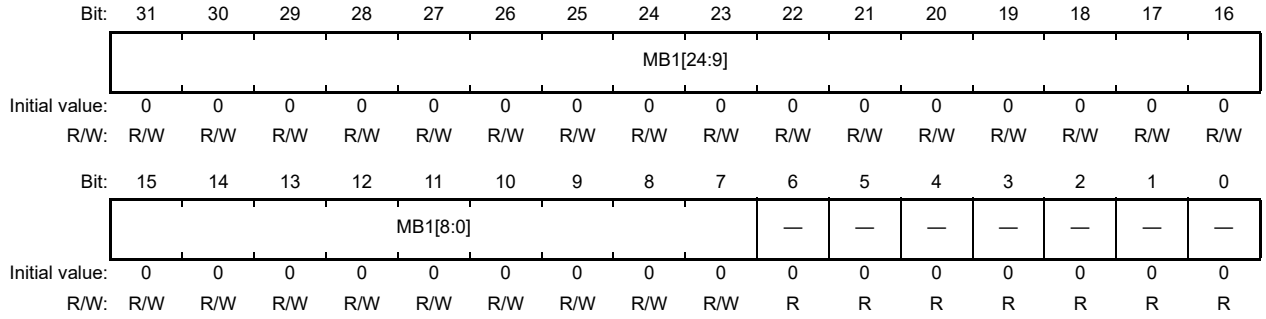
Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 4	IS[8:0]	H'000	R/W	Image Stride These bits specify the image stride. Set the value obtained by dividing the number of pixels by 16 as shown in the table below. When the UDS is in use: Specify a value no less than the post-clipping width (CL_HSIZE) . When the UDS is not in use: Specify a value no less than the pre-clipping width (EPPrC - SPPrC). Note: Set the parameters in accord with the output format as listed in the table below.
3 to 0	—	All 0	R	Reserved bits that indicate the lower-order four bits of the image stride. These bits are always read as 0. The write value should always be 0.

Output Format	Unit for Setting (Number of Pixels)	Unit for Setting (IS[8:0])
YCbCr-422 8-bit data	64	4
YCbCr-422 10-bit data	32	2
YC separation YCbCr-422, 8-bit Y and 8-bit C data	128	8
YC separation YCbCr-422, 10-bit Y and 10-bit C data	64	4
YC separation YCbCr-422, 10-bit Y and 8-bit C data	128	8
RGB565	64	4
ARGB1555	64	4
RGB888	32	2
ARGB8888	32	2
RAW8 *1	64	4

Note 1. When the output format in use is RAW8, images are output as 1 byte per pixel. In this case, the image stride is obtained by multiplying the number of pixels specified in these bits by 2 bytes.

48.2.10 Video n Memory Base 1 Register (VnMB1)

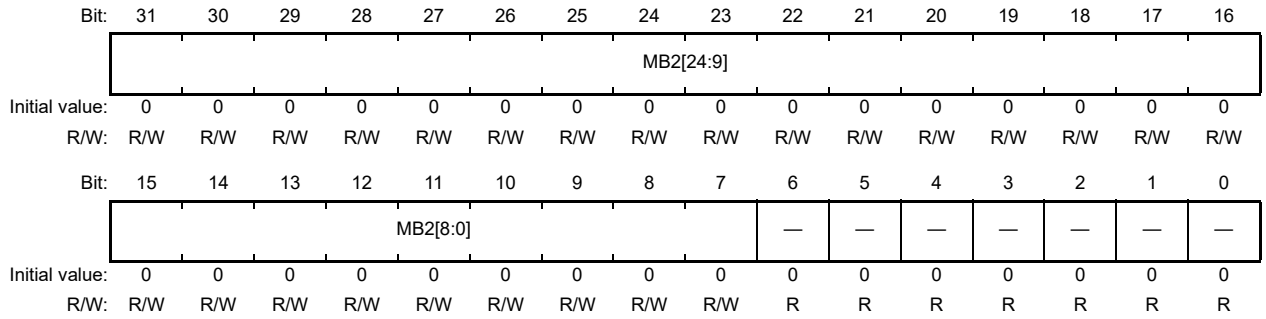
Note: Availability of channels: n = 0



Bit	Bit Name	Initial Value	R/W	Description
31 to 7	MB1[24:0]	H'000 0000	R/W	Memory Base Address 1 These bits specify the transfer start address in frame buffer 1. Specify a value for physical address bits [31:7] in units of 128 bytes. If the module is in continuous frame capture mode, this value is used as the MB1 address in the following capture sequence: MB1 → MB2 → MB3 → MB1 → MB2 → MB3. In single frame capture mode, this value is used as the capture address. Specify a memory address taking into account the image size so that the image data does not exceed an area boundary on the address map.
6 to 0	—	All 0	R	Reserved bits that indicate the lower-order seven bits of memory base address 1 (a multiple of 128 bytes). These bits are always read as 0. The write value should always be 0.

48.2.11 Video n Memory Base 2 Register (VnMB2)

Note: Availability of channels: n = 0



Bit	Bit Name	Initial Value	R/W	Description
31 to 7	MB2[24:0]	H'000 0000	R/W	Memory Base Address 2 These bits specify the transfer start address in frame buffer 2. Specify a value for physical address bits [31:7] in units of 128 bytes. If the module is in continuous frame capture mode, this value is used as the MB2 address in the following capture sequence: MB1 → MB2 → MB3 → MB1 → MB2 → MB3. Specify a memory address taking into account the image size so that the image data does not exceed an area boundary on the address map.
6 to 0	—	All 0	R	Reserved bits that indicate the lower-order seven bits of memory base address 2 (a multiple of 128 bytes). These bits are always read as 0. The write value should always be 0.

48.2.12 Video n Memory Base 3 Register (VnMB3)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB3[24:9]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB3[8:0]									—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	MB3[24:0]	H'000 0000	R/W	Memory Base Address 3 These bits specify the transfer start address in frame buffer 3. Specify a value for physical address bits [31:7] in units of 128 bytes. If the module is in continuous frame capture mode, this value is used as the MB3 address in the following capture sequence: MB1 → MB2 → MB3 → MB1 → MB2 → MB3. Specify a memory address taking into account the image size so that the image data does not exceed an area boundary on the address map.
6 to 0	—	All 0	R	Reserved bits that indicate the lower-order seven bits of memory base address 3 (a multiple of 128 bytes). These bits are always read as 0. The write value should always be 0.

48.2.13 Video n Line Count Register (VnLC)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	LC[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	LC[11:0]	H'000	R	Line Count These bits show the line position in the current capture field.

48.2.14 Video n Interrupt Enable Register (VnIE)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIE2	—	—	—	—	—	—	—	—	—	—	—	—	—	VFE	VRE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	FIE	—	SIE	EFE	FOE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	FIE2	0	R/W	Field Interrupt Enable 2 This bit enables or disables INTC output for field interrupts. Interrupt signals by this enable bit is asserted irrespective of whether capture is taking place. 0: Field interrupts are disabled. 1: Field interrupts are enabled.
30 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	VFE	0	R/W	VSYNC Falling Edge Detect Interrupt Enable This bit enables or disables VSYNC falling edge detect interrupts. Interrupt signals by this enable bit is asserted irrespective of whether capture is taking place. 0: VSYNC falling edge detect interrupts are disabled. 1: VSYNC falling edge detect interrupts are enabled.
16	VRE	0	R/W	VSYNC Rising Edge Detect Interrupt Enable This bit enables or disables VSYNC rising edge detect interrupts. Interrupt signals by this enable bit is asserted irrespective of whether capture is taking place. 0: VSYNC rising edge detect interrupts are disabled. 1: VSYNC rising edge detect interrupts are enabled.
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	FIE	0	R/W	Field Interrupt Enable This bit enables or disables field-switching interrupts. This interrupt enable setting is valid when the CA bit in VnMS is 1. 0: Field-switching interrupts are disabled. 1: Field-switching interrupts are enabled.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	SIE	0	R/W	Scanline Interrupt Enable This bit enables or disables scanline interrupts. This interrupt enable setting is valid when the CA bit in VnMS is 1. 0: Scanline interrupts are disabled. 1: Scanline interrupts are enabled.
1	EFE	0	R/W	End of Frame Interrupt Enable This bit enables or disables end of frame interrupts. This interrupt enable setting is valid when the CA bit in VnMS is 1. 0: End of frame interrupts are disabled. 1: End of frame interrupts are enabled.
0	FOE	0	R/W	FIFO Overflow Interrupt Enable This bit enables or disables FIFO overflow interrupts. This interrupt enable setting is valid when the CA bit in VnMS is 1. 0: FIFO overflow interrupts are disabled. 1: FIFO overflow interrupts are enabled.

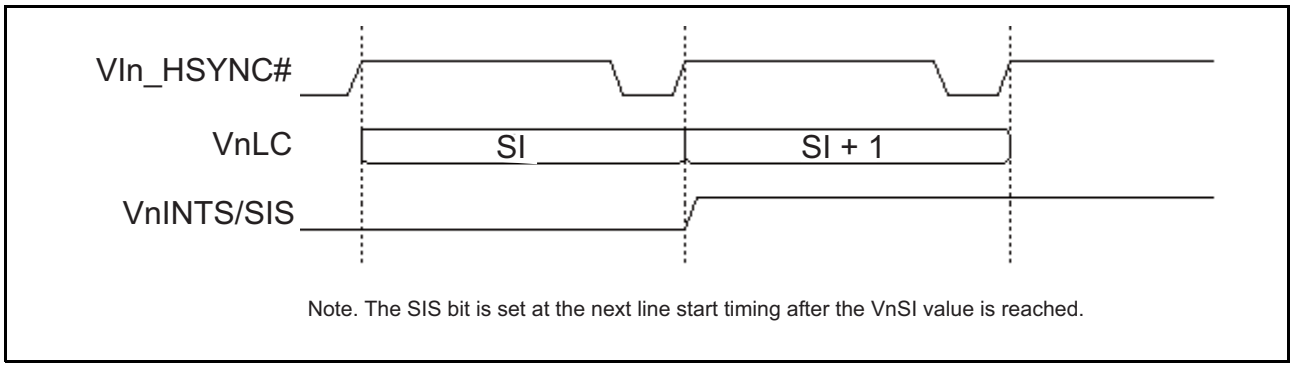
48.2.15 Video n Interrupt Status Register (VnINTS)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIS2	—	—	—	—	—	—	—	—	—	—	—	—	—	VFS	VRS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	FIS	—	SIS	EFS	FOS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	FIS2	0	R/W	Field Interrupt Status 2 This bit shows that the field has changed. After being set to 1, this bit is cleared to 0 by writing 1. Note: This bit is set to 1 irrespective of whether or not capture is taking place. Be sure to clear this bit to 0 before using it.
30 to 18	—	All 0	R	Reserved These bits are always read as 0 or 1. The write value should always be 0.
17	VFS	0	R/W	VSYNC Falling Edge Detect Interrupt Status This bit shows that a VSYNC falling edge has been detected. After being set to 1, this bit is cleared to 0 by writing 1. Note: This bit is set to 1 irrespective of whether or not capture is taking place. Be sure to clear this bit to 0 before using it.
16	VRS	0	R/W	VSYNC Rising Edge Detect Interrupt Status This bit shows that a VSYNC rising edge has been detected. After being set to 1, this bit is cleared to 0 by writing 1. Note: This bit is set to 1 irrespective of whether or not capture is taking place. Be sure to clear this bit to 0 before using it.
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	FIS	0	R/W	Field Interrupt Status This bit shows that a field has been captured in the active capture operation. After being set to 1, this bit is cleared to 0 by writing 1.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	SIS	0	R/W	Scanline Interrupt Status This bit shows that the number of lines specified by VnSI has been reached in the active capture operation. After being set to 1, this bit is cleared to 0 by writing 1. This bit is set to 1 at the next line start timing after the value in the VnLC register matches the VnSI register setting. This timing is shown in Figure 48.2, Scanline Interrupt Status Generation Timing.
1	EFS	0	R/W	End of Frame Interrupt Status This bit shows that the last frame has been reached in the active capture operation. This bit is set to 1 at the end of even field (field 2). After being set to 1, this bit is cleared to 0 by writing 1.
0	FOS	0	R/W	FIFO Overflow Interrupt Status This bit shows that the FIFO has overflowed in the active capture operation. After being set to 1, this bit is cleared to 0 by writing 1. If the FIFO overflows, the FIFO data is overwritten by the pixel data captured after the overflow, and the resultant data is sent to the frame buffer.



Note. The SIS bit is set at the next line start timing after the VnSI value is reached.

Figure 48.2 Scanline Interrupt Status Generation Timing

48.2.16 Video n Scanline Interrupt Register (VnSI)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SI[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SI[10:0]	H'000	R	Scanline Interrupt Setting These bits specify a value to be compared with the VnLC register value in each field while the SIE bit in the VnIE register is set to 1. When this value matches the VnLC register value, an interrupt signal is asserted. Note: When these bits are set to H'000, the value of the scanline interrupt status bit (SIS) in the video n interrupt status register (VnINTS) is always 0.

48.2.17 Video n Data Mode Register (VnDMR)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	A8BIT[7:0]								—	—	—	—	—	—	—	EVA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	YMODE[2:0]			YC_THR	—	—	EXRGB	—	—	—	BPSM	—	ABIT	DTMD[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	A8BIT[7:0]	H'00	R/W	Alpha 8 These bits set the alpha value for the ARGB8888 format output.
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	EVA	0	R/W	Even Field Address Offset This bit specifies the address offset of the even field (field 2) in memory in odd-/even-field capture mode. 0: Data are stored from the base address in external memory. 1: Data are stored from the base address plus the memory width in external memory.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 12	YMODE[2:0]	000	R/W	YC Data Transfer Mode These bits specify the transfer method of Y/CbCr when the data conversion mode (DTMD) is 10 (YC separation). 000: Both Y and CbCr data are transferred to memory. 001: Only Y data is transferred to memory as 8-bit data. 010: 10-bit CrCb data is converted to 8-bit CrCb data and both Y data and CbCr data are transferred to memory. 011: 10-bit CrCb data is converted to 8-bit CrCb data and only Y data is transferred to memory. 100: Setting prohibited 101: Setting prohibited 110: Setting prohibited 111: Setting prohibited
11	YC_THR	0	R/W	YC Data Through Mode Y and CbCr data are transferred to memory as 10-bit data in accordance with the input format when this bit is set to 1. 0: Y and CbCr data are transferred to memory in accordance with the setting in the YMODE[2:0] bits. 1: Y and CbCr data are transferred to memory as 10-bit data in accordance with the input format.
10 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	EXRGB	0	R/W	Extension RGB Conversion Mode 0: RGB data extension processing is not performed. 1: Data is extended to 32-bit RGB conversion when DTMD[1:0] is set to 00 or 01 as the data conversion mode.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	BPSM	0	R/W	Output Data Byte Swap Mode 0: Bytes are not swapped in output data. 1: Bytes are swapped in output data. Note: When YCbCr-422 data is output in big endian, data is transferred in the YUYV format in most cases. To transfer data in the UYVY format, set this bit to 1.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	ABIT	0	R/W	Alpha Bit This bit specifies the alpha value for data in ARGB-1555 output mode. 0: The alpha value is set to 0. 1: The alpha value is set to 1.
1, 0	DTMD[1:0]	00	R/W	Data Conversion Mode These bits set the format for storing RGB888 or YCbCr422 data after LUT conversion, in the external memory. *1 00: Data is not converted. 01: RGB is converted to ARGB before output. 10: YC is separated before output. *2 11: Setting prohibited

Note 1. The data conversion modes that can be set are shown in table 49.5. Do not set any other mode that is not listed in the table. RGB and YCbCr data after LUT conversion should be set as shown in Table 48.5.
Note 2. Do not set for any other data format except for YCbCr422; set YC separation only for YCbCr data format.

Table 48.5 Data Conversion Settings

• RGB Data Conversion Modes

Format of Data Stored in Memory	VnDMR/ YMODE[2:0]	VnDMR/ EXRGB	VnDMR/ YC_THR	VnDMR/ DTMD[1:0]	Remarks
RGB-565 (16 bits/pixel) format	000	0	0	00	
RGB-888 (32 bits/pixel) format	000	1	0	00	
ARGB-1555 (16 bits/pixel) format	000	0	0	01	The alpha bit is set with the ABIT bit in VnDMR.
ARGB-8888 (32 bits/pixel) format	000	1	0	01	The alpha bit is set with the A8BIT bit in VnDMR.

• YCbCr Data Conversion Modes

Format of Data Stored in Memory	VnDMR/ YMODE[2:0]	VnDMR/ EXRGB	VnDMR/ YC_THR	VnDMR/ DTMD[1:0]	Remarks
YCbCr-422 (8 bits) transfer	000	0	0	00	Set the BPSM bit in VnDMR to 1 to transfer in UYVY format in big endian. Make these settings if the setting of the INF[2:0] bits in VnMC is B'100.
YCbCr-422 (10 bits) transfer	000	0	1	00	
Y (8 bits)/CbCr (8 bits) separation transfer	000	0	0	10	CbCr transfer destination is determined by the VnUVAOF register setting.
Y (10 bits)/CbCr (10 bits) separation transfer	000	0	1	10	CbCr transfer destination is determined by the VnUVAOF register setting.
Y (8 bits) transfer	001	0	0	10	
Y (10 bits)/CbCr (8 bits) separation transfer	010	0	0	10	CbCr transfer destination is determined by the VnUVAOF register setting.
Y (10 bits) transfer	011	0	0	10	

Note: If any combination of values not listed above is specified, correct operation is not guaranteed.

• 8-bit user defined data Mode

Format of Data Stored in Memory	VnDMR/ YMODE[2:0]	VnDMR/ EXRGB	VnDMR/ YC_THR	VnDMR/ DTMD[1:0]	Remarks
RAW8	000	0	0	00	MIPI CSI-2 IF Only

48.2.18 Video n Data Mode Register 2 (VnDMR2)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FTEV	FTEH
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VLV[3:0]				—	HLV[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	FTEV	0	R/W	VSYNC Field Toggle Mode Enable The VSYNC field toggle mode changes the capture field signal level according to the count of input VSYNC signal assertion. As the VIN controls capture operation only when the input field signal level changes, select the VSYNC field toggle mode when capturing MIPI CSI-2 input data. 0: The field toggle function according to the VSYNC count is disabled. 1: The field toggle function according to the VSYNC count is enabled. The period before a toggle should be specified in the VLV bits. Note: Do not set both FTEH and FTEV at the same time.
16	FTEH	0	R/W	HSYNC Field Toggle Counter Enable 0: The field toggle function according to the capture active line is disabled. 1: The field toggle function according to the capture active line is enabled. The period before a toggle should be specified in the HLV bits. Note: Do not set both FTEH and FTEV at the same time.
15 to 12	VLV[3:0]	H'0	R/W	VSYNC Field Toggle Mode Transition Period These bits specify the count of vertical sync signal input before the VSYNC field toggle mode is entered. After a transition to the VSYNC field toggle mode, the capture field signal is toggled every time VSYNC is input. When a change in the input field signal is detected, the toggle mode is canceled. H'0: The field signal is toggled at every VSYNC input. H'1: Toggle mode is entered after VSYNC is input once. H'2: Toggle mode is entered after VSYNC is input two times. H'3: Toggle mode is entered after VSYNC is input three times. : H'E: Toggle mode is entered after VSYNC is input 14 times H'F: Toggle mode is entered after VSYNC is input 15 times. Note: If the field signal changes while the transition period is counted, the counter is initialized.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10 to 0	HLV[10:0]	H'000	R/W	HSYNC Field Toggle Count Value The HSYNC field toggle counter counts the capture active lines. If the external field signal does not change before the prespecified counter value is reached, the capture field signal is toggled. H'000: The field signal is toggled for every valid line. H'001: The field signal is toggled for every single valid line. H'002: The field signal is toggled for every two valid lines. : H'7FF: The field signal is toggled for every 2047 valid lines. Note: For the period before a toggle, specify a value greater than one VSYNC period.

48.2.19 Video n UV Address Offset Register (VnUVAOF)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	UVAOF[24:9]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UVAOF[8:0]									—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	UVAOF[24:0]	H'000 0000	R/W	UV Data Address Offset These bits specify the transfer offset address for the YC separation YCbCr-422 UV. Specify bits 31 to 7 of the physical address in 128-byte units. Note: The specified address should be equal to or greater than the Y transfer size. Otherwise, the overwriting of Y data occurs.
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

48.2.20 YC-RGB Conversion Coefficient Registers

YC → RGB color space conversion is performed with the following formula. Each of the coefficients can be set through the registers. Here, if 8-bit data format is selected, the coefficients set with the YMUL, CSUB, and YSUB bits in the VnCSCC1 register, the RCRMUL and GCRMUL bits in the VnCSCC2 register, and the GCBMUL and BCBMUL bits in the VnCSCC3 register are used to convert the color space. When 10-bit data format is selected, the YMUL2 bit in the VnCSCE1 register, the CSUB2 and YSUB2 bits in the VnCSCE2 register, the RCRMUL2 and GCRMUL2 bits in the VnCSCE3 register, and the GCBMUL2 and BCBMUL2 bits in the VnCSCE4 register are used in color space conversion.

$$R = \left(\begin{array}{c}\textit{VnCSCC1/}\\ \textit{YMUL[9:0]}\\ \textit{or}\\ \textit{VnCSCE1/}\\ \textit{YMUL2[13:0]}\end{array}\right) \times (Y - \left(\begin{array}{c}\textit{VnCSCC1/}\\ \textit{YSUB[7:0]}\\ \textit{or}\\ \textit{VnCSCE2/}\\ \textit{YSUB2[11:0]}\end{array}\right)) + \left(\begin{array}{c}\textit{VnCSCC2/}\\ \textit{RCRMUL[9:0]}\\ \textit{or}\\ \textit{VnCSCE3/}\\ \textit{RCRMUL2[13:0]}\end{array}\right) \times (Cr - \left(\begin{array}{c}\textit{VnCSCC1/}\\ \textit{CSUB[7:0]}\\ \textit{or}\\ \textit{VnCSCE2/}\\ \textit{CSUB2[11:0]}\end{array}\right))$$

$$G = \left(\begin{array}{c}\textit{VnCSCC1/}\\ \textit{YMUL[9:0]}\\ \textit{or}\\ \textit{VnCSCE1/}\\ \textit{YMUL2[13:0]}\end{array}\right) \times (Y - \left(\begin{array}{c}\textit{VnCSCC1/}\\ \textit{YSUB[7:0]}\\ \textit{or}\\ \textit{VnCSCE2/}\\ \textit{YSUB2[11:0]}\end{array}\right)) - \left(\begin{array}{c}\textit{VnCSCC2/}\\ \textit{GCRMUL[9:0]}\\ \textit{or}\\ \textit{VnCSCE3/}\\ \textit{GCRMUL2[13:0]}\end{array}\right) \times (Cr - \left(\begin{array}{c}\textit{VnCSCC1/}\\ \textit{CSUB[7:0]}\\ \textit{or}\\ \textit{VnCSCE2/}\\ \textit{CSUB2[11:0]}\end{array}\right)) - \left(\begin{array}{c}\textit{VnCSCC3/}\\ \textit{GCBMUL[9:0]}\\ \textit{or}\\ \textit{VnCSCE4/}\\ \textit{GCBMUL2[13:0]}\end{array}\right) \times (Cb - \left(\begin{array}{c}\textit{VnCSCC1/}\\ \textit{CSUB[7:0]}\\ \textit{or}\\ \textit{VnCSCE2/}\\ \textit{CSUB2[11:0]}\end{array}\right))$$

$$B = \left(\begin{array}{c}\textit{VnCSCC1/}\\ \textit{YMUL[9:0]}\\ \textit{or}\\ \textit{VnCSCE1/}\\ \textit{YMUL2[13:0]}\end{array}\right) \times (Y - \left(\begin{array}{c}\textit{VnCSCC1/}\\ \textit{YSUB[7:0]}\\ \textit{or}\\ \textit{VnCSCE2/}\\ \textit{YSUB2[11:0]}\end{array}\right)) + \left(\begin{array}{c}\textit{VnCSCC3/}\\ \textit{BCBMUL[9:0]}\\ \textit{or}\\ \textit{VnCSCE4/}\\ \textit{BCBMUL2[13:0]}\end{array}\right) \times (Cb - \left(\begin{array}{c}\textit{VnCSCC1/}\\ \textit{CSUB[7:0]}\\ \textit{or}\\ \textit{VnCSCE2/}\\ \textit{CSUB2[11:0]}\end{array}\right))$$

Note: Set the coefficients for ITU-R BT.601 (8 bits) as the initial values.

$$R = 1.164 \times (Y - 16) + 1.596 \times (Cr - 128)$$
$$G = 1.164 \times (Y - 16) - 0.813 \times (Cr - 128) - 0.392 \times (Cb - 128)$$
$$B = 1.164 \times (Y - 16) + 2.017 \times (Cb - 128)$$

48.2.20.1 Video n Color Space Conversion Coefficient 1 Register (VnCSCC1)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	YMUL[9:0]									
Initial value:	0	0	0	0	0	0	0	1	0	0	1	0	1	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	YSUB[7:0]								CSUB[7:0]							
Initial value:	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 16	YMUL[9:0]	H'129	R/W	Y Data Multiplication Coefficient These bits specify the multiplication coefficient for Y data in YCbCr-422 → RGB-888 color space conversion. (Initial value: 1.164) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256.
15 to 8	YSUB[7:0]	H'10	R/W	Y Data Subtraction Coefficient These bits specify the subtraction coefficient for Y data in YCbCr-422 → RGB-888 color space conversion. (Initial value: 16) Specify an unsigned 8-bit integer.
7 to 0	CSUB[7:0]	H'80	R/W	CbCr Data Subtraction Coefficient These bits specify the subtraction coefficient for Cb and Cr data in YCbCr-422 → RGB-888 color space conversion. (Initial value: 128) Specify an unsigned 8-bit integer.

48.2.20.2 Video n Color Space Conversion Coefficient 2 Register (VnCSCC2)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	RCRMUL[9:0]									
Initial value:	0	0	0	0	0	0	0	1	1	0	0	1	1	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GCRMUL[9:0]									
Initial value:	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 16	RCRMUL[9:0]	H'198	R/W	Cr Multiplication Coefficient for R Data Calculation These bits specify the Cr multiplication coefficient for the R data calculation equation in YCbCr-422 → RGB-888 color space conversion. (Initial value: 1.596) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	GCRMUL[9:0]	H'0D0	R/W	Cr Multiplication Coefficient for G Data Calculation These bits specify the Cr multiplication coefficient for the G data calculation equation in YCbCr-422 → RGB-888 color space conversion. (Initial value: 0.813) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256.

48.2.20.3 Video n Color Space Conversion Coefficient 3 Register (VnCSCC3)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	GCBMUL[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	BCBMUL[9:0]									
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 16	GCBMUL[9:0]	H'064	R/W	Cb Multiplication Coefficient for G Data Calculation These bits specify the Cb multiplication coefficient for the G data calculation equation in YCbCr-422 → RGB-888 color space conversion. (Initial value: 0.392) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	BCBMUL[9:0]	H'204	R/W	Cb Multiplication Coefficient for B Data Calculation These bits specify the Cb multiplication coefficient for the B data calculation equation in YCbCr-422 → RGB-888 color space conversion. (Initial value: 2.017) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256.

48.2.20.4 Video n YC → RGB Calculation Setting Extension Register 1 (VnCSCE1)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	YMUL2[13:0]													
Initial value:	0	0	0	1	0	0	1	0	1	0	0	1	1	1	1	1
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 0	YMUL2[13:0]	H'129F	R/W	Y Multiplication Coefficient 2 for RGB Calculation These bits specify the multiplication coefficient for Y data in YCbCr-422 → RGB-101010 color space conversion. (Initial value: 1.164) Specify an unsigned 14-bit integer obtained by multiplying the desired coefficient value by 4096.

48.2.20.5 Video n YC → RGB Calculation Setting Extension Register 2 (VnCSCE2)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	YSUB2[11:0]											
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CSUB2[11:0]											
Initial value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	YSUB2[11:0]	H'100	R/W	Y Subtraction Coefficient 2 for RGB Calculation These bits specify the subtraction coefficient for Y data in YCbCr-422 → RGB-101010 color space conversion. (Initial value: 256) Specify an unsigned 12-bit integer.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	CSUB2[11:0]	H'800	R/W	CbCr Subtraction Coefficient 2 for RGB Calculation These bits specify the subtraction coefficient for Cb and Cr data in YCbCr-422 → RGB-101010 color space conversion. (Initial value: 2048) Specify an unsigned 12-bit integer.

48.2.20.6 Video n YC → RGB Calculation Setting Extension Register 3 (VnCSCE3)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	RCRMUL2[13:0]													
Initial value:	0	0	0	1	1	0	0	1	1	0	0	0	1	0	0	1
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	GCRMUL2[13:0]													
Initial value:	0	0	0	0	1	1	0	1	0	0	0	0	0	0	1	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 16	RCRMUL2[13:0]	H'1989	R/W	Cr Multiplication Coefficient 2 for R Calculation These bits specify the Cr multiplication coefficient for the R data calculation equation in YCbCr-422 → RGB-101010 color space conversion. (Initial value: 1.596) Specify an unsigned 14-bit integer obtained by multiplying the desired coefficient value by 4096.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 0	GCRMUL2[13:0]	H'0D02	R/W	Cr Multiplication Coefficient 2 for G Calculation These bits specify the Cr multiplication coefficient for the G data calculation equation in YCbCr-422 → RGB-101010 color space conversion. (Initial value: 0.813) Specify an unsigned 14-bit integer obtained by multiplying the desired coefficient value by 4096.

48.2.20.7 Video n YC → RGB Calculation Setting Extension Register 4 (VnCSCE4)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	GCBMUL2[13:0]													
Initial value:	0	0	0	0	0	1	1	0	0	1	0	0	0	1	0	1
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	BCBMUL2[13:0]													
Initial value:	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 16	GCBMUL2[13:0]	H'0645	R/W	Cb Multiplication Coefficient 2 for G Calculation These bits specify the Cb multiplication coefficient for the G data calculation equation in YCbCr-422 → RGB-101010 color space conversion. (Initial value: 0.392) Specify an unsigned 14-bit integer obtained by multiplying the desired coefficient value by 4096.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 0	BCBMUL2[13:0]	H'2045	R/W	Cb Multiplication Coefficient 2 for B Calculation These bits specify the Cb multiplication coefficient for the B data calculation equation in YCbCr-422 → RGB-101010 color space conversion. (Initial value: 2.017) Specify an unsigned 14-bit integer obtained by multiplying the desired coefficient value by 4096.

48.2.21 UDS Control Registers

Note: The following operators, notations are defined for explaining the functions of UDS.
< x >: Discard decimal places of value x

48.2.21.1 Video n Scaling Control Registers (VnUDS_CTRL)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	AMD	—	—	—	—	—	—	—	—	—	BC	—	NE_RCR	NE_GY	NE_BCB
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	AMD	0	R/W	Pixel Count at Scale-Up Specifies the number of pixels generated through scale-up in the UDS. This bit setting is ignored for scale-down. 0: Pixel count after scale-up is 1 + <(n - 1) × scale-up factor> 1: Pixel count after scale-up is <n × scale-up factor> Note: n: Number of pixels input to UDS
29 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	BC	0	R/W	Pixel Component Interpolation Method at Scale-Up/Down Specifies the method for interpolating pixel components at scale-up/-down. 0: Bilinear or nearest neighbor interpolation method is used 1: Interpolation method equivalent to 4 to 17 taps in accordance with the scaling factor is used (multi-tap mode)
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	NE_RCR	0	R/W	R/Cr Interpolation Method When Bilinear/Nearest Neighbor Interpolation is Selected Specifies the interpolation method of the R/Cr component when bilinear/nearest neighbor interpolation is selected (BC = 0). 0: Bilinear method 1: Nearest neighbor method*
17	NE_GY	0	R/W	G/Y Interpolation Method When Bilinear/Nearest Neighbor Interpolation is Selected Specifies the interpolation method of the G/Y component when bilinear/nearest neighbor interpolation is selected (BC = 0). 0: Bilinear method 1: Nearest neighbor method*
16	NE_BCB	0	R/W	B/Cb Interpolation Method When Bilinear/Nearest Neighbor Interpolation is Selected Specifies the interpolation method of the B/Cb component when bilinear/nearest neighbor interpolation is selected (BC = 0). 0: Bilinear method 1: Nearest neighbor method*
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: *This method can be used only when the scale-up/-down factor is 1/1 to 1/4.

48.2.21.2 Video n Scaling Factor Registers (VnUDS_SCALE)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HMANT [3:0]				HFRAC [11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VMANT [3:0]				VFRAC [11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	HMANT[3:0]	H'0	R/W	Multiplier (Integral Part) of Horizontal Scaling Factor These bits specify the integral part of the horizontal scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size. A value from H'0 to H'F can be specified. Select a value within the range shown in Table 48.6
27 to 16	HFRAC[11:0]	H'000	R/W	Multiplier (Fractional Part) of Horizontal Scaling Factor These bits specify the fractional part of the horizontal scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size. A value from H'800 to H'FFF can be specified when an image is upscaled (the HMANT value is 0) in the horizontal direction. A value from H'000 to H'FFF can be specified when an image is downscaled (the HMANT value is not 0) in the horizontal direction. Select a value within the range shown in Table 48.6.
15 to 12	VMANT[3:0]	H'0	R/W	Multiplier (Integral Part) of Vertical Scaling Factor These bits specify the integral part of the vertical scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size. A value from H'0 to H'F can be specified. Select a value within the range shown in Table 48.7.
11 to 0	VFRAC[11:0]	H'000	R/W	Multiplier (Fractional Part) of Vertical Scaling Factor These bits specify the fractional part of the vertical scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size. A value from H'556 to H'FFF can be specified when an image is upscaled (the VMANT value is 0) in the vertical direction. A value from H'000 to H'FFF can be specified when an image is downscaled (the VMANT value is not 0) in the vertical direction. Select a value within the range shown in Table 48.7.

The HMANT and HFRAC bits set the scale-up/-down factor for an image in the horizontal direction, and the VMANT and VFRAC bits set the scale-up/-down factor for an image in the vertical direction. The UDS operation switches between horizontal scale-up and horizontal scale-down according to the HMANT and HFRAC bit settings, as shown in Table 48.6. (Setting a value outside the ranges of Table 48.6 in UDS operation is prohibited). Table 48.6 and Table 48.7 show the settings in the horizontal and vertical directions.

Note that settings for scaling in the horizontal direction and settings for scaling in the vertical direction can be made independently. Therefore, a setting for scale-up in the horizontal direction and scale-down in the vertical direction is possible. In such a case, because the formula (described later) for obtaining the image size after scale-up/-down is different between scale-up and scale-down, a formula matching the scale-up or scale-down operation should be selected independently for the horizontal direction and vertical direction.

Table 48.6 Switching of Horizontal Scale-Up/Down Operation According to HMANT and HFRAC Bit Settings

HMANT	HFRAC	UDS Operation
H'0	H'800 to H'FFF	Scale-up
H'1	H'000	Same size (no scale-up/-down)
	H'001 to H'FFF	Scale-down
H'2 to H'F	H'000 to H'FFF	

Table 48.7 Switching of Vertical Scale-Up/Down Operation According to VMANT and VFRAC Bit Settings

VMANT	VFRAC	UDS Operation
H'0	H'556 to H'FFF	Scale-up
H'1	H'000	Same size (no scale-up/-down)
	H'001 to H'FFF	Scale-down
H'2 to H'F	H'000 to H'FFF	

Described here is the method for calculating the size of the upscaled/downscaled image that was obtained based on this register setting. First, define the variables necessary for calculating the horizontal size of the upscaled/downscaled image, as shown below.

$$hscale = \frac{4096}{4096 \times m_h + f_h}$$

m_h is the value of VnUDS_SCALE.HMANT, and f_h is the value of VnUDS_SCALE.HFRAC.
This formula expresses the estimate of the scale-up/-down factor processed by the UDS. If the horizontal size of the image before scale-up/-down is set as $hsize_{org}$, the horizontal size of the image after scale-up/-down can be roughly obtained through $hsize_{org} \times$ hscale.
Similarly, define the variables necessary for calculating the vertical size of the upscaled/downscaled image.

$$vscale = \frac{4096}{4096 \times m_v + f_v}$$

When setting mv as the value of VnUDS_SCALE.VMANT, fv as the value of VnUDS_SCALE.VFRAC, and $vsize_{org}$ as the vertical size of the image before scale-up/-down, the vertical size of the image after scale-up/-down can be roughly obtained through $vsize_{org} \times$ vscale.
When the UDS performs scale-down with the settings of Table 48.6, using the variables defined so far, the horizontal size of the downscaled image $hsize_{down\_scaled}$ and the vertical size of the downscaled image $vsize_{down\_scaled}$ become as follows:

$$hsize_{down\_scaled} = \left\langle 1 + \left(\left\langle 1 + \frac{\langle hsize_{org} - 1 \rangle}{m_h'} \right\rangle - 1 \right) \times m_h' \times hscale \right\rangle$$

$$vsize_{down\_scaled} = \left\langle 1 + \left(\left\langle 1 + \frac{\langle vsize_{org} - 1 \rangle}{m_v'} \right\rangle - 1 \right) \times m_v' \times vscale \right\rangle$$

Since the division of hscale and vscale has to be executed at the end of the above formulas, respectively, formulas considering the order of operation become as follows:

$$hsize_{down\_scaled} = \left\langle 1 + \left(\left(\left\langle 1 + \frac{\langle hsize_{org} - 1 \rangle}{m_h'} \right\rangle - 1 \right) \times m_h' \times 4096 \right) \Big/ (4096 \times m_h + f_h) \right\rangle$$

$$vsize_{down\_scaled} = \left\langle 1 + \left(\left(\left\langle 1 + \frac{\langle vsize_{org} - 1 \rangle}{m_v'} \right\rangle - 1 \right) \times m_v' \times 4096 \right) \Big/ (4096 \times m_v + f_v) \right\rangle$$

The value of m_h' or m_v', which is shown in Table 48.8, changes according to the setting of VnUDS_SCALE.HMANT or VnUDS_SCALE.VMANT.

Table 48.8 m_h' or m_v' Setting

VnUDS_SCALE.HMANT Setting (VnUDS_SCALE.VMANT Setting)	m_h' (m_v')
1 to 3	1
4 to 7	2
8 to 15	4

When the UDS performs scale-up with the settings of Table 48.6 and VnUDS_CTRL.AMD is 0, the horizontal size of the upscaled image $hsize_{up\_scaled}$ and the vertical size of the upscaled image $vsize_{up\_scaled}$ become as follows:

$$hsize_{up\_scaled} = \langle 1 + (hsize_{org} - 1) \times hscale \rangle$$

$$vsize_{up\_scaled} = \langle 1 + (vsize_{org} - 1) \times vscale \rangle$$

Similar to the scale-down case, formulas considering the division of hscale and vscale become as follows:

$$hsize_{up\_scaled} = \langle 1 + ((hsize_{org} - 1) \times 4096) / (4096 \times m_h + f_h) \rangle$$

$$vsize_{up\_scaled} = \langle 1 + ((vsize_{org} - 1) \times 4096) / (4096 \times m_v + f_v) \rangle$$

When VnUDS_CTRL.AMD is 1, the horizontal size of the upscaled image $hsize_{up\_scaled}$ and the vertical size of the upscaled image $vsize_{up\_scaled}$ become as follows:

$$hsize_{up\_scaled} = \langle hsize_{org} \times hscale \rangle$$

$$vsize_{up\_scaled} = \langle vsize_{org} \times vscale \rangle$$

After considering the division of hscale and vscale, the formulas become as follows:

$$hsize_{up\_scaled} = \left\langle (hsize_{org} \times 4096) / (4096 \times m_h + f_h) \right\rangle$$

$$vsize_{up\_scaled} = \left\langle (vsize_{org} \times 4096) / (4096 \times m_v + f_v) \right\rangle$$

48.2.21.3 Video n Passband Registers (VnUDS_PASS_BWIDTH)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	BWIDTH_H [6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	BWIDTH_V [6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 16	BWIDTH_H [6:0]	H'00	R/W	Horizontal Signal Passband at Image Scale-Up/Down Set these bits following the method described later.
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	BWIDTH_V [6:0]	H'00	R/W	Vertical Signal Passband at Image Scale-Up/Down Set these bits following the method described later.

The method for setting the passband in the horizontal direction of an image is described. When the VnUDS_SCALE.HMANT bits for horizontal scale-up/-down factor setting are not 0, set the BWIDTH_H bits according to the following formula. When the VnUDS_SCALE.HMANT bits are 0, set 64 in the BWIDTH_H bits.

$$BWIDTH\_H = \left\langle 64 \times \frac{4096 \times m_h'}{4096 \times m_h + f_h} \right\rangle \quad \text{(VnUDS\_SCALE.HMANT} \neq 0\text{)}$$

$$BWIDTH\_H = 64 \quad \text{(VnUDS\_SCALE.HMANT} = 0\text{)}$$

m_h is the value of VnUDS_SCALE.HMANT, and f_h is the value of VnUDS_SCALE.HFRAC. For the m_h' value, see Table 48.8. The method for setting the passband in the vertical direction of an image is similar to that for the horizontal direction described earlier. Since only the correspondence relationship of the registers is changed as shown below, replace the variables in the previous explanation as shown below when reading.

BWIDTH_H → BWIDTH_V

$m_h' \rightarrow m_v'$

$m_h \rightarrow m_v$

$f_h \rightarrow f_v$

VnUDS_SCALE.HMANT → VnUDS_SCALE.VMANT

VnUDS_SCALE.HFRAC → VnUDS_SCALE.VFRAC

48.2.21.4 Video n UDS Output Size Clipping Registers (VnUDS_CLIP_SIZE)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	CL_HSIZE[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CL_VSIZE[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	CL_HSIZE [11:0]	H'000	R/W	Clipping Size of Horizontal Pixel Count after Scale-Up/-Down The horizontal width of an image output from the scaling filter is adjusted (clipped or padded) to match the pixel count set in the CL_HSIZE bits. The setting range is 4 to 2,048 in a scale-up/down operation (see Table 48.6). These bits always have to be set when using the UDS, regardless of the scale-up, scale-down, or no-scaling setting by the VnUDS_SCALE register.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	CL_VSIZE [11:0]	H'000	R/W	Clipping Size of Vertical Pixel Count after Scale-Up/-Down The vertical width of an image output from the scaling filter is adjusted (clipped or padded) to match the pixel count set in the CL_VSIZE bits. The setting range is 4 to 2,048 in a scale-up/down operation (see Table 48.6). These bits always have to be set when using the UDS, regardless of the scale-up, scale-down, or no-scaling setting by the VnUDS_SCALE register.

Figure 48.3 shows the configuration of the UDS. The UDS consists of a scaling filter and clipping circuit, such as the configuration shown in Figure 48.3. The scaling filter and clipping circuit are independent of each other.

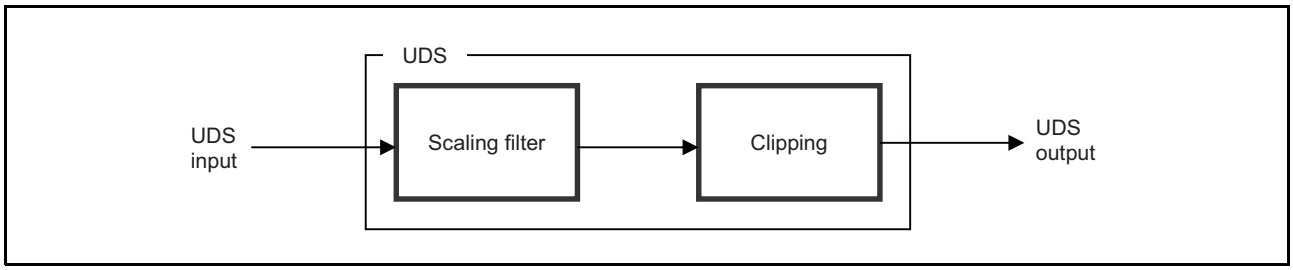


Figure 48.3 UDS Configuration

The size of the image actually output by the scaling filter (refer to section 48.2.21.2, Video n Scaling Factor Registers (VnUDS_SCALE) for calculation) is determined from the size of the image input to the scaling filter and the VnUDS_SCALE setting; that is, two types of image are output by the UDS according to the relationship between the size of the image actually output from the scaling filter ($hsize_{scaled}$ and $vsize_{scaled}$) and this register setting as shown in Figure 48.4.

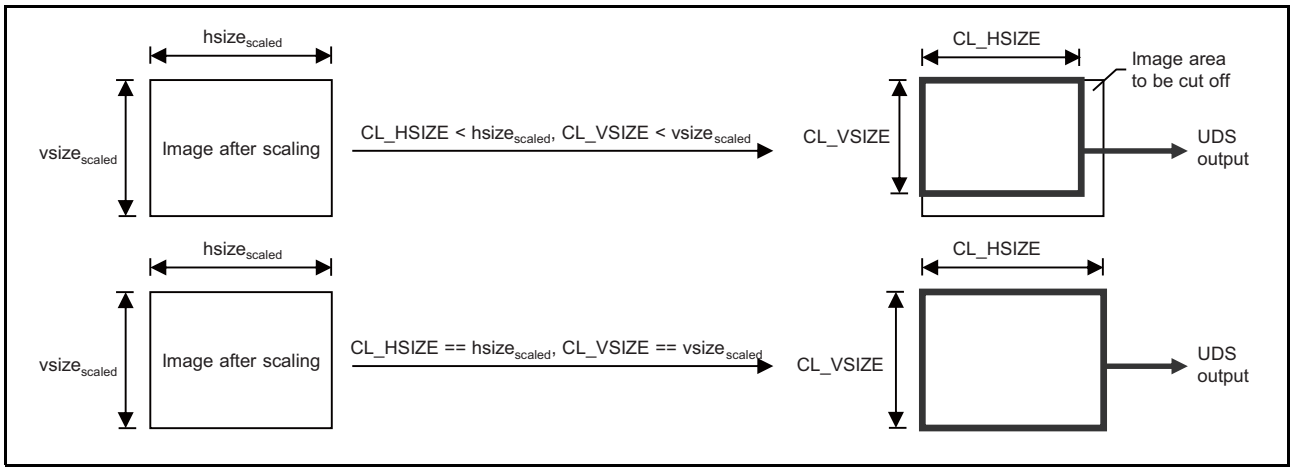


Figure 48.4 UDS Output Image for Each CL_HSIZE/VSIZE Setting

When the settings of the CL_HSIZE and CL_VSIZE bits are smaller than the horizontal and vertical pixel counts ($hsize_{scaled}$ and $vsize_{scaled}$) actually output by the scaling filter, the upscaled/downscaled image is clipped to become the UDS output image.

When the settings of the CL_HSIZE and CL_VSIZE bits are equal to the horizontal and vertical pixel counts ($hsize_{scaled}$ and $vsize_{scaled}$) actually output by the scaling filter, the image actually output by the scaling filter becomes the UDS output image without change.

Do not set the CL_HSIZE and CL_VSIZE bits to values that satisfy CL_HSIZE > $hsize_{scaled}$ or CL_VSIZE > $vsize_{scaled}$.

48.2.22 Video n Lookup Table Pointer (VnLUTP)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	LTYPR[9:0]										LTCBPR[9:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTCBPR[9:0]						LTCRPR[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 20	LTYPR[9:0]	H'000	R/W	Lookup Table Y Pointer These bits set the LUT pointer to the Y and R data obtained as the result of color space conversion.
19 to 10	LTCBPR[9:0]	H'000	R/W	Lookup Table Cb Pointer These bits set the LUT pointer to the Cb and G data obtained as the result of color space conversion.
9 to 0	LTCRPR[9:0]	H'000	R/W	Lookup Table Cr Pointer These bits set the LUT pointer to the Cr and B data obtained as the result of color space conversion.

Note: Set the LUT pointer corresponding to the upper 10 bits of the 12-bit data of color space conversion result.
The access pointer to the LUT will be automatically incremented by writing to the VnLUTD register.
There will be no automatic increment at read access.

48.2.23 Video n Lookup Table Data Register (VnLUTD)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	LTYDT[7:0]							
Initial value:	0	0	0	0	0	0	0	0	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTCBDT[7:0]								LTCRDT[7:0]							
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	LTYDT[7:0]	H'000	R/W	Lookup Table Y Data These bits set the LUT conversion data for Y and R data after color space conversion.
15 to 8	LTCBDT[7:0]	H'000	R/W	Lookup Table Cb Data These bits set the LUT conversion data for Cb and G data after color space conversion.
7 to 0	LTCRDT[7:0]	H'000	R/W	Lookup Table Cr Data These bits set the LUT conversion data for Cr and B data after color space conversion.

Note: The 8-bit data after LUT conversion is subjected to upper shift to carry out capture control as 12-bit data.
As for reading from the VnLUTD register, the data read the second time, including dummy read, after the VnLUTP register has been set is valid.

48.2.24 RGB-YC Conversion Coefficient Registers

Color space conversion from RGB to YC is done with the following formula. Each of the coefficients can be set with the registers.

$$Y = YCLRP \times R + YCLGP \times G + YCLBP \times B + YCLAP$$
$$Cb = CBCLRP \times R + CBCLGP \times G + CBCLBP \times B + CBCLAP$$
$$Cr = CRCLRP \times R + CRCLGP \times G + CRCLBP \times B + CRCLAP$$

Note: Set the coefficients for ITU-R BT.601 (8 bits) as the initial value.

$$Y = 0.257 \times R + 0.504 \times G + 0.098 \times B + 16$$
$$Cb = -0.148 \times R - 0.291 \times G + 0.439 \times B + 128$$
$$Cr = 0.439 \times R - 0.368 \times G - 0.071 \times B + 128$$

48.2.24.1 Video n RGB → Y Calculation Setting Register 1 (VnYCCR1)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	YCLRP[12:0]												
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	YCLRP[12:0]	H'0107	R/W	R Multiplication Coefficient for Y Calculation These bits specify the R multiplication coefficient for the Y data calculation equation in RGB-888 → YCbCr-422 color space conversion. (Initial value: 263) Specify a 13-bit signed integer obtained by multiplying the desired coefficient value by 1024. The MSB is a sign bit.

48.2.24.2 Video n RGB → Y Calculation Setting Register 2 (VnYCCR2)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	YCLBP[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	YCLGP[12:0]												
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 16	YCLBP[12:0]	H'0064	R/W	B Multiplication Coefficient for Y Calculation These bits specify the B multiplication coefficient for the Y data calculation equation in RGB-888 → YCbCr-422 color space conversion. (Initial value: 100) Specify a 13-bit signed integer obtained by multiplying the desired coefficient value by 1024. The MSB is a sign bit.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	YCLGP[12:0]	H'0204	R/W	G Multiplication Coefficient for Y Calculation These bits specify the G multiplication coefficient for the Y data calculation equation in RGB-888 → YCbCr-422 color space conversion. (Initial value: 516) Specify a 13-bit signed integer obtained by multiplying the desired coefficient value by 1024. The MSB is a sign bit.

48.2.24.3 Video n RGB → Y Calculation Setting Register 3 (VnYCCR3)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	YEXPEN	—	—	YCLSFT[4:0]					YCLHEN	—	—	—	—	—	—	YCLCEN
Initial value:	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	YCLAP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	YEXPEN	0	R/W	Y Calculation Sign Extension Enable This bit controls the sign extension for Y data calculation in RGB888 → YCbCr422 color space conversion. 0: Disables YC conversion sign bit. 1: Enables YC conversion sign bit.
30, 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 24	YCLSFT[4:0]	H'0A	R/W	Y Calculation Shift Down Volume These bits set the amount of down shift for Y calculation in RGB888 → YCbCr422 color space conversion (Initial value: 10) Unit: Bit shift count
23	YCLHEN	0	R/W	Y Calculation Shift Down Result Round-Off Enable This bit enables round-off process for Y data calculation in RGB888 → YCbCr422 color space conversion 0: Round down to down shift process 1: Round-off to down shift process is enabled.
22 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	YCLCEN	0	R/W	Y Calculation Data Clip Enable This bit enables data clipping process for Y data calculation in RGB888 → YCbCr422 color space conversion. 0: Data clipping process is disabled. 1: Data clipping process is enabled.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	YCLAP[11:0]	H'010	R/W	Y Calculation Data Normalized Additional Value These bits set the Y data addition constant for RGB888 → YCbCr422 color space conversion. (Initial value: 16)

48.2.24.4 Video n RGB → Cb Calculation Setting Register 1 (VnCBCCR1)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CBCLRP[12:0]												
Initial value:	0	0	0	1	1	1	1	1	0	1	1	0	1	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	CBCLRP[12:0]	H'1F68	R/W	R Multiplication Coefficient for Cb Calculation These bits specify the R multiplication coefficient for the Cb data calculation equation in RGB-888 → YCbCr-422 color space conversion. (Initial value: -152) Specify a 13-bit signed integer obtained by multiplying the desired coefficient value by 1024. The MSB is a sign bit.

48.2.24.5 Video n RGB → Cb Calculation Setting Register 2 (VnCBCCR2)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CBCLBP[12:0]												
Initial value:	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CBCLGP[12:0]												
Initial value:	0	0	0	1	1	1	1	0	1	1	0	1	0	1	1	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 16	CBCLBP[12:0]	H'01C2	R/W	B Multiplication Coefficient for Cb Calculation These bits specify the B multiplication coefficient for the Cb data calculation equation in RGB-888 → YCbCr-422 color space conversion. (Initial value: 450) Specify a 13-bit signed integer obtained by multiplying the desired coefficient value by 1024. The MSB is a sign bit.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	CBCLGP[12:0]	H'1ED6	R/W	G Multiplication Coefficient for Cb Calculation These bits specify the G multiplication coefficient for the Cb data calculation equation in RGB-888 → YCbCr-422 color space conversion. (Initial value: -298) Specify a 13-bit signed integer obtained by multiplying the desired coefficient value by 1024. The MSB is a sign bit.

48.2.24.6 Video n RGB → Cb Calculation Setting Register 3 (VnCBCCR3)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CBEXPEN	—	—	CBCLSFT[4:0]					CBCLHEN	—	—	—	—	—	—	CBCLCEN
Initial value:	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CBCLAP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CBEXPEN	0	R/W	Cb Calculation Sign Extension Enable This bit controls the sign extension for Cb data calculation in RGB888 → YCbCr422 color space conversion. 0: Disables YC conversion sign bit. 1: Enables YC conversion sign bit.
30, 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 24	CBCLSFT[4:0]	H'0A	R/W	Cb Calculation Shift Down Volume These bits set the amount of down shift for Cb calculation in RGB888 → YCbCr422 color space conversion (Initial value: 10) Unit: Bit shift count
23	CBCLHEN	0	R/W	Cb Calculation Shift Down Result Round-Off Enable This bit enables round-off process for Cb data calculation in RGB888 → YCbCr422 color space conversion 0: Round down to down shift process 1: Round-off to down shift process is enabled.
22 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CBCLCEN	0	R/W	Cb Calculation Data Clip Enable This bit enables data clipping process for Cb data calculation in RGB888 → YCbCr422 color space conversion. 0: Data clipping process is disabled. 1: Data clipping process is enabled.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	CBCLAP[11:0]	H'080	R/W	Cb Calculation Data Normalized Additional Value These bits set the Cb data addition constant for RGB888 → YCbCr422 color space conversion. (Initial value: 128)

48.2.24.7 Video n RGB → Cr Calculation Setting Register 1 (VnCRCCR1)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CRCLRP[12:0]												
Initial value:	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	CRCLRP[12:0]	H'01C2	R/W	R Multiplication Coefficient for Cr Calculation These bits specify the R multiplication coefficient for the Cr data calculation equation in RGB-888 → YCbCr-422 color space conversion. (Initial value: 450) Specify a 13-bit signed integer obtained by multiplying the desired coefficient value by 1024. The MSB is a sign bit.

48.2.24.8 Video n RGB → Cr Calculation Setting Register 2 (VnCRCCR2)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CRCLBP[12:0]												
Initial value:	0	0	0	1	1	1	1	1	1	0	1	1	0	1	1	1
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CRCLGP[12:0]												
Initial value:	0	0	0	1	1	1	1	0	1	0	0	0	0	1	1	1
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 16	CRCLBP[12:0]	H'1FB7	R/W	B Multiplication Coefficient for Cr Calculation These bits specify the B multiplication coefficient for the Cr data calculation equation in RGB-888 → YCbCr-422 color space conversion. (Initial value: -73) Specify a 13-bit signed integer obtained by multiplying the desired coefficient value by 1024. The MSB is a sign bit.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	CRCLGP[12:0]	H'1E87	R/W	G Multiplication Coefficient for Cr Calculation These bits specify the G multiplication coefficient for the Cr data calculation equation in RGB-888 → YCbCr-422 color space conversion. (Initial value: -377) Specify a 13-bit signed integer obtained by multiplying the desired coefficient value by 1024. The MSB is a sign bit.

48.2.24.9 Video n RGB → Cr Calculation Setting Register 3 (VnCRCCR3)

Note: Availability of channels: n = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CREXPEN	—	—	CRCLSFT[4:0]					CRCLHEN	—	—	—	—	—	—	CRCLCEN
Initial value:	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CRCLAP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CREXPEN	0	R/W	Cr Calculation Sign Extension Enable This bit controls the sign extension for Cr data calculation in RGB888 → YCbCr422 color space conversion. 0: Disables YC conversion sign bit. 1: Enables YC conversion sign bit.
30, 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 24	CRCLSFT[4:0]	H'0A	R/W	Cr Calculation Shift Down Volume These bits set the amount of down shift for Cr calculation in RGB888 → YCbCr422 color space conversion (Initial value: 10) Unit: Bit shift count
23	CRCLHEN	0	R/W	Cr Calculation Shift Down Result Round-Off Enable This bit enables round-off process for Cr data calculation in RGB888 → YCbCr422 color space conversion 0: Round down to down shift process 1: Round-off to down shift process is enabled.
22 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CRCLCEN	0	R/W	Cr Calculation Data Clip Enable This bit enables data clipping process for Cr data calculation in RGB888 → YCbCr422 color space conversion. 0: Data clipping process is disabled. 1: Data clipping process is enabled.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	CRCLAP[11:0]	H'080	R/W	Cr Calculation Data Normalized Additional Value These bits set the Cr data addition constant for RGB888 → YCbCr422 color space conversion. (Initial value: 128)

48.3 Operation

48.3.1 Input Interface

The VIN captures video data in the MIPI CSI-2 interface and stores it in external memory. The following tables show the interface and data format supported.

Table 48.9 Supported Interfaces

Input interface	MIPI CSI-2	YCbCr-422	8 bits	Supported
			10 bits	Supported
		RGB-888	24 bits	Supported
		8-bit user defined data (RAW8)	8 bits	Supported
Output interface	RGB output	RGB-565	16 bits	Supported
		ARGB-1555	16 bits	Supported
		RGB-888	32 bits	Supported
		ARGB-8888	32 bits	Supported
	YCbCr output	YCbCr-422 multiplexed	8 bits	Supported
			10 bits	Supported
		YCbCr-422 separated Y/CbCr	Y: 8 bits CbCr: 8 bits	Supported
			Y: 10 bits CbCr: 8 bits	Supported
			Y: 10 bits CbCr: 10 bits	Supported
		YCbCr-422 separated only Y	8 bits	Supported
			10 bits	Supported

Note 1. When capturing MIPI CSI-2 input data, be sure to enable the internal field signal generation function.
Note 2. Dithering is performed for RGB-888 → RGB-565 conversion.
Note 3. RGB data is converted and transferred to memory from the video module after 8-bit precision conversion.
Note 4. For details of output interfaces, see section 48.3.9, Output Data Format.

48.3.2 Capture Mode

With the VIN, either of the single frame capture mode or continuous frame capture mode can be selected.
Specifying the capture field in IM bits of the video n main control register (VnMC) and then setting the SC bit in the video n frame capture register (VnFC) to 1 provides single frame capture mode. In this mode, the current frame is captured when the SC bit write timing (current scanline position) is smaller than the value of the video n start line pre-clip register (VnSLPrC), or the next frame is captured in other cases. The capture data is transferred to the memory address that is set in the video n memory base 1 register (VnMB1). For using single frame capture mode, necessary procedure of register setting is as followings.

1. Set 1 to the module enable bit (ME) in the video n main control register (VnMC).
2. Set 1 to the single frame capture mode bit (SC) in the video n frame capture register (VnFC).

Note: When using this setting, don't set 2'b01 to the interlace mode bits (IM) in the video n main control register (VnMC).

Specifying the capture field in IM bits of VnMC and then setting the CC bit of VnFC to 1 provides continuous frame capture mode, in which capture data is sequentially transferred to the addresses that are set in MB1 to MB3. In this case, the latest captured frame ID is shown in the FBS bits in the video n module status register (VnMS). For using continuous frame capture mode, necessary procedure of register setting is as followings.

1. Set 1 to the module enable bit (ME) in the video n main control register (VnMC).
2. Set 1 to the continuous frame capture mode bit (CC) in the video n frame capture register (VnFC)

Note: Don't set 1 to the single frame capture mode bit (SC) and the continuous frame capture mode bit (CC) at the same time.

When capturing progressive data, in which the field signal does not change, use the internal field signal generation function to toggle the internal field signal. For details, see section 48.2.18, Video n Data Mode Register 2 (VnDMR2).

48.3.3 Size Clipping

Image data that has been captured is pre-clipped according to the settings of the following registers: Video n start line pre-clip (VnSLPrC), Video n end line pre-clip (VnELPrC), Video n start pixel pre-clip (VnSPPrC), and Video n end pixel pre-clip (VnEPPrC).
For post-clipping after horizontal or vertical scaling, see section 48.2.21.4, Video n UDS Output Size Clipping Registers (VnUDS_CLIP_SIZE).

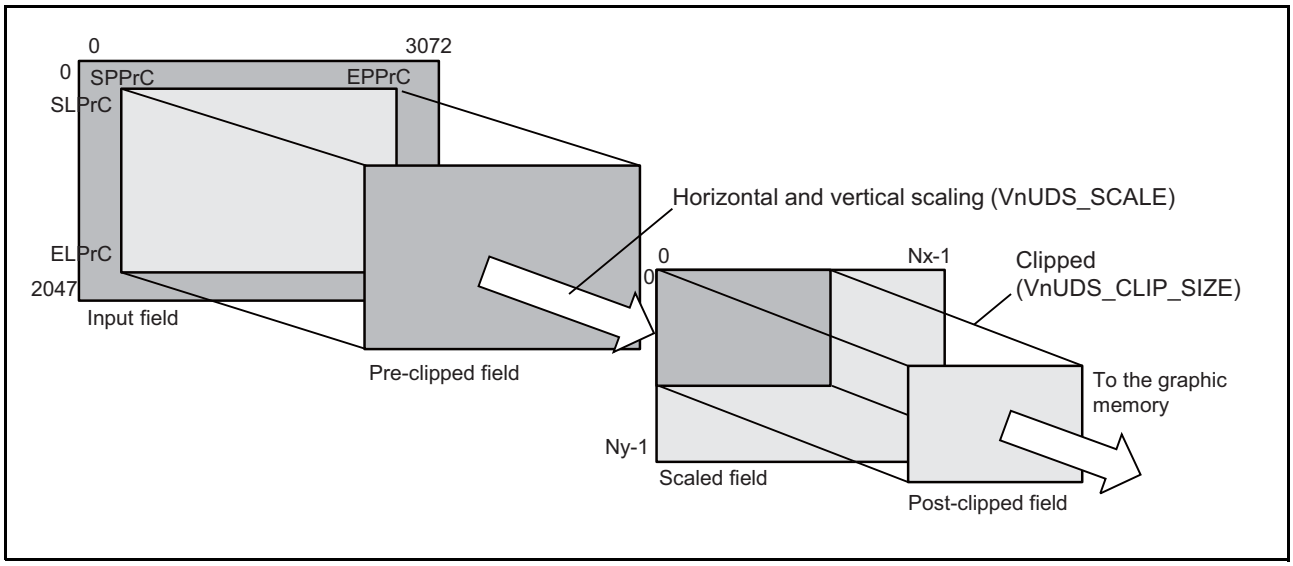


Figure 48.5 Example of Clipping

For all the post-clipped lines, the lengths of individual lines written into the memory are defined by the video n image stride register (VnIS). The setting can be larger than the post-clipped frame width, but cannot be smaller than the width. The VnIS register must be filled with a value larger than the horizontal post-clipping width. The input field in the above figure shows an effective image area from the video decoder; the VIN does not allow anything exceeding the image area to be captured.

Note: Each of the following registers specifies a distance from the starting point in the effective image area: Video n start line pre-clip (VnSLPrC), Video n end line pre-clip (VnELPrC), Video n start pixel pre-clip (VnSPPrC), and Video n end pixel pre-clip (VnEPPrC).

48.3.4 Vertical Scaling

For details on scaling in the vertical direction, see section 48.2.21.2, Video n Scaling Factor Registers (VnUDS_SCALE).

48.3.5 Horizontal Scaling

For details on scaling in the horizontal direction, see section 48.2.21.2, Video n Scaling Factor Registers (VnUDS_SCALE).

48.3.6 Color Conversion Function

(1) YC-RGB Color Conversion

When the data input format is YCbCr, set the BPS bit in the VnMC register to 0 to convert YCbCr data to RGB data. If an 8-bit data format is used for the capture interface, the color conversion is carried out according to the matrix coefficients set in the VnCSCC1, VnCSCC2, and VnCSCC3 registers. Otherwise, it is carried out according to the matrix coefficients set in the VnCSCE1, VnCSCE2, VnCSCE3, and VnCSCE4 registers. All of the input YCbCr data is extended to 12-bit data before carrying out color conversion.
Here, if the BPS bit in VnMC is set to 1, the data is stored in memory as it is in YCbCr format.

$$R = \left[\begin{array}{c}VnCSCC1/\\YMUL[9:0]\\or\\VnCSCE1/\\YMUL2[13:0]\end{array}\right] \times \left(Y - \left[\begin{array}{c}VnCSCC1/\\YSUB[7:0]\\or\\VnCSCE1/\\YSUB2[11:0]\end{array}\right]\right) + \left[\begin{array}{c}VnCSCC2/\\RCRMUL[9:0]\\or\\VnCSCE3/\\RCRMUL2[13:0]\end{array}\right] \times \left(Cr - \left[\begin{array}{c}VnCSCC1/\\CSUB[7:0]\\or\\VnCSCE1/\\CSUB2[11:0]\end{array}\right]\right)$$

$$G = \left[\begin{array}{c}VnCSCC1/\\YMUL[9:0]\\or\\VnCSCE1/\\YMUL2[13:0]\end{array}\right] \times \left(Y - \left[\begin{array}{c}VnCSCC1/\\YSUB[7:0]\\or\\VnCSCE1/\\YSUB2[11:0]\end{array}\right]\right) - \left[\begin{array}{c}VnCSCC2/\\GCRMUL[9:0]\\or\\VnCSCE3/\\GCRMUL2[13:0]\end{array}\right] \times \left(Cr - \left[\begin{array}{c}VnCSCC1/\\CSUB[7:0]\\or\\VnCSCE1/\\CSUB2[11:0]\end{array}\right]\right) - \left[\begin{array}{c}VnCSCC3/\\GCBMUL[9:0]\\or\\VnCSCE4/\\GCBMUL2[13:0]\end{array}\right] \times \left(Cb - \left[\begin{array}{c}VnCSCC1/\\CSUB[7:0]\\or\\VnCSCE1/\\CSUB2[11:0]\end{array}\right]\right)$$

$$B = \left[\begin{array}{c}VnCSCC1/\\YMUL[9:0]\\or\\VnCSCE1/\\YMUL2[13:0]\end{array}\right] \times \left(Y - \left[\begin{array}{c}VnCSCC1/\\YSUB[7:0]\\or\\VnCSCE1/\\YSUB2[11:0]\end{array}\right]\right) + \left[\begin{array}{c}VnCSCC3/\\BCBMUL[9:0]\\or\\VnCSCE4/\\BCBMUL2[13:0]\end{array}\right] \times \left(Cb - \left[\begin{array}{c}VnCSCC1/\\CSUB[7:0]\\or\\VnCSCE1/\\CSUB2[11:0]\end{array}\right]\right)$$

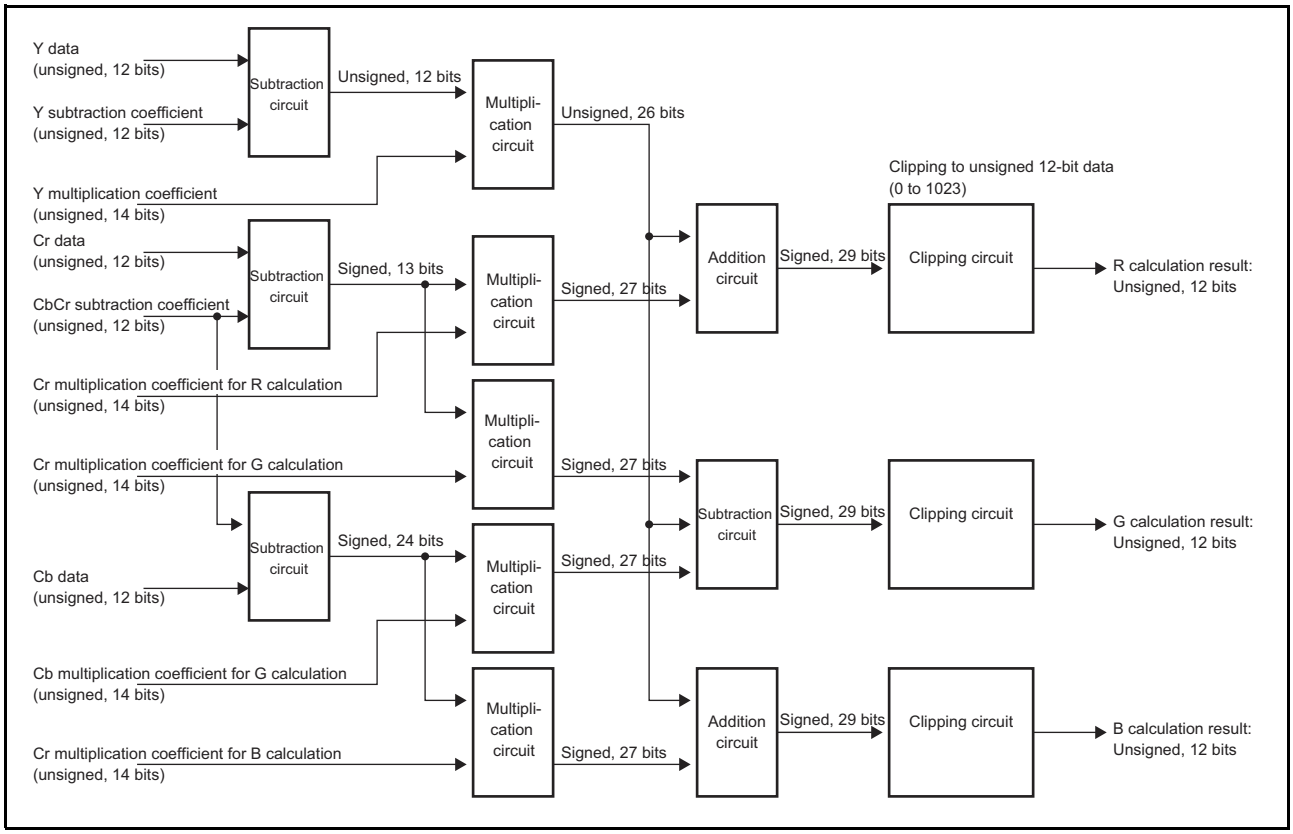


Figure 48.6 YCbCr → RGB Calculation Circuit Configuration

Examples of color space that can be set in YC-RGB conversion coefficient registers are shown below. See the register descriptions for details of the coefficients.

Table 48.10 Examples of YC-RGB Conversion Coefficient Register Settings

YC-RGB Conversion Coefficient	YMUL	YSUB	CSUB	RCRMUL	GCRMUL	GCBMUL	BCBMUL
ITU-R BT.601 (initial value) 16 ≤ Y ≤ 235, 16 ≤ Cb, Cr ≤ 240	1.164	16	128	1.596	0.813	0.392	2.017
Luminance expansion example 1 ≤ Y ≤ 254, 16 ≤ Cb, Cr ≤ 240	1.008	1	128	1.596	0.392	0.813	2.017

Specify an integer in each addition coefficient bit field. For each multiplication coefficient, specify a value obtained by multiplying the desired coefficient value by 256.

Example: When the desired multiplication coefficient is 1.164

1.164 × 256 = 297 (set value: B'0100101001)

Note 1. In order to process the data outside the range prescribed by the ITU-R BT.601 standard, set the CLP[1:0] bits in VnMC to 11. The data is clipped to the specified value before color space conversion.

Note 2. The YC-RGB color conversion data is unconditionally rounded to the 0 ≤ R, G, B ≤ 255 range.

(2) RGB-YC Color Conversion

When the data input format is RGB, set the BPS bit in VnMC to 0 for color conversion of RGB data into YCbCr data format. Color conversion from RGB to YCbCr is done according to the matrix coefficients set in RGB-YC conversion coefficient registers (VnYCCR1-VnYCCR3/VnCBCCR1-VnCBCCR3/VnCRCCR1-VnCRCCR3). All of the input RGB data is extended to 12-bit data before color conversion.
If 1 is set to the BPS bit in VnMC, the data will be stored in the memory as it is in RGB format.

$$Y = ((YCLRP \times R + YCLGP \times G + YCLBP \times B) \times 2^{YCLSFT}) + YCALP$$
$$Cb = ((CBCLRP \times R + CBCLGP \times G + CBCLBP \times B) \times 2^{CBCLSFT}) + CBCALP$$
$$Cr = ((CRCLRP \times R + CRCLGP \times G + CRCLBP \times B) \times 2^{CRCLSFT}) + CRCALP$$

The following data rounding functions of RGB-YCbCr color conversion function can be independently set to each pixel. The circuit configuration of Y data is given below.

Table 48.11 Data Rounding Functions of RGB-YC Color Conversion Function

Function	Symbol	Description
Sign extension enable	YEXPEN	Enables/disables signed bits of the matrix multiplication result.
Multiplication result shift down amount	YCLSFT[4:0]	Amount of shift down in the matrix multiplication result
Rounding off enable	YCLHEN	Enables/disables rounding off to the shift down amount.
Clipping enable	YCLCEN	Enables/disables data rounding process between 0 to 1023 of the output data.

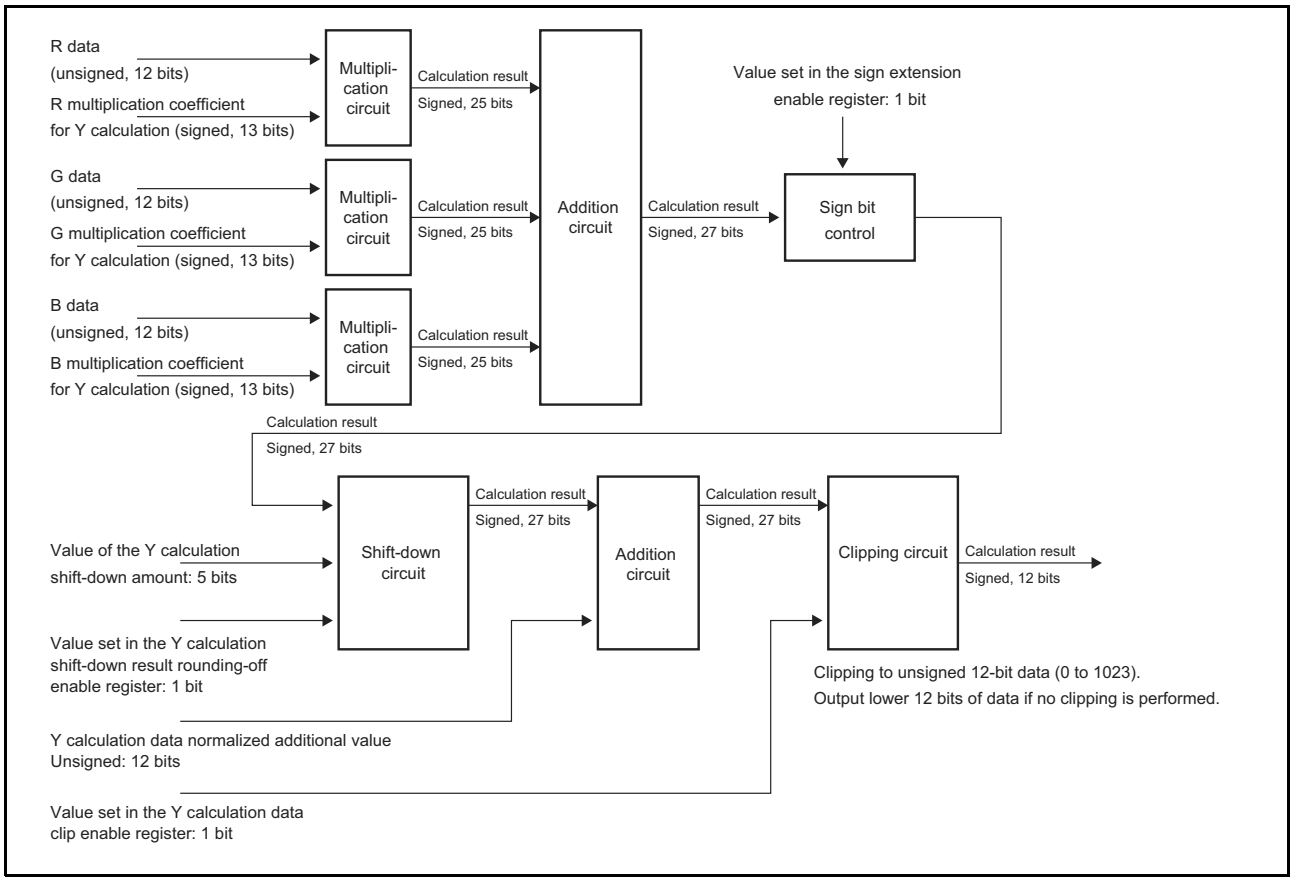


Figure 48.7 Y Data Circuit Configuration of RGB → YCbCr Color Conversion Function

48.3.7 Image Data Format Conversion Functions

(1) Lookup Table (LUT) Precision Conversion Function

Set the LUTE bit in VnMC to 1 to enable table conversion of each pixel data of Y, Cb, and Cr or R, G, and B data after color conversion.

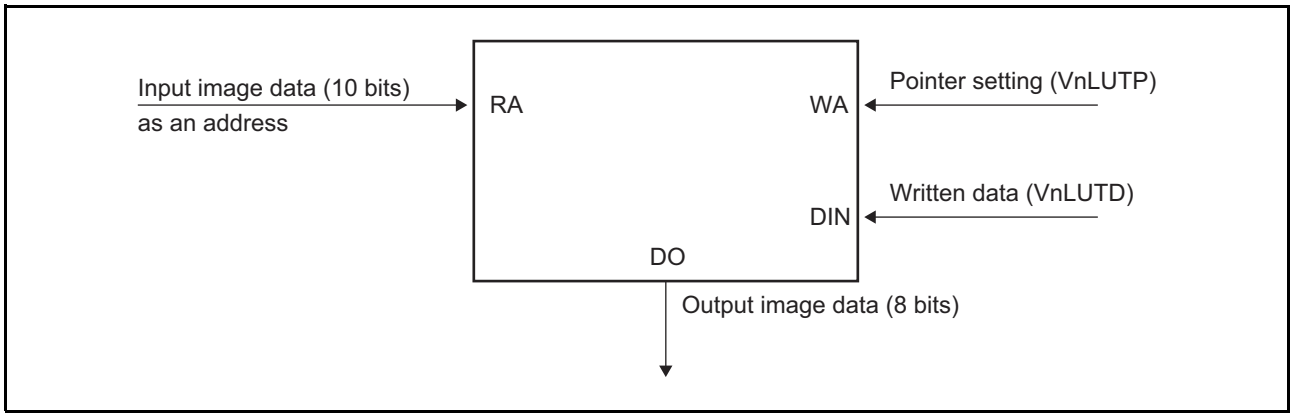


Figure 48.8 Lookup Table

Pointer Access:

The pointer is incremented every time a write access to the VnLUTD register is made.
The incremented pointer value is read when the VnLUTP register is read. Since the pointer is not incremented at read access to the VnLUTD register, the pointer should be set just before reading.
The procedure given below must be followed to access the lookup table.
The lookup table cannot be accessed from the CPU during the conversion process.
(For write/read access to the LUT from the CPU, the LUTE bit in VnMC should be 0.)

Write access:

1. Write the access destination address in VnLUTP.
2. Write the data in VnLUTD. Thereby the data will be reflected in the lookup table.
   Also the pointer will automatically increase by 1.
3. Writing to VnLUTD will consecutively update the lookup table.

Read Access:

1. Write the access destination address in VnLUTP.
2. Read VnLUTD. This is dummy read; thus the read data needs to be discarded.
3. Read VnLUTD. This reading allows the address data written in VnLUTP to be acquired.

[Notes on Using the Lookup Table]

1. Set the lookup table only after capture operation has stopped.
2. Rewrite all before using the lookup table.
3. Data set to the lookup table should be within the range compatible to the input bit width.

(2) YCbCr422 → YC Separation Function

When transferring YCbCr data to memory, Y data and CbCr data can be separated and transferred to different address spaces.

To perform YCbCr separation transfer, set the DTMD[1:0] bits in the VnDMR register to B'10. In this case, the Y data will be transferred to the address set in the memory base address register and CbCr data will be transferred to the address obtained by adding the value set in the VnUVAOF register to the memory base address register.

If the YMODE[0] bits in the VnDMR register are set, only Y data will be transferred to memory and CbCr data is not transferred to memory.

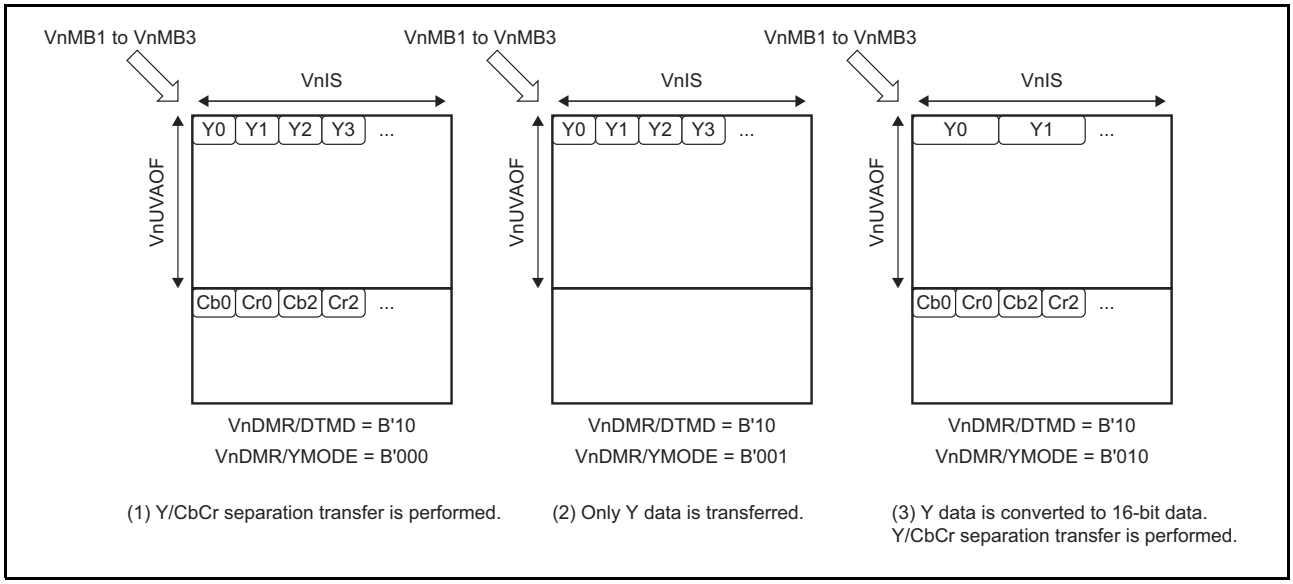


Figure 48.9 Y/Cb/Cr Separation in Big Endian

(3) Dithering Function

Dithering is performed when the internal RGB-888 format after color space conversion is converted to the RGB-565 or RGB-1555 format. The dithering mode can be selected with the DC[1:0] bits in VnMC.

(a) Dithering with Cumulative Addition

Set the DC[1:0] bits in VnMC to B'00 to perform dithering using the cumulative addition method in horizontal pixel units.

Rn [7:3] ≤ (Rx[7:0] + Rn-1[2:0]) >> 3
Gn [7:2] ≤ (Gx[7:0] + Gn-1[1:0]) >> 2
Bn [7:3] ≤ (Bx[7:0] + Bn-1[2:0]) >> 3

Where:
(Rn, Gn, Bn) = Output RGB-565 pixel
(Rx, Gx, Bx) = Input RGB-888 pixel
(Rn-1, Gn-1, Bn-1) = Pseudo random error (LSB of the cumulative error)

(b) Ordered Dithering

Set the DC[1:0] bits in VnMC to B'01 to perform dithering using the ordered dithering method.

Rn [7:3] ≤ (Rx[7:0] + D42xy) >> 3
Gn [7:2] ≤ (Gx[7:0] + D22xy) >> 2
Bn [7:3] ≤ (Bx[7:0] + D42xy) >> 3

Where:
(Rn, Gn, Bn) = Output RGB-565 pixel
(Rx, Gx, Bx) = Input RGB-888 pixel
(D22xy, D42xy) = Input x, y coordinate dithering matrix result

$$D22 = \begin{bmatrix} 0 & 3 \\ 2 & 1 \end{bmatrix}, \quad D42 = \begin{bmatrix} 0 & 3 & 4 & 7 \\ 2 & 1 & 6 & 5 \end{bmatrix}$$

48.3.8 Internal Field Signal Generation

As the VIN controls capture of data in interlaced mode, correct capture control is not achieved if the external field signal level does not change.

Through the internal field signal generation function, the VIN can control the capture field signal even when the input field signal does not change, such as during progressive data capturing. The following settings can be made for the internal field generation function through the FTEV and FTEH bits in the video n data mode register 2 (VnDMR2).

- VSYNC field toggle mode (FTEV = 1 in VnDMR2)
  When this setting is made, the VSYNC field toggle mode is entered for capture field signal control if the input field signal does not change for the VSYNC cycles specified by the VLV bits in VnDMR2. The toggle mode is canceled when a change in the external field signal level is detected (the capture operation is controlled according to the input field signal).
- HSYNC field toggle counter (FTEH = 1 in VnDMR2)
  This counter counts the capture active lines. If the external field signal does not change until the count reaches the HLV setting in VnDMR2, the capture field signal is controlled.

Note 1. Do not set both the FTEV and FTEH bits in VnDMR2 at the same time.
Note 2. Immediately after cancellation of the toggle mode, capture control is skipped for one VSYNC cycle in some cases depending on the input field signal state.

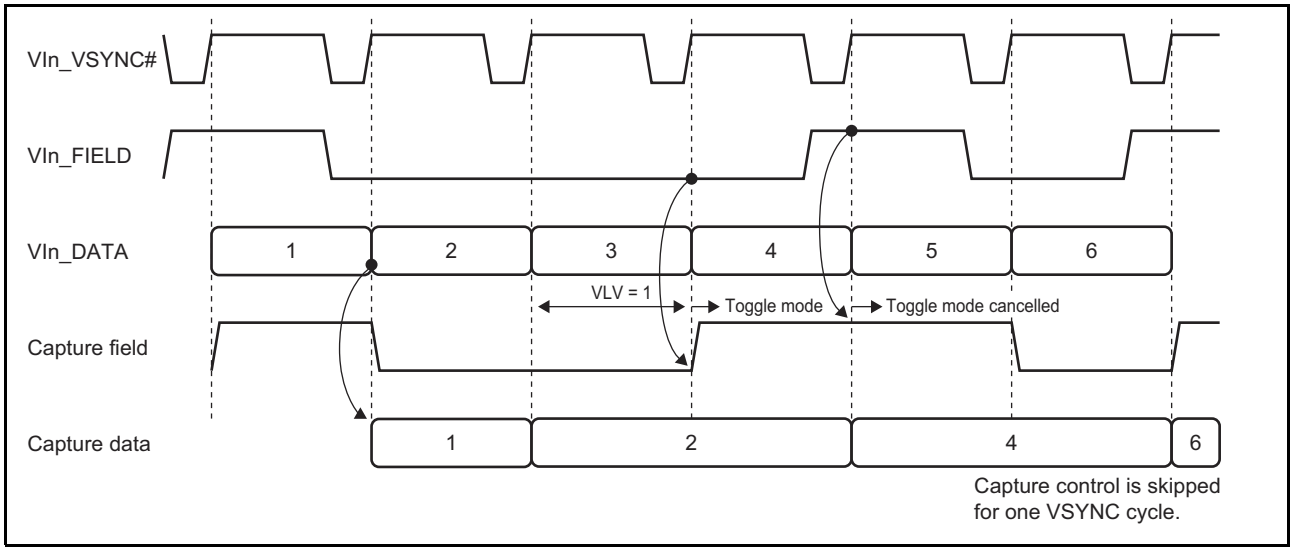


Figure 48.10 Overview and Notes of VSYNC Field Toggle Mode

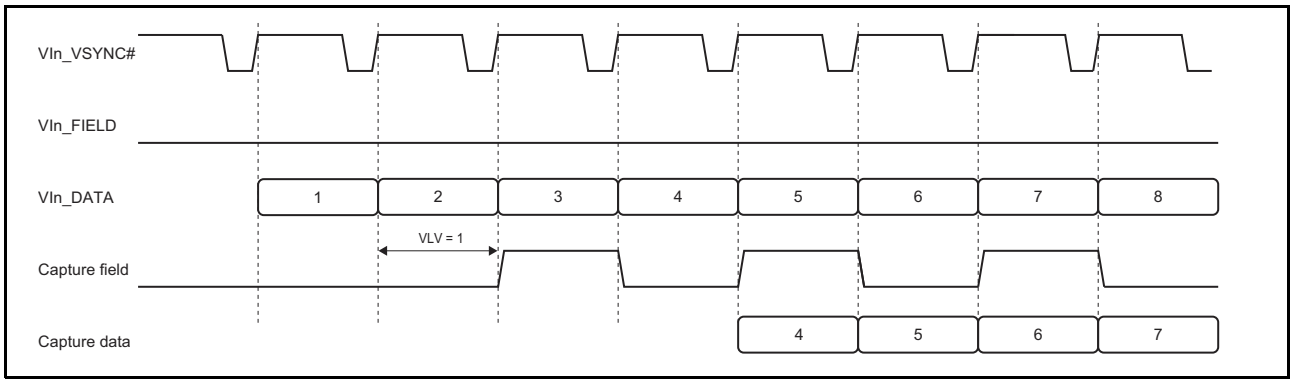


Figure 48.11 VSYNC Field Toggle Mode When the External Field Signal Does Not Change

48.3.9 Output Data Format

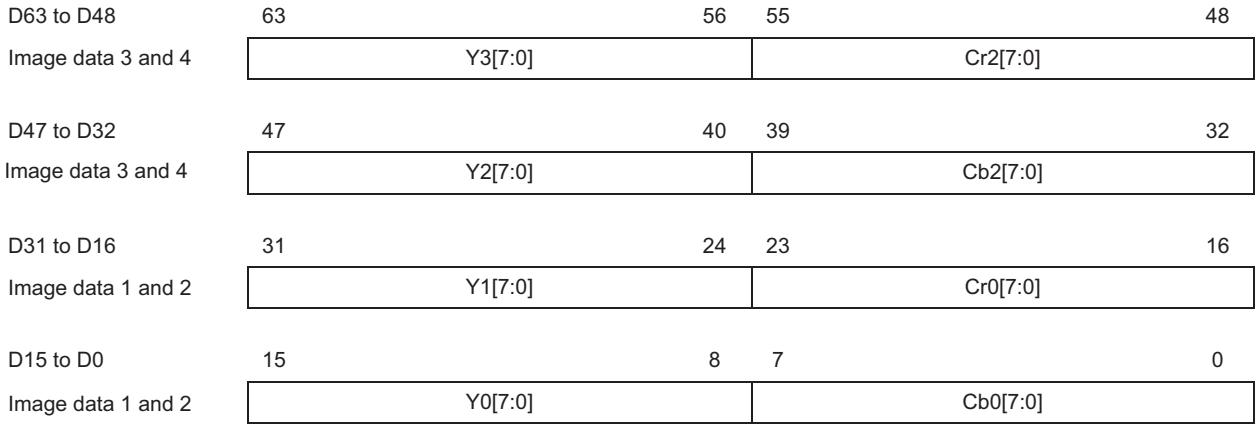
The VIN can output image data in the following formats. The figures in this section assume that data is stored in unified memory in little endian.

(1) YC: YCbCr-422, 8 bits

The 8-bit YUV image data in the YC (YCbCr) = 4:2:2 format is shown below. YC data can be switched between the UYVY format and YUYV format through the BPSM bit in VnDMR.

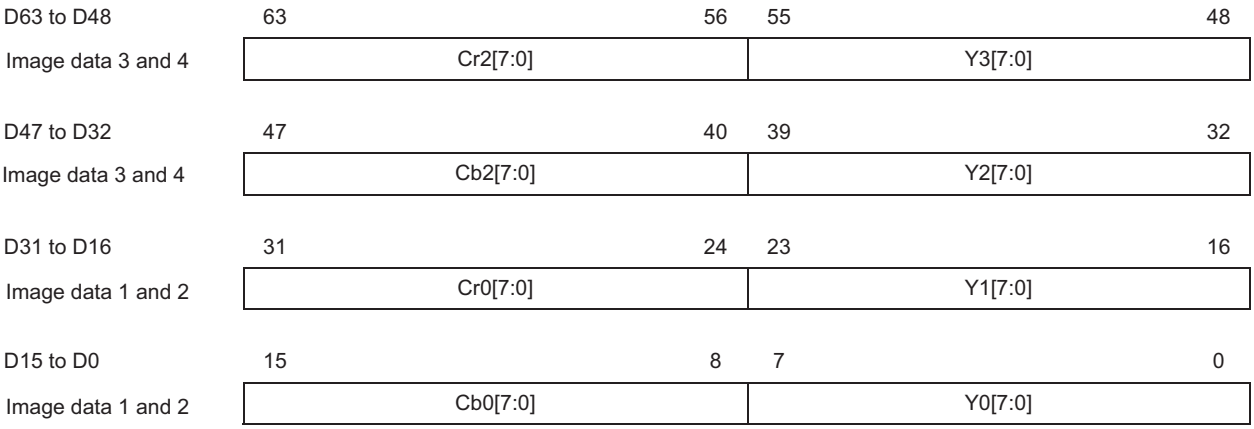
- BPSM = 0 in VnDMR: UYVY format

• 8-bit YCbCr-422 data (UYVY format)



- BPSM = 1 in VnDMR: YUYV format

• 8-bit YCbCr-422 data (YUYV format)



(2) YC: YCbCr-422, 10 bits

The 10-bit YUV image data in the YC (YCbCr) = 4:2:2 format is shown below.

Note 1. In this format, the YC_THR bit in VnDMR should be set to 1.
Note 2. Word addresses of YCbCr data are output in accordance with the VnIS register setting.

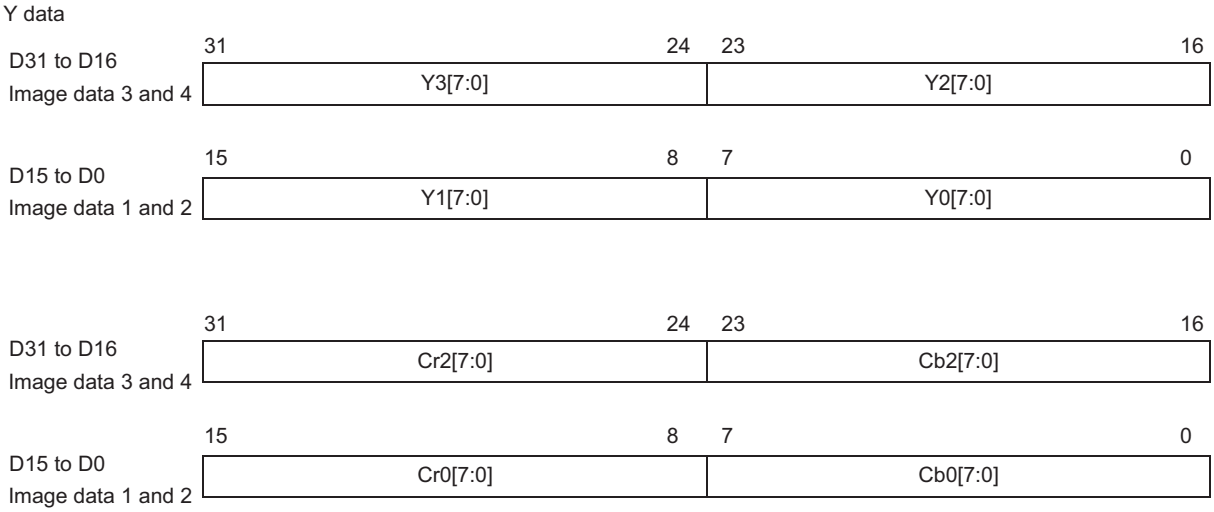
- 10-bit YCbCr-422 data: (UYVY format)

	127					122	121	112
D127 to D112 Image data 8	0	0	0	0	0	0	Y3[9:0]	
	111					106	105	96
D111 to D96 Image data 7	0	0	0	0	0	0	Cr2[9:0]	
	95					90	89	80
D95 to D80 Image data 6	0	0	0	0	0	0	Y2[9:0]	
	79					74	73	64
D79 to D64 Image data 5	0	0	0	0	0	0	Cb2[9:0]	
	63					58	57	48
D63 to D48 Image data 4	0	0	0	0	0	0	Y1[9:0]	
	47					42	41	32
D47 to D32 Image data 3	0	0	0	0	0	0	Cr0[9:0]	
	31					26	25	16
D31 to D16 Image data 2	0	0	0	0	0	0	Y0[9:0]	
	15					10	9	0
D15 to D0 Image data 1	0	0	0	0	0	0	Cb0[9:0]	

(3) YC: YC Separation YCbCr-422, Y (8 bits)/C (8 bits)

This is 8-bit YUV image data in the YC separated YC (YCbCr) = 4:2:2 format. UV data is supported by the NV16 format only.

If the YMODE[2:0] bits in the video n data mode register (VnDMR) are set to B'000 or B'001, setting the DTMD[1:0] bits in VnDMR to B'10 changes the format to 4:2:2. UV data is transferred to the address obtained by adding the value set in the video n UV address offset register (VnUVAOF) to the memory base address register. If the YMODE[2:0] bits in VnDMR are set to B'001, only Y data can be transferred and UV data cannot be transferred.



(4) YC: YC Separation YCbCr-422 Y (10 bits)/C (10 bits)

This is 10-bit YUV image data in the YC separated YC (YCbCr) = 4:2:2 format. UV data is supported by the NV16 format only.

If the YMODE[2:0] bits in the video n data mode register (VnDMR) are set to B'000 or B'001, setting the DTMD[1:0] bits in VnDMR to B'10 changes the format to 4:2:2. UV data is transferred to the address obtained by adding the value set in the video n UV address offset register (VnUVAOF) to the memory base address register. If the YMODE[2:0] bits in VnDMR are set to B'001, only Y data can be transferred and UV data cannot be transferred.

Note 1. In this format, the YC_THR bit in VnDMR should be set to 1.
Note 2. Word addresses of YCbCr data are output in accordance with the VnIS register setting.

Y data (10 bits)

	63					58	57	48
D63 to D48 Image data 4	0	0	0	0	0	0	Y3[9:0]	

	47					42	41	32
D47 to D32 Image data 3	0	0	0	0	0	0	Y2[9:0]	

	31					26	25	16
D31 to D16 Image data 2	0	0	0	0	0	0	Y1[9:0]	

	15					10	9	0
D15 to D0 Image data 1	0	0	0	0	0	0	Y0[9:0]	

Cb, Cr data (10 bits)

	63					58	57	48
D63 to D48 Image data 4	0	0	0	0	0	0	Cr2[9:0]	

	47					42	41	32
D47 to D32 Image data 3	0	0	0	0	0	0	Cb2[9:0]	

	31					26	25	16
D31 to D16 Image data 2	0	0	0	0	0	0	Cr0[9:0]	

	15					10	9	0
D15 to D0 Image data 1	0	0	0	0	0	0	Cb0[9:0]	

(5) YC: YC Separation YCbCr-422, Y (10 bits)/C (8 bits)

This is 10-bit YUV image data in the YC separated YC (YCbCr) = 4:2:2 format. UV data is supported by the NV16 format only.
If the setting of the YMODE[2:0] bits in the video n data mode register (VnDMR) is B'010 or B'011, setting the DTMD[1:0] bits in VnDMR to B'10 leads to the conversion of 10-bit CrCb data to 8-bit CrCb data and changing the format to 4:2:2.
UV data is transferred to the address obtained by adding the value set in the video n UV address offset register (VnUVAOF) to the memory base address register. If the YMODE[2:0] bits in VnDMR are set to B'011, only Y data can be transferred and UV data cannot be transferred.

Note 1. Word addresses of Y data are output in accordance with the VnIS register setting.
Note 2. Byte addresses of CbCr data are output in accordance with the VnIS register setting.

Y data

	63					58	57	48
D63 to D48 Image data 4	0	0	0	0	0	0	Y3[9:0]	
D47 to D32 Image data 3	47					42	41	32
	0	0	0	0	0	0	Y2[9:0]	
D31 to D16 Image data 2	31					26	25	16
	0	0	0	0	0	0	Y1[9:0]	
D15 to D0 Image data 1	15					10	9	0
	0	0	0	0	0	0	Y0[9:0]	

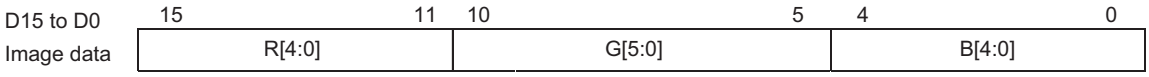
Cb, Cr data

	31	24	23	16
D31 to D16 Image data 3 and 4	Cr2[7:0]		Cb2[7:0]	
D15 to D0 Image data 1 and 2	15	8	7	0
	Cr0[7:0]		Cb0[7:0]	

(6) 16 Bits/Pixel: RGB-565

The RGB levels are expressed through 5 bits for R, 6 bits for G, and 5 bits for B.

16-bit/pixel data (RGB data) format

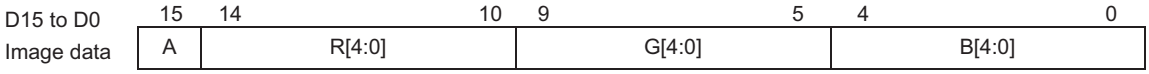


(7) 16 Bits/Pixel: ARGB-1555

The ARGB levels are expressed through 1 bit for A, 5 bits for R, 5 bits for G, and 5 bits for B. For data conversion to ARGB-1555, the lowest bit of the G data in RGB-565 data is truncated, and the A value specified through the register is added.

Set the DTMD[1:0] bits in the video n data mode register (VnDMR) to B'01 to specify conversion to ARGB-1555, and specify the A value in the ABIT bit in VnDMR.

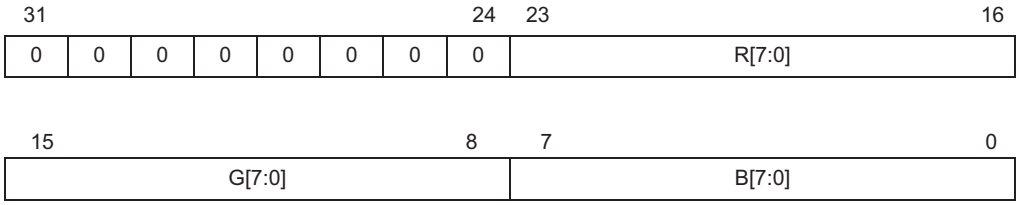
16-bit/pixel data (ARGB data) format



(8) 32 Bits/Pixel: RGB-888

The RGB levels are expressed through 8 bits for R, 8 bits for G, and 8 bits for B. Bits 31 to 24 are fixed to 0.

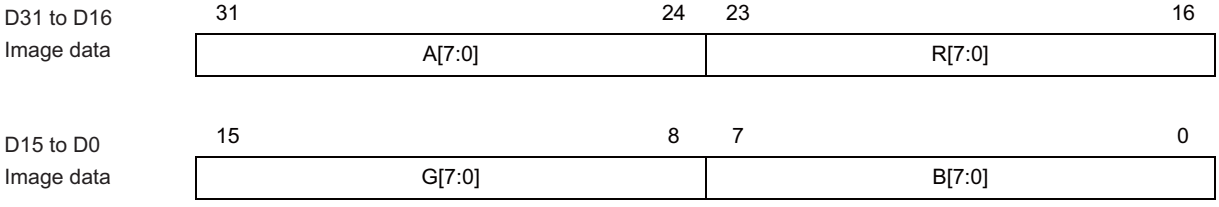
32-bit/pixel data (RGB data) format



(9) 32 Bits/Pixel: ARGB-8888

The ARGB levels are expressed through 8 bits for A, 8 bits for R, 8 bits for G, and 8 bits for B. The A value specified by the A8BIT[7:0] bits in VnDMR is set in bits 31 to 24.

32-bit/pixel data (ARGB data) format

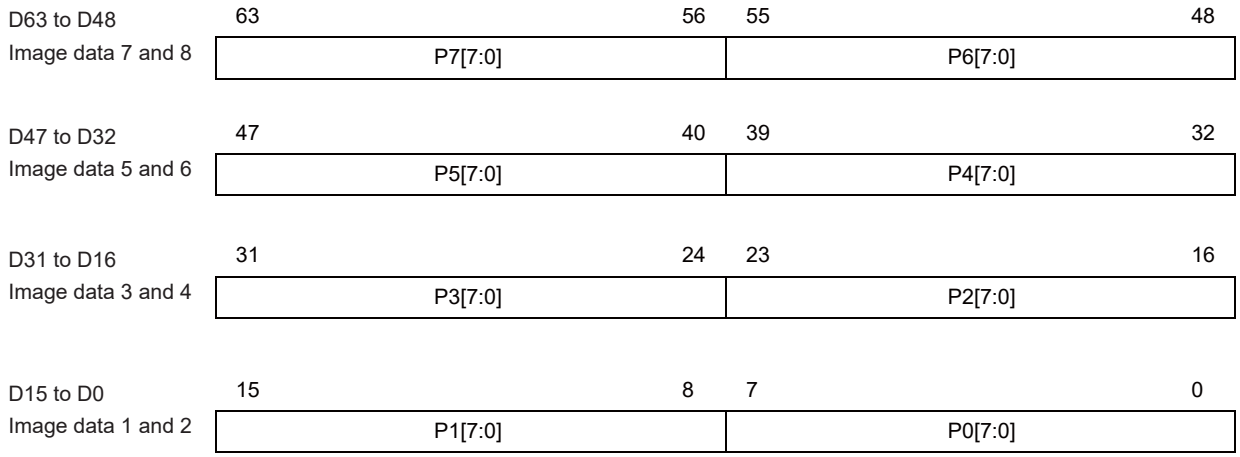


(10) Raw: 8 bits

The format of 8-bit raw image data is shown below.

Note 1. The data transfer is in 4-byte units.
Note 2. Byte addresses of raw data are output in accordance with the VnIS register setting.

8-bit raw data format



48.3.10 Endianess Conversion

The VIN stores captured data in memory in little endian with the initial settings. Set the EN bit in the video n main control register (VnMC) to 1 to convert data into big endian before storing in memory.
Endian conversion in word units is controlled through the EN bit in VnMC, and swapping in byte units is controlled through the BPSM bit in the video n data mode register (VnDMR). For conversion to big endian, specify the BPSM bit as shown in the following table according to the data format specified through the DTMD bits in VnDMR and BPS bit in VnMC.

Table 48.12 Endian Conversion Unit

Data Format	BPS in VnMC	DTMD[1:0] in VnDMR	EXRGB in VnDMR	BPSM in VnDMR	Endian Conversion Unit
RGB-565	0	00	0	0	Word units
RGB-888	0	00	1	0	Longword units
YCbCr-422	1	00	0	1	Byte units
ARGB-1555	0	01	0	0	Word units
ARGB-8888	0	01	1	0	Longword units
YC	1	10	0	0	Byte units

The following figures show endian conversions in byte, two-byte, and four-byte units.

Endian conversion in byte units:

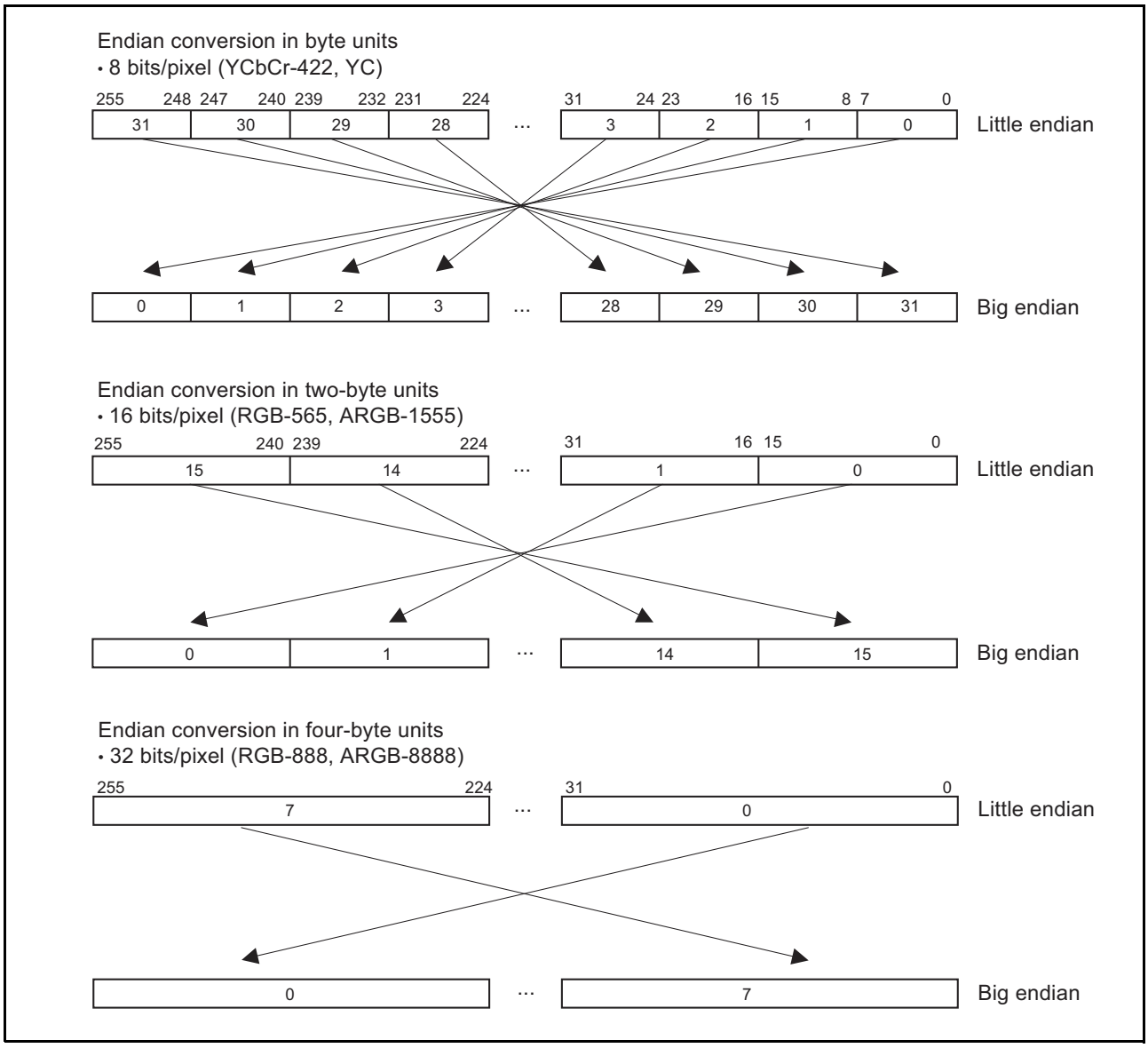


Figure 48.12 Data Alignment Conversion from Little Endian to Big Endian

48.3.11 Module Standby Mode

The LSI supports module standby mode in which clock supply to the VIN is stopped. The VIN should not be accessed during module standby mode. For details of the module standby function, refer to section 52, Power-Down Modes.

48.3.12 Transition to Module Standby Mode

1. Clear the module enable bit (ME) in the video n main control register (VnMC) and the continuous frame capture mode bit (CC) and the single frame capture mode bit (SC) in the video n frame capture register (VnFC) to 0 to stop the VIN.
2. Confirm that the capture active bit (CA) in the video n module status register (VnMS) is cleared to 0.*1
3. Set MSTP to 1, assigned to this module in the setting of section 52, Power-Down Modes.

Note 1. The camera device should be kept operation until the CA bit in VnMS set to 0.

48.3.13 Cancellation of Module Standby Mode and Restarting of VIN

1. Set MSTP to 0, assigned to this module in the setting of section 52, Power-Down Modes.
2. Set the module enable bit (ME) in the video n main control register (VnMC) to 1 to start the VIN.
3. Set the continuous frame capture mode bit (CC) or the single frame capture mode bit (SC) in the video n frame capture register (VnFC) to 1.

48.3.14 Interrupts

The VIN is capable of generating 1 interrupt.
For details, refer to section 7, Interrupt Controller.

Table 48.13 Interrupt assignment

Interrupt Name	Cause of interrupt
VIN.	FIS2 or VFS or VRS or FIS or SIS or EFS or FOS bit in V0INTS

48.3.15 Initialization Procedure

The VIN initialization procedure can be executed either before or after the CSI2 initialization procedure.

[VIN initialization procedure]

1. Set the VIN registers.
2. Set the VSYNC field toggle mode enable bit (FTEV) in the video n data mode register 2 (VnDMR2) to 1. The VSYNC field toggle mode transition period bits (VLV[3:0]) in the VnDMR2 shall be set to H'0.
3. Set the module enable bit (ME) in the video n main control register (VnMC) to 1 to start the VIN.
4. Set the continuous frame capture mode bit (CC) or the single frame capture mode bit (SC) in the video n frame capture register (VnFC) to 1.

[CSI2 initialization procedure]

1. Set the Link and PHY of the CSI-2 module registers*1.
2. Start the camera device.
3. Confirm that the PHY of the CSI-2 module starts.

Note 1. For details, see section 47, MIPI CSI2 Interface.

48.3.16 Capture Stop Procedure

1. Clear the module enable bit (ME) in the video n main control register (VnMC) and the continuous frame capture mode bit (CC) and the single frame capture mode bit (SC) in the video n frame capture register (VnFC) to 0 to stop the VIN.
2. Confirm that the video capture active status bit (CA) in the video n module status register (VnMS) is cleared to 0.*1
3. Set the MSTP bit to 1 in the module stop control register.
4. Set the SRST bit to 1 in the software reset register.
5. Reset the Link and PHY of the CSI-2 module.
6. Stop the camera device.

Note 1. The camera device should be kept operating until the CA bit in VnMS is set to 0.

48.3.17 Capture Restarting Procedure

The VIN initial procedure can be executed either before or after the CSI2 initial procedure.

[VIN restarting procedure]

1. Set the SRST bit to 0 in the software reset register.
2. Set the MSTP bit to 0 in the module stop control register.
3. Set the VIN registers.
4. Set the VSYNC field toggle mode enable bit (FTEV) in the video n data mode register 2 (VnDMR2) to 1. The VSYNC field toggle mode transition period bits (VLV[3:0]) in the VnDMR2 shall be set to H'0.
5. Set the module enable bit (ME) in the video n main control register (VnMC) to 1 to start the VIN.
6. Set the continuous frame capture mode bit (CC) or the single frame capture mode bit (SC) in the video n frame capture register (VnFC) to 1.

[CSI2 restarting procedure]

1. Set the Link and PHY of the CSI-2 module registers.
2. Start the camera device.
3. Confirm that the PHY of the CSI-2 module starts.

48.4 Usage Notes

48.4.1 Specifications

The following shows the specifications.

Table 48.14 Specifications

Item	Description
Input from MIPI CSI-2 interface	In the input of progressive images through the MIPI CSI-2 interface, use the internal field signal generation function. For detail, see section 48.2.18, Video n Data Mode Register 2 (VnDMR2).
Limitation on register update	If a register is updated during capture, data captured immediately after the register update cannot be guaranteed. A register that supports the internal update mode as shown in Table 48.2 is updated immediately after the register is written to. To update other registers, stop capture operation and then update them.
Field capture mode image quality	Images of odd-numbered, odd/even-numbered and even-numbered fields, captured by VnMC/IM interlace mode bit settings, contain every other line from the input interlace images. Therefore, note that the horizontal resolution for video display is in units of fields.
Limitation on horizontal scaling	When scaling-up is used, the horizontal size that can be input is up to 2048 pixels. When scaling-down is used, the horizontal size that can be output is up to 2048 pixels.
Interrupt event timing	An interrupt event asserted by this module indicates the time when the interrupt event occurs in VIN, not the time when the transfer of captured data to memory is completed.
Pixel post-clip setting	Even when the output format is YCbCr-422, pixel post-clipping is applied in accordance with the VnUDS_CLIP_SIZE setting. Therefore, specify an even value for the clipping size in the CL_HSIZE bits when the output format is YCbCr-422.
Coefficient settings for color space conversion	Specify appropriate values in the color space conversion coefficient 1 to 3 registers (VnCSCC1 to VnCSCC3) to keep the RGB image data within the range 0 < R', G', B' < 255. After calculation for color space conversion, pixel values less than 0 normalized to 0, and pixel values greater than or equal to 255 normalized to 255.
Scaling up	When horizontal and vertical scaling up is specified, the amount of memory transfer becomes greater with an increase in traffic. Note the amount of traffic on the entire system when using the scaling-up function because the overall transfer efficiency of the system might degrade due to the increased use of the internal buses.
YC Restrictions on YC separation function	Set the offset register (VnUVAOF) that stores UV data such that the storage areas of Y and UV data are not the same.
RGB-888 → RGB-565 conversion function	In the dithering process by cumulative addition, if the same color is captured as in a blue back image, periodic noise may be generated by the cumulative addition process (carried by addition). In this case, set the DC[1:0] bits in VnMC to the ordered dithering.
The maximum size that can be captured	The maximum number of lines and pixels that can be captured at this time is shown below. The maximum number of lines that can be captured = 2048 (the maximum value of the clipping register) – The vertical blanking time The maximum number of pixels that can be captured = 2048 (the maximum value of the clipping register) – The horizontal blanking time
Clipping size setting	The common pre-clipping size for odd and even fields is specified in VnSLPrC, VnELPrC, VnSPPrC, and VnEPPrC. The common post-clipping size for odd and even fields is specified in VnUDS_CLIP_SIZE.
Limitation on interlace input	When an interlace image is input, be sure to supply the field signal to VIN. When an interlace image is input from the MIPI CSI-2 interface, it is necessary to create the field signal in the CSI2 module. For the details of the field signal creation in CSI2, see section 47, MIPI CSI2 Interface.

Table 48.14 Specifications

Item	Description
UDS scaler setup sequence	[Initial setting ~ Capture start] [1] Set the SCLE bit in VnMC to 1. Set the other bits in VnMC to 0. [2] Set the UDS registers. [3] Set the VIN registers. [4] Set the ME bit in VnMC to 1. [5] Set the CC bit in VnFC to 1. [6] Capture starts. [Procedure to stop capturing and disable the UDS registers] [1] Set the ME bit in VnMC to 0. Keep the current values of the other bits in VnMC unchanged. [2] Set the CC bit in VnFC to 0. [3] Wait until the CA bit in VnMS is set to 0. (Maximum two vsync cycles) [4] Set the SCLE bit in VnMC to 0. [Resume capturing while the UDS is in use] [5] Change the values of the UDS control registers. [6] Set the SCLE bit in VnMC to 1. [7] Set the ME bit in VnMC to 1. [8] Set the CC bit in VnFC to 1. [9] Capture restarts. (After a maximum of two vsync cycles)
Horizontal clipping specification	Horizontal clipping size shall be set as follows.

Capture data	Pre-clipping start unit	Clipping size unit
YCbCr-422 data	2 pixels	2 pixels
RGB data	2 pixels	2 pixels
RAW data	4 pixels	4 pixels

48.4.2 Limitations on Usage

Table 48.15 Limitations on Usage

Item	Description
Frame buffers	Allocate the frame buffers in the on-chip RAM. To prevent performance to be degraded due to a page conflict when accessing the on-chip RAM, the maximum number of masters that simultaneously access the page where the frame buffers are allocated should be four, which includes the VIN.

49. SD/MMC Host Interface

Note: Development of the SD host--related products needs the conclusion of the following agreement. SD Host/ Ancillary Product License Agreement (SD HALA)

49.1 Overview

49.1.1 Features

- SD memory/IO card interface (1-bit/4-bit SD bus)
- SD, SDHC, and SDXC SD memory card access supported
- Default, high-speed, UHS-I/SDR50, DDR50, and SDR104 transfer modes supported
- SD clock (SD_CLK) frequency = Bφ frequency/2^n (n = 0 to 9)
- Error check function: CRC7 (for command/response), CRC16 (for data)
- Interrupt request: 1
- Card detect function
- Write protect support
- MMC interface (1-/4-/8-bit MMC bus)
  Note: Channel 1 only supports a 1- or 4-bit MMC bus.
- e-MMC device access supported
- Backward-compatible, high-speed, HS200 transfer modes supported
- High-priority interrupt (HPI) supported

Block Diagram

Figure 49.1 shows a block diagram of the SD/MMC host interfaces.

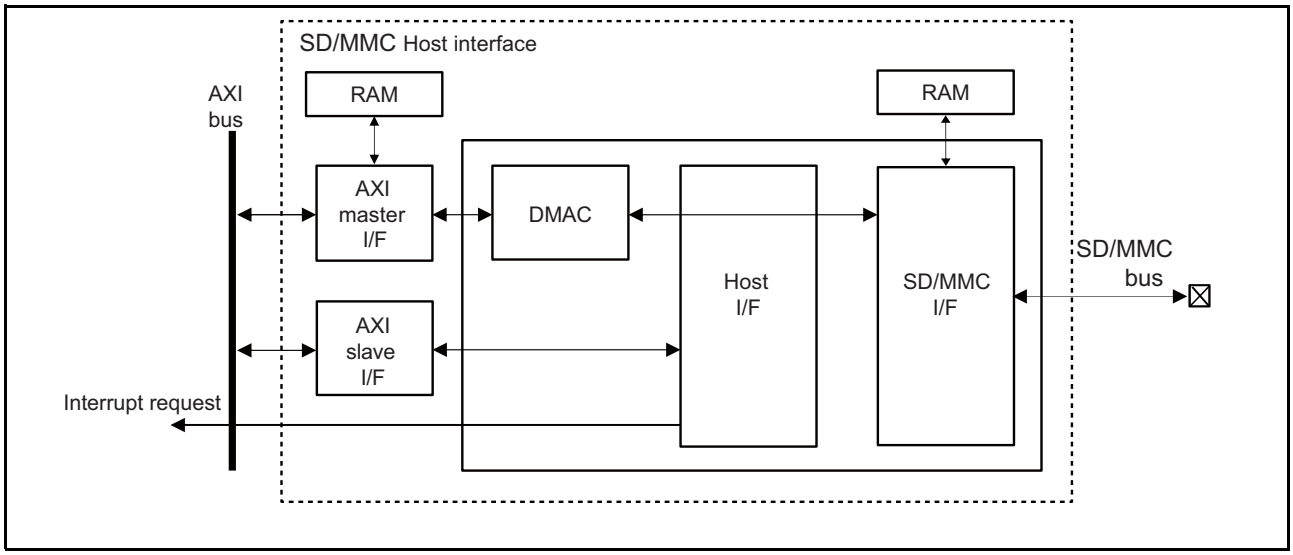


Figure 49.1 Block Diagram of SD/MMC Host Interface

49.1.2 External Pins

Table 49.1 lists the input and output pins used by the interface. The operating voltage on the pins of the SD/MMC host interface is 3.3 V or 1.8 V. Channels 0 and 1 can operate with different voltages. Before using the pins of the SD/MMC host interface, be sure to set the dedicated pin POC control register (PPOC) and SD/MMC host interface dedicated pin driving ability control registers 0 to 2 (PSDMMC0 to PSDMMC2).
For details, see the following sections:
section 51.3.30, Dedicated Pin POC Control Register (PPOC),
section 51.3.31, SD/MMC Host Interface Dedicated Pin Driving Ability Control Register 0 (PSDMMC0),
section 51.3.32, SD/MMC Host Interface Dedicated Pin Driving Ability Control Register 1 (PSDMMC1),
section 51.3.33, SD/MMC Host Interface Dedicated Pin Driving Ability Control Register 2 (PSDMMC2).

Table 49.1 Pin Configuration

PIN Name	I/O	Function
SDx_CLK*1	O	SD/MMC clock output
SDx_CMD*1	I/O	SD/MMC command output, response input
SDx_DAT0*1	I/O	SD/MMC Data 0 [bit 0]
SDx_DAT1*1	I/O	SD/MMC Data 1 [bit 1], SDIO interrupt
SDx_DAT2*1	I/O	SD/MMC Data 2 [bit 2], read wait
SDx_DAT3*1	I/O	SD/MMC Data 3 [bit 3], Card detection
SD0_DAT4*1	I/O	SD/MMC Data 4 [bit 4],
SD0_DAT5*1	I/O	SD/MMC Data 5 [bit 5]
SD0_DAT6*1	I/O	SD/MMC Data 5 [bit 6]
SD0_DAT7*1	I/O	SD/MMC Data 7 [bit 7]
SDx_CD*1	I	SD/MMC card detection*2
SDx_WP*1	I	SD/MMC write protection*2
SD0_RST#	O	SD/MMC reset

Note 1. x(= 0, 1) is the channel number of the SD/MMC host interface.
Note 2. Fix to 1 when not in use.

49.1.3 Register Configuration

The base addresses for each channel are as follows. The SD interface and MMC interface are switched by setting the command type register (SD_CMD).

Channel 0: H'E8228000
Channel 1: H'E822A000

Table 49.2 Register Configurations

Name	Abbreviation	Address	Access Width	Mirror
Command type register	SD_CMD	H'0000	16/32/64	
Command argument registers	SD_ARG	H'0010	16/32/64	
	SD_ARG1	H'0018	16/32/64	SD_ARG[31:16]
Data stop register	SD_STOP	H'0020	16/32/64	
Block count register	SD_SECCNT	H'0028	16/32/64	
Card response registers	SD_RSP10	H'0030	16/32/64	
	SD_RSP1	H'0038	16/32/64	SD_RSP10[31:16]
	SD_RSP32	H'0040	16/32/64	SD_RSP10[63:32]
	SD_RSP3	H'0048	16/32/64	SD_RSP32[31:16]
	SD_RSP54	H'0050	16/32/64	
	SD_RSP5	H'0058	16/32/64	SD_RSP54[31:16]
	SD_RSP76	H'0060	16/32/64	SD_RSP54[63:32]
	SD_RSP7	H'0068	16/32/64	SD_RSP76[31:16]
SD card interrupt flag register 1	SD_INFO1	H'0070	16/32/64	
SD card interrupt flag register 2	SD_INFO2	H'0078	16/32/64	
SD_INFO1 interrupt mask register	SD_INFO1_MASK	H'0080	16/32/64	
SD_INFO2 interrupt mask register	SD_INFO2_MASK	H'0088	16/32/64	
SD clock control register	SD_CLK_CTRL	H'0090	16/32/64	
Transfer data length register	SD_SIZE	H'0098	16/32/64	
Card access control option register	SD_OPTION	H'00A0	16/32/64	
SD error status register 1	SD_ERR_STS1	H'00B0	16/32/64	
SD error status register 2	SD_ERR_STS2	H'00B8	16/32/64	
SD buffer read/write register	SD_BUF0	H'00C0	16/32/64	
SDIO mode control register	SDIO_MODE	H'00D0	16/32/64	
SDIO interrupt flag register	SDIO_INFO1	H'00D8	16/32/64	
SDIO_INFO1 interrupt mask register	SDIO_INFO1_MASK	H'00E0	16/32/64	
DMA mode enable register	CC_EXT_MODE	H'0360	16/32/64	
Software reset register	SOFT_RST	H'0380	16/32/64	
Version register	VERSION	H'0388	16/32/64	
Host interface mode setting register	HOST_MODE	H'0390	16/32/64	
SD interface mode setting register	SDIF_MODE	H'0398	16/32/64	
SD status register*1	SD_STATUS	H'03C8	16/32/64	
DMAC mode register	DM_CM_DTRAN_MODE	H'0820	16/32/64	
DMAC control register	DM_CM_DTRAN_CTRL	H'0828	16/32/64	
DMAC software reset register	DM_CM_RST	H'0830	16/32/64	
DMAC interrupt register 1	DM_CM_INFO1	H'0840	16/32/64	
DM_CM_INFO1 interrupt mask register	DM_CM_INFO1_MASK	H'0848	16/32/64	
DMAC interrupt register 2	DM_CM_INFO2	H'0850	16/32/64	

Table 49.2 Register Configurations

Name	Abbreviation	Address	Access Width	Mirror
DM_CM_INFO2 interrupt mask register	DM_CM_INFO2_MASK	H'0858	16/32/64	
DMAC address register	DM_DTRAN_ADDR	H'0880	16/32/64	
SCC register area*2	—	—	—	—

Note 1. Only for channel 0.
Note 2. Refer to section 49.7, SCC Register Descriptions.

49.2 Register Description

49.2.1 Command Type Register (SD_CMD)

The command type register (SD_CMD) is used to select the command type and response type. The command sequence is started by writing to SD_CMD.
For details on the SD_CMD setting, refer to section 49.4.14, Example of SD_CMD Register Setting.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 14	MD7 MD6	00	R/W	Multiple Block Transfer Mode (enabled at multiple block transfer) 00: CMD12 is automatically issued at multiple block transfer. 01: CMD12 is not automatically issued at multiple block transfer. 10: Setting prohibited 11: Setting prohibited
13	MD5	0	R/W	Single/Multiple Block Transfer (enabled when the command with data is handled) 0: Single block transfer 1: Multi block transfer
12	MD4	0	R/W	Write/Read Mode (enabled when the command with data is handled) 0: Write (SD/MMC host interfaces -> SD card) 1: Read (SD/MMC host interfaces <- SD card)
11	MD3	0	R/W	Data Mode (Command Type) 0: Command without data transfer (bc, bcr, ac) 1: Command with data transfer (adtc)
10 9 8	MD2 MD1 MD0	000	R/W	Mode/Response Type 000: Normal mode The response type and the transfer mode are selected by SD_CMD[7:0], and the SD_CMD[15:11] setting is disabled. 001: Setting prohibited 010: Setting prohibited 011: Extended mode/No response 100: Extended mode/R1, R5, R6, or R7 response from the SD card 101: Extended mode/R1b response from the SD card 110: Extended mode/R2 response from the SD card 111: Extended mode/R3 or R4 response from the SD card Some commands cannot be used in normal mode. For details, see section 49.4.14, Example of SD_CMD Register Setting to select mode/response type.
7 6	C1 C0	00	R/W	00: CMD 01: ACMD 10: Setting prohibited 11: Setting prohibited
5 4 3 2 1 0	CF45 CF44 CF43 CF42 CF41 CF40	000000	R/W	Command Index These bits specify Command Format[45:40] (command index). [Examples] CMD6: SD_CMD[7:0] = B'0000 0110 CMD18: SD_CMD[7:0] = B'0001 0010 ACMD13: SD_CMD[7:0] = B'0100 1101

Note: SD_CMD cannot be written to when the CBSY bit in SD_INFO2 is 1.

49.2.2 Command Argument Register (SD_ARG)

Command arguments for SD cards are set in the SD command argument registers (SD_ARG). Set the command arguments before writing to SD_CMD.
Note that the argument of CMD12 within command sequences is H'0000 0000 regardless of the setting of SD_ARG.

SD_ARG

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CF39 to CF8	All 0	R/W	Set command format[39:8] (argument).

SD_ARG1 (Mirror of SD_ARG[31:16])

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Fixed 0
15 to 0	CF39 to CF24	All 0	R/W	Set command format[39:24] (argument).

49.2.3 Data Stop Register (SD_STOP)

The data stop register (SD_STOP) is used to enable or disable block counting at multiple block transfer, and to control the issuing of CMD12 within command sequences.

Bit	Bit Name	Initial Value	R/W	Description
63 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
18	—	0*1	R/W	Reserved The write value should always be 0.
17	HPIMODE	0*1	R/W	HPI Mode Enable 0: Disables HPI mode. 1: Enables HPI mode.
16	HPICMD	0*1	R/W*3	HPI Command Issue When HPICMD is set to 1 while HPIMODE is 1, the HPI command (CMD12) is issued. This bit is cleared to 0 when reception of the response to CMD12 is completed. The timing with which this bit is set to 1 is as follows. • After reception of the response to CMD12 that was issued by setting the STP bit to 1 has been completed during the CMD6/CMD38 or CMD25 sequence. • After reception of the response to CMD24/CMD25 has been completed After HPICMD is set to 1, do not write 0 to this bit while the CBSY bit in SD_INFO2 is 1. Do not set this bit to 1 when the CBSY bit in SD_INFO2 is 0.
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8	SEC	0*1	R/W	Block Count Enable*2 0: Disables SD_SECCNT setting value. 1: Enables SD_SECCNT setting value. Set SEC to 1 at multiple block transfer. When SD_CMD is set as follows to start the command sequence while SEC is set to 1, CMD12 is automatically issued to stop multi-block transfer with the number of blocks which is set to SD_SECCNT. 1.CMD18 or CMD25 in normal mode (SD_CMD[10:8] = 000) 2.SD_CMD[15:13] = 001 in extended mode (CMD12 is automatically issued, multiple block transfer) When the command sequence is halted because of a communications error or timeout, CMD12 is not automatically issued.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	STP	0*1	R/W	Stop • When STP is set to 1 during multiple block transfer, CMD12 is issued to halt the transfer through the SD/MMC host interface. However, if a command sequence is halted because of a communications error or timeout, CMD12 is not issued. Although continued buffer access is possible even after STP has been set to 1, the buffer access error bit (ERR5 or ERR4) in SD_INFO2 will be set accordingly. • When STP has been set to 1 during transfer for single block write, the access end flag is set when SD_BUF becomes empty, and CMD12 is not issued. If SD_BUF does contain data, the access end flag is set on completion of reception of the busy state without CMD12 having been issued. • When STP has been set to 1 during transfer for single block read, the access end flag is set immediately after setting of the STP bit and CMD12 is not issued. • When STP is set to 1 during reception of the busy state after an R1b response, the access end flag is set on completion of reception of the busy state without CMD12 having been issued. • When STP is set to 1 after a command sequence has been completed, CMD12 is not issued and the access end flag is not set. • Set STP to 1 after the response end flag has been set. • Set STP to 0 after the response end flag has been set.

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.
Note 2. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.
Note 3. It is effective only if 1 is written.

49.2.4 Block Count Register (SD_SECCNT)

The block count register (SD_SECCNT) is used to specify the number of transfer blocks at multiple block transfer.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 0	CNT31 to CNT0	All 0	R/W	Number of Transfer Blocks* When H'0000 0001 is set, the number of transfer blocks is 1. : When H'0000 FFFF is set, the number of transfer blocks is 65535. : When H'FFFF FFFF is set, the number of transfer blocks is 4294967295. Do not set this register to H'0000 0000 if multiple blocks are to be transferred.

Note: * Do not change the value of these bits when the CBSY bit in SD_INFO2 is set to 1.

49.2.5 SD Card Response Registers (SD_RSP)

The SD card response registers (SD_RSP) hold the response from the SD card.

SD_RSP10

Bit	Bit Name	Initial Value	R/W	Description
63 to 0	R71 to R8	All 0	R	Hold the response from the SD card

SD_RSP1 (Mirror of SD_RSP10[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	R39 to R24	All 0	R	Hold the response from the SD card

SD_RSP32 (Mirror of SD_RSP10[63:32])

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 0	R71 to R40	All 0	R	Hold the response from the SD card

SD_RSP3 (Mirror of SD_RSP32[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	R71 to R56	All 0	R	Hold the response from the SD card

SD_RSP54

Bit	Bit Name	Initial Value	R/W	Description
63 to 56	—	All 0	R	These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
55 to 0	R127 to R72	All 0	R	Hold the response from the SD card

SD_RSP5 (Mirror of SD_RSP54[31:16])

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	R103 to R88	All 0	R	Hold the response from the SD card

SD_RSP76 (Mirror of SD_RSP54[63:32])

Bit	Bit Name	Initial Value	R/W	Description
63 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
23 to 0	R127 to R104	All 0	R	Hold the response from the SD card

SD_RSP7 (Mirror of SD_RSP76[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7 to 0	R127 to R120	All 0	R	Hold the response from the SD card

Table 49.3 lists the response types and corresponding SD_RSP registers.

Table 49.3 Response Types and Corresponding SD_RSP Registers

Response Types	SD_RSP Registers
R1, R1b[39:8]	SD_RSP10 SD_RSP54*
R2[127:8]	SD_RSP54 and SD_RSP10
R3[39:8]	SD_RSP10
R4[39:8]	SD_RSP10
R5[39:8]	SD_RSP10
R6[39:8]	SD_RSP10
R7[39:8]	SD_RSP10

Note: The response to CMD18 and to CMD25 is stored in both R[39:8] and R[103:72].
This makes it possible to confirm the response to CMD18 and CMD25 by reading R[103:72] even if the response to automatic CMD12 is stored in R[39:8].

49.2.6 SD Card Interrupt Flag Register (SD_INFO1)

The SD card interrupt flag register 1 (SD_INFO1) indicates the response end and access end in the command sequence. This register also indicates the card detect/write protect state.
For CMD12 and CMD52 (SDIO abort) at multiple block transfer, INFO0 is not set but only INFO2 is set.
Even if the command sequence is halted because of a communications error or timeout, INFO0 or INFO2 is set.
INFO10, INFO9, and INFO8 change depending on the SDDAT3 state after a reset is released and continue to change in 4-bit transfer mode.
To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	HPIRES	0*2	R/W*1	Response Reception Completion [Setting condition] When reception of the response to CMD12 that was issued by setting the STP bit to 1 is completed during the CMD6/CMD38 or CMD25 sequence in HPI mode. [Clearing condition] When 0 is written to HPIRES
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
10	INFO10	Unknown	R	Indicates the SDDAT3 state. 0: SDDAT3 is set to 0. 1: SDDAT3 is set to 1.
9	INFO9	0	R/W*1	SDDAT3 Card Insertion [Setting condition] After change in SDDAT3 from 0 to 1, two cycles of 2 Bφ have elapsed with SDDAT3 held 1. [Clearing condition] When 0 is written to INFO9
8	INFO8	0	R/W*1	SDDAT3 Card Removal [Setting condition] After change in SDDAT3 from 1 to 0, two cycles of 2 Bφ have elapsed with SDDAT3 held 0. [Clearing condition] When 0 is written to INFO8
7	INFO7	Unknown	R	Write Protect Indicates the ISDWP state. 0: ISDWP is set to 1. 1: ISDWP is set to 0.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
5	INFO5	Unknown	R	Indicates the ISDCD state. 0: Indicates that Mcycle has elapsed with ISDCD held 1. 1: Indicates that Mcycle has elapsed with ISDCD held 0. Mcycle is set by bits 3 to 0 in SD_OPTION.
4	INFO4	0	R/W*1	ISDCD Card Insertion [Setting condition] After change in ISDCD from 1 to 0, Mcycle has elapsed with ISDCD held 0. [Clearing condition] When 0 is written to INFO4 Mcycle is set by bits 3 to 0 in SD_OPTION.
3	INFO3	0	R/W*1	ISDCD Card Removal [Setting condition] After change in ISDCD from 0 to 1, Mcycle has elapsed with ISDCD held 1. [Clearing condition] When 0 is written to INFO3 Mcycle is set by bits 3 to 0 in SD_OPTION.

Bit	Bit Name	Initial Value	R/W	Description
2	INFO2	0*2	R/W*1	Access End [Setting conditions] 1. When read access to the buffer is completed in the case of transfer for single block read 2. When read access to the buffer for the last block of data is completed in the case of transfer for multiple block read 3. When read access to the buffer and reception of the response to CMD12 are completed in the case of transfer for multiple block read with automatic issuing of CMD12 4. When reception of the busy state after reception of the CRC status is completed in the case of transfer for single block write 5. When reception of the busy state after reception of the CRC status of the last block of data is completed in the case of transfer for multiple block write 6. When reception of the response busy state for CMD12 is completed in the case of transfer for multiple block write with automatic issuing of CMD12 7. When reception of the response to CMD12 that was issued by setting the STP bit to1 is completed in the case of transfer for multiple block read 8. When reception of the response busy state for CMD12 that was issued by setting the STP bit to1 is completed in the case of transfer for multiple block write 9. When reception of the response to CMD52 that was issued by setting the IOABT bit to1 is completed in the case of transfer for multiple block read 10. When reception of the response to CMD52 that was issued by setting the IOABT bit to1 is completed in the case of transfer for multiple block write In addition to the above conditions, this bit is set when a command sequence is halted because of a communications error or timeout. [Clearing condition] When 0 is written to INFO2 When the access end bit is set to 1, the command sequence is terminated.
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
0	INFO0	0*2	R/W*1	Response End [Setting conditions] 1. When the reception of the response is completed 2. When the transmission of the command not requiring a response is completed, 3. When receiving busy reception after R1b response 4. When reception of the response to CMD52 that was issued by setting the C52PUB bit to1 is completed in the case of transfer for multiple block read 5. When reception of the response to CMD52 that was issued by setting the C52PUB bit to1 is completed in the case of transfer for multiple block write, In addition to the above conditions, this bit is set when a command sequence is halted because of a communications error or timeout. [Clearing condition] When 0 is written to INFO0 When issuing a command without data, the command sequence ends when the response end is set to 1.

Note 1. It is effective only if 0 is written.
Note 2. The value is initialized by a reset and also in the case of a reset by the SDRST bit in SOFT_RST.

49.2.7 SD Card Interrupt Flag Register (SD_INFO2)

The SD card interrupt flag register 2 (SD_INFO2) indicates the access status of the SD buffer (SD_BUF) and SD card. To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	ILA	0*2	R/W*1	Illegal Access Error [Setting conditions] 1. Writing of data to SD_CMD within a command sequence (CBSY=1) 2. When SD_CMD[11] = 1 (command with data transfer) and SD_CMD[7:0]= B'0000 1100 (CMD12) are set in SD_CMD [Clearing condition] When 0 is written to ILA
14	CBSY	0*2	R	Command Type Register Busy 0: A command sequence has been completed. 1: A command sequence is being executed.
13	SCLKDIVEN	1*2	R	0: The SD bus (CMD, DAT) is busy. Do not attempt to write to the SD_CLK_CTRL register. 1: The SD bus (CMD, DAT) is not busy. When a command sequence is started by writing to SD_CMD, the CBSY bit is set to 1 and, at the same time, the SCLKDIVEN bit is set to 0. The SCLKDIVEN bit is set to 1 after 8 cycles of SDCLK have elapsed after setting of the CBSY bit to 0 due to completion of the command sequence.
12	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
11	—	0	R/W	Reserved The write value should always be 1.
10	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
9	BWE	0*2	R/W*1	SD_BUF Write Enable 0: Data cannot be written in SD_BUF0. 1: Data can be written in SD_BUF0. [Setting conditions] 1. When SD_BUF is empty at single block transfer 2. When either bank 1 or bank 2 of SD_BUF is empty at multiple block transfer [Clearing conditions] 1. When 0 is written to BWE 2. Writing of a block of data to SD_BUF by DMA transfer When data is written to SD_BUF0 by the CPU, clear BWE and then write amount of data specified by SD_SIZE*3
8	BRE	0*2	R/W*1	SD_BUF Read Enable 0: Data cannot be read from SD_BUF0. 1: Data can be read from SD_BUF0. [Setting conditions] 1. When data set in SD_SIZE is stored in SD_BUF at single block transfer 2. When data set in SD_SIZE is stored in either bank 1 or bank 2 of SD_BUF at multiple block transfer [Clearing conditions] 1. When 0 is written to BRE 2. Reading of a block of data from SD_BUF by DMA transfer When data is read from SD_BUF0 by the CPU, clear BRE and then read amount of data specified by SD_SIZE*3. Even if a CRC error or an END error occurs while block data is read, data is stored in SD_BUF and BRE is set.

Bit	Bit Name	Initial Value	R/W	Description
7	DAT0	Unknown	R	SDDAT0 Indicates the SDDAT0 state. 0: SDDAT0 is set to 0. 1: SDDAT0 is set to 1. If the data timeout (ERR3) is set but the response timeout (ERR6) is not set after the Erase command has been issued, the end of the Erase sequence (DAT0 = 1) is confirmed by polling DAT0. If a communications error or timeout occurs during a write sequence, the DAT0 bit may retain the value 0. While the SD clock (SDCLK) is stopped, the DAT0 bit retains the value before the clock is stopped.
6	ERR6	0*2	R/W*1	Response Timeout [Setting condition] When a response is not received even after 640 cycles of SDCLK have elapsed (including a response to a command issued within a command sequence*5) [Clearing condition] When 0 is written to ERR6 The command sequence is halted by a response timeout.*4
5	ERR5	0*2	R/W*1	SD_BUF Illegal Read Access [Setting conditions] 1. When SD_BUF is empty while SD_BUF0 is read 2. When data with a CRC error or END error is read from SD_BUF0 [Clearing condition] When 0 is written to ERR5
4	ERR4	0*2	R/W*1	SD_BUF Illegal Write Access [Setting conditions] 1. When data is written to SD_BUF0 while it is not in the data read/write command state 2. When data is written to SD_BUF0 while SD_BUF is full 3. When data is written to SD_BUF0 while an error occurs in the CRC status or CRC status length 4. When data is written to SD_BUF0 while the interface remains in a busy state for at least Ncycle after the CRC status [Clearing condition] When 0 is written to ERR4 Ncycle is set by bits 7 to 4 in SD_OPTION.
3	ERR3	0*2	R/W*1	Data Timeout (except response timeout) [Setting conditions] 1. After R1b response, the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle. 2. After CRC status, the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle. 3. After write data, the CRC status is not received even after Ncycle has elapsed. 4. After read command, read data is not received even after Ncycle has elapsed. 5. After CMD12 has been issued within a command sequence, the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle. 6. After the reception of read data, read data for the next block are not received even after Ncycle has elapsed. 7. After release of the read wait state, read data for the next block are not received even after Ncycle has elapsed. [Clearing condition] When 0 is written to ERR3 Ncycle is set by bits 7 to 4 in SD_OPTION. The command sequence is halted by the data timeout.

Bit	Bit Name	Initial Value	R/W	Description
2	ERR2	0*2	R/W*1	END Error [Setting conditions] 1. When an error occurs in the response length (and the end bit has not been detected) 2. When an error occurs in the read data length (and the end bit has not been detected among the valid bits) 3. When an error occurs in the CRC status length (and the end bit has not been detected) 4. An error in the length of a response to a command issued within a command sequence*5 (i.e. the end bit has not been detected) [Clearing condition] When 0 is written to ERR2 The command sequence is halted by the End error.*4
1	ERR1	0*2	R/W*1	CRC Error [Setting conditions] 1. When an error occurs in the CRC status (i.e. the received CRC status was not 010') 2. When a CRC error occurs in the read data 3. When a CRC error occurs in the response 4. A CRC error in the response to a command issued within a command sequence*5 [Clearing condition] When 0 is written to ERR1 The command sequence is halted by the CRC error.*4
0	ERR0	0*2	R/W*1	CMD Error [Setting conditions] 1. The command index of the transmitted command differing from the command index of the received response 2. The command index of a command issued within a command sequence*5 differing from the command index of the received response [Clearing condition] When 0 is written to ERR0 The command sequence is halted by the CMD error.*4

Note 1. It is effective only if 0 is written.
Note 2. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.
Note 3. When the WMODE bit in HOST_MODE is 0, the single byte from the fraction of a full 16-bit unit is regarded as excess data due to an odd value for the number of bytes setting in SD_SIZE. When the WMODE bit in HOST_MODE is 1, the single byte or three bytes from the fraction of a full 64-bit unit are regarded as excess data due to an odd value for the number of bytes setting in SD_SIZE, or the two bytes from the fraction of a full 64-bit unit are regarded as excess data due if the value for the number of bytes setting in SD_SIZE is even but is not on a four-byte boundary.
Note 4. After the C52PUB bit in SDIO_MODE has been set to 1, if a communications error or timeout for response occurs in response to the CMD52 that is issued, since the command sequence has not been completed, complete the sequence with error processing as in usage examples in Figure 49.18 under section 49, IO_RW_EXTENDED (CMD53/Multiple Block Read) or in Figure 49.21 under section 49.4.9, IO_RW_EXTENDED (CMD53/Multiple Block Write).
Note 5. "Command issued within a command sequence" refers to CMD12 when automatic issuing is enabled for multiple block transfer by the setting in SD_CMD, CMD12 when the STP bit in SD_STOP is set to 1, or CMD52 when the C52PUB or IOABT bit in SDIO_MODE is set to 1.

49.2.8 SD_INFO1 Interrupt Mask Register (SD_INFO1_MASK)

The SD_INFO1 interrupt mask register (SD_INFO1_MASK) is used to enable or disable the SD_INFO1 interrupt. When 0 is set in SD_INFO1_MASK while the corresponding flag in SD_INFO1 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	IMASK16	1	R/W	HPIRES interrupt masked
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9	IMASK9	1	R/W	INFO9 interrupt masked
8	IMASK8	1	R/W	INFO8 interrupt masked
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
4	IMASK4	1	R/W	INFO4 interrupt masked
3	IMASK3	1	R/W	INFO3 interrupt masked
2	IMASK2	1	R/W	INFO2 interrupt masked
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
0	IMASK0	1	R/W	INFO0 interrupt masked

49.2.9 SD_INFO2 Interrupt Mask Register (SD_INFO2_MASK)

The SD_INFO2 interrupt mask register (SD_INFO2_MASK) is used to enable or disable the SD_INFO2 interrupt. When 0 is set in SD_INFO2_MASK while the corresponding flag in SD_INFO2 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	IMASK	1	R/W	ILA interrupt masked
14 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
11	—	1	R/W	Reserved The write value should always be 1.
10	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
9	BMASK1	1	R/W	BWE interrupt masked
8	BMASK0	1	R/W	BRE interrupt masked
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
6	EMASK6	1	R/W	ERR6 interrupt masked
5	EMASK5	1	R/W	ERR5 interrupt masked
4	EMASK4	1	R/W	ERR4 interrupt masked
3	EMASK3	1	R/W	ERR3 interrupt masked
2	EMASK2	1	R/W	ERR2 interrupt masked
1	EMASK1	1	R/W	ERR1 interrupt masked
0	EMASK0	1	R/W	ERR0 interrupt masked

49.2.10 SD Clock Control Register (SD_CLK_CTRL)

The SD clock control register (SD_CLK_CTRL) is used to control the SD clock (SDCLK) output and to set the frequency. Set SCLKEN to 1 before writing to SD_CMD to issue a command. Do not write to SD_CLK_CTRL while the SCLKDIVEN bit in SD_INFO2 is set to 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	—	0	R/W	Reserved The write value should always be 0.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
10	—	0*1	R/W	Reserved The write value should always be 0.
9	SDCLKOFFEN	0	R/W	SD Clock (SDCLK) Output Automatic Control Enable 0: Automatic control for SD clock (SDCLK) output is disabled. 1: Automatic control for SD clock (SDCLK) output is enabled. This function of automatic control for SD clock (SDCLK) output causes SDCLK output only within a command sequence. The timing with which SDCLK output starts and stops is as follows. SDCLK output starts after writing to SD_CMD. SDCLK output stops when 8 cycles of SDCLK have elapsed after the end of the command sequence. In addition, SDCLK is fixed to 0 while SCLKEN of SD_CLK_CTRL is 0, regardless of the value of this bit.
8	SCLKEN	0*1	R/W*2	SD Clock (SDCLK) Output Control Enable 0: SD clock (SDCLK) output is disabled. The SDCLK signal is fixed 0. 1: SD clock (SDCLK) output is enabled.
7	DIV7	0	R/W*2	SD Clock (SDCLK) B'1000 0000: Bφ/512 B'0100 0000: Bφ/256 B'0010 0000: Bφ/128 B'0001 0000: Bφ/64 B'0000 1000: Bφ/32 B'0000 0100: Bφ/16 B'0000 0010: Bφ/8 B'0000 0001: Bφ/4 B'0000 0000: Bφ/2 B'1111 1111: Bφ Other settings are prohibited. In addition, in the case of data transfer in DDR mode (DDR bit in SDIF_MODE = 1), do not set DIV[7:0] to B'1111 1111.
6	DIV6	0	R/W*2	
5	DIV5	1	R/W*2	
4	DIV4	0	R/W*2	
3	DIV3	0	R/W*2	
2	DIV2	0	R/W*2	
1	DIV1	0	R/W*2	
0	DIV0	0	R/W*2	

Note 1. This initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.
Note 2. Writing to SD_CLK_CTRL is impossible when the CBSY bit in SD_INFO2 is 1.

Notes on when setting the SD clock (SDCLK) to Bφ(DIV[7:0] = B'1111 1111)

When changing the setting of bits DIV[7:0] to B'1111 1111, or from B'1111 1111 to other setting, perform the following processing before writing to SD_CMD.

(1) Set the SCLKEN bit to 0 by writing to SD_CLK_CTRL. (Do not change the setting of bits other than SCLKEN at this time.)
(2) Change the setting of bits DIV[7:0] by writing to SD_CLK_CTRL. (Do not change the setting of bits other than DIV[7:0] at this time. The SCLKEN bit should retain the value 0.)
(3) Set the SCLKEN bit to 1 by writing to SD_CLK_CTRL. (Do not change the setting of bits other than SCLKEN at this time.)

Also when changing the setting of bits DIV[7:0] to B'1111 1111 after having set the SDRST bit in SOFT_RST to 0 and then changed it to 1, perform this processing before writing to SD_CMD.

49.2.11 Transfer Data Length Register (SD_SIZE)

The transfer data length register (SD_SIZE) is used to specify the transfer data size.

Bit	Bit Name	Initial Value	R/W	Description
63 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
11, 10	—	All 0	R	Reserved
9 to 0	LEN9 to LEN0	1000000000	R/W	Transfer Data Size*1 These bits specify a size between 1 and 512 bytes for single block transfer. In cases of multiple block transfer with automatic issuing of CMD12 (CMD18 and CMD25), the only specifiable transfer data size is 512 bytes. Furthermore, in cases of multiple block transfer without automatic issuing of CMD12, as well as 512 bytes, 32, 64, 128, and 256 bytes are specifiable. However, in the reading of 32, 64, 128, and 256 bytes for the transfer of multiple blocks, this is restricted to multiple block transfer by CMD53. Additionally, if a command accompanies data transfer, do not set these bits to 0. Do not specify a data size larger than 512 bytes.

Note: * Do not change the values of these bits when the CBSY bit in SD_INFO2 is 1.

49.2.12 SD Card Access Control Option Register (SD_OPTION)

The SD card access control option register (SD_OPTION) is used to set the bus width and timeout counter.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	WIDTH	0*1	R/W	Bus width*2 {WIDTH,WIDTH8} = 01: 8 bits width {WIDTH,WIDTH8} = 00: 4 bits width {WIDTH,WIDTH8} = 10,11: 1 bit width In the case of data transfer in DDR mode (DDR bit in SDIF_MODE = 1), do not set this bit to 1. In the case of writing of one-byte block, 8 bits width cannot be specified for the bus width. Change the bus width to 4 bits or 1 bit before writing one-byte block.
14	—	1	R	Reserved This bit is always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to this bit.
13	WIDTH8	0*1	R/W	Bus width*2 See the description of the WIDTH bit.
12 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9	EXTOP	0*1	R/W	Timeout Mode Select 0: Bits TOP27 to TOP24 specify the timeout count from SDCLK*2^{13} to SDCLK*2^{27}. 1: Bits TOP27 to TOP24 specify the timeout count from SDCLK*2^{14} to SDCLK*2^{28}.
8	TOUTMASK	0*1	R/W	Timeout Mask 0: Enables timeout. 1: Disables timeout. (The ERR6 and ERR3 bits in SD_INFO2 and the E6 to E0 bits in SD_ERR_STS2 are not set.) If a timeout occurs while it is disabled, perform a software reset to terminate a command sequence.
7	TOP27	1*1	R/W	Timeout Counter*2 0000: SDCLK*2^{13} 0001: SDCLK*2^{14} : 1101: SDCLK*2^{26} 1110: SDCLK*2^{27} 1111: Setting prohibited
6	TOP26	1*1	R/W	
5	TOP25	1*1	R/W	
4	TOP24	0*1	R/W	
3	CTOP24	1*1	R/W	Card Detect Time Counter 0000: SDCLK*2^{10} 0001: SDCLK*2^{11} : 1101: SDCLK*2^{23} 1110: SDCLK*2^{24} 1111: Setting prohibited
2	CTOP23	1*1	R/W	
1	CTOP22	1*1	R/W	
0	CTOP21	0*1	R/W	

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.
Note 2. Do not change the values of these bits when the CBSY bit in SD_INFO2 is 1.

49.2.13 SD Error Status Register 1 (SD_ERR_STS1)

The SD error status register 1 (SD_ERR_STS1) indicates the CRC status, CRC error, End error, and CMD error.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 15	—	Unknown	R	Reserved These bits are always read as unknown. The write value should always be 0.
14	E14	0*1	R	These bits hold the CRC status. (normal: 010)
13	E13	1*1		
12	E12	0*1		
11	E11	0*1	R	Set to 1 when an error occurs in the CRC status.
10	E10	0*1	R	Set to 1 when a CRC error occurs in the read data.
9	E9	0*1	R	Set to 1 when a CRC error occurs in the response to a command issued within a command sequence*2. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E8.
8	E8	0*1	R	Set to 1 when a CRC error occurs in a response (other than a response to a command issued within a command sequence*2).
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5	E5	0*1	R	Set to 1 when an error occurs in the CRC status length (and the end bit has not been detected).
4	E4	0*1	R	Set to 1 when an error occurs in the read data length (and the end bit has not been detected among the valid bits).
3	E3	0*1	R	Set to 1 when an error occurs in the response length to a command issued within a command sequence*2. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E2.
2	E2	0*1	R	Set to 1 when an error occurs in the response length (other than a response to a command issued within a command sequence*2).
1	E1	0*1	R	Set to 1 when an error occurs in the command index of the response to a command issued within a command sequence*2. In cases where CMD12 is issued by setting a command index in SD_CMD, this is Indicated in E0.
0	E0	0*1	R	Set to 1 when an error occurs in the command index of a response (other than a response to a command issued within a command sequence*2).

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.
Note 2. "Command issued within a command sequence" refers to CMD12 when automatic issuing is enabled for multiple block transfer by the setting in SD_CMD, CMD12 when the STP bit in SD_STOP is set to 1, or CMD52 when the C52PUB or IOABT bit in SDIO_MODE is set to 1.

49.2.14 SD Error Status Register 2 (SD_ERR_STS2)

The SD error status register 2 (SD_ERR_STS2) indicates the timeout state. Ncycle is set by bits 7 to 4 in SD_OPTION.

Bit	Bit Name	Initial Value	R/W	Description
63 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
6	E6	0*1	R	Set to 1 when the interface remains in a busy state for at least Ncycle after the CRC status.
5	E5	0*1	R	Set to 1 when the CRC status is not received even after Ncycle has elapsed after data writing.
4	E4	0*1	R	Set to 1 when read data is not received even after Ncycle has elapsed after the command has been read. Set to 1 when read data for the next block are not received even after Ncycle has elapsed after the reception of read data. Set to 1 when read data for the next block are not received even after Ncycle has elapsed after release of the read wait state.
3	E3	0*1	R	Set to 1 when the interface remains in a busy state for at least Ncycle after CMD12 has been issued within a command sequence. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E2.
2	E2	0*1	R	Set to 1 when the interface remains in a busy state for at least Ncycle after R1b response.
1	E1	0*1	R	Set to 1 when the response to a command issued within a command sequence*2 is not received even after 640 cycles of SDCLK have elapsed. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E0.
0	E0	0*1	R	Set to 1 when the response (other than a response to a command issued within a command sequence*2) is not received even after 640 cycles of SDCLK have elapsed.

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.
Note 2. "Command issued within a command sequence" refers to CMD12 when automatic issuing is enabled for multiple block transfer by the setting in SD_CMD, CMD12 when the STP bit in SD_STOP is set to 1, or CMD52 when the C52PUB or IOABT bit in SDIO_MODE is set to 1.

49.2.15 SD Buffer Read/Write Register (SD_BUF0)

Bit	Bit Name	Initial Value	R/W	Description
63 to 0	BUF63 to BUF0	Unknown	R/W	When writing to the SD card, the write data is written to this register. When reading from the SD card, the read data is read from this register. This register is internally connected to two 512-byte buffers (SD_BUF). When both buffers are not empty at multiple block read, suspend data reception by stopping the SD clock. When either buffer becomes empty, restart data reception by starting supply of the SD clock.

Note: When using the DMAC, the bus width should be fixed at 64 bits.

49.2.16 SDIO Mode Control Register (SDIO_MODE)

The SDIO mode control register (SDIO_MODE) controls the CMD52 issuance and the read wait state at multiple block transfer, and the reception of SDIO interrupt. C52PUB and IOABT should not be set to 1 simultaneously.

Bit	Bit Name	Initial Value	R/W	Description
63 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9	C52PUB	0	R/W	SDIO None Abort • When C52PUB is set to 1 in the CMD53 (multiple block) write sequence, CMD52 is automatically issued between blocks if SD_BUF becomes empty.C52PUB is automatically cleared to 0 after reception of the response to CMD52 is completed. Additionally, if C52PUB is set to 1 while the last block is being transferred, CMD52 is not issued. In this case, C52PUB is automatically cleared to 0 after the access end flag has been set to 1. • When C52PUB and RWREQ are set to 1 in the CMD53 (multiple block) read sequence, the block transfer enters the read wait state between blocks and CMD52 is automatically issued. C52PUB is automatically cleared to 0 after reception of the response to CMD52 is completed. Additionally, if C52PUB is set to 1 while the last block is being transferred, CMD52 is not issued. In this case, C52PUB is automatically cleared to 0 after the access end flag has been set to 1. • If C52PUB is set to 1 in the CMD53 (multiple block) read sequence, be sure to set RWREQ to 1 as well as C52PUB. • Set SD_ARG before setting C52PUB to 1. • Set C52PUB to 1 after the response end flag has been set.
8	IOABT	0	R/W	SDIO Abort • When IOABT is set to 1 in the CMD53 (multiple block) sequence, the CMD53 sequence is halted and CMD52 is issued. However, if a command sequence is halted because of a communications error or timeout, CMD52 is not issued. Although continued buffer access is possible even after IOABT has been set to 1, the buffer access error bit (ERR5 or ERR4) in SD_INFO2 will be set accordingly. Set SD_ARG before setting IOABT to 1. • When IOABT has been set to 1 during transfer for single block write, the access end flag is set when SD_BUF becomes empty, and CMD52 is not issued. If SD_BUF does contain data, the access end flag is set on completion of reception of the busy state without CMD52 having been issued. • When IOABT has been set to 1 during transfer for single block read, the access end flag is set immediately after setting of IOABT and CMD52 is not issued. • When IOABT is set to 1 during reception of the busy state after an R1b response, the access end flag is set on completion of reception of the busy state without CMD52 having been issued. • When IOABT is set to 1 after a command sequence has been completed, CMD52 is not issued and the access end flag is not set. • Set IOABT to 1 after the response end flag has been set. Set IOABT to 0 after the access end flag has been set.
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2	RWREQ	0	R/W	Read Wait Request When RWREQ is set to 1 in the CMD53 (multiple block) read sequence, the block transfer enters the read wait state between blocks. [Read wait state releasing] (1) The read wait state is released, when RWREQ is cleared to 0 in the read wait state. (2) When IOABT is set to 1 in the read wait state, RWREQ is automatically cleared to 0 after CMD52 has been issued, and then the read wait state is released. (3) When C52PUB and RWREQ are set to 1 simultaneously in the CMD53 (multiple block) read sequence, the read wait state is not automatically released. Therefore, after the CMD52 response is received, clear RWREQ. (Be sure to set RWREQ and C52PUB simultaneously.) When RWREQ is set to 1 while the last block in the CMD53 (multiple block) read sequence is transferred, the read wait state is not entered and RWREQ is automatically cleared to 0 by setting access end. Set RWREQ to 1 after the response end flag has been set.

Bit	Bit Name	Initial Value	R/W	Description
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
0	IOMOD	0	R/W	SDIO Mode*1 0: Disables the SD/MMC host interfaces to receive SDIO interrupt from the SDIO card 1: Enables the SD/MMC host interfaces to receive SDIO interrupt from the SDIO card

Note: Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.

49.2.17 SDIO Interrupt Flag Register (SDIO_INFO1)

The SDIO interrupt flag register (SDIO_INFO1) indicates the status regarding to the SDIO card access. To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	EXWT	0*2	R/W*1	[Setting condition] While the last block in the CMD53 (multiple block) read sequence is transferred, RWREQ in SDIO_MODE is set to 1. [Clearing condition] When 0 is written to EXWT
14	EXPUB52	0*2	R/W*1	[Setting conditions] 1. While the last block in the CMD53 (multiple block) sequence is transferred, C52PUB in SDIO_MODE is set to 1. 2. While C52PUB is set to 1 in the CMD53 (multiple block) write sequence, the last block is transferred. [Clearing condition] When 0 is written to EXPUB52
13 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2, 1	—	All 0*2	R/W	Reserved The write value should always be 1. The value may change during operation.
0	IOIRQ	0*2	R/W*1	[Setting condition] When SDIO interrupt from an SDIO card is received while IOMOD in SDIO_MODE is set to 1. [Clearing condition] When 0 is written to IOIRQ*3

Note 1. It is effective only if 0 is written.
Note 2. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.
Note 3. Before clearing this bit, access the SDIO card to negate the SDIO interrupt signal from the SDIO card. If the interrupt signal is not negated, this bit may be set again.

49.2.18 SDIO_INFO1 Interrupt Mask Register (SDIO_INFO1_MASK)

The SDIO_INFO1 interrupt mask register (SDIO_INFO1_MASK) enables or disables the SD_INFO1 interrupt. When 0 is set in SDIO_INFO1_MASK while the corresponding flag in SD_INFO1 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	MEXWT	1	R/W	EXWT interrupt masked
14	MEXPUB52	1	R/W	EXPUB52 interrupt masked
13 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2	—	1	R/W	Reserved The write value should always be 1.
1	—	1	R/W	Reserved The write value should always be 1.
0	IOMSK	1	R/W	IOIRQ interrupt masked

49.2.19 DMA Mode Enable Register (CC_EXT_MODE)

The DMA mode enable register (CC_EXT_MODE) enables the DMA transfer.

Bit	Bit Name	Initial Value	R/W	Description
63 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
12	—	1	R	Reserved This bit is always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to this bit.
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9, 8	—	All 0	R/W	Reserved The write value should always be 0.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5	—	0	R/W	Reserved The write value should always be 0.
4	—	1	R	Reserved This bit is always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to this bit.
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1	DMASDRW	0	R/W	SD_BUF Read/Write DMA Transfer* 0: The SD_BUF read/write DMA transfer is disabled. 1: The SD_BUF read/write DMA transfer is enabled.
0	—	0	R/W	Reserved The write value should always be 0.

Note: * Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.

49.2.20 Software Reset Register (SOFT_RST)

The software reset register (SOFT_RST) sets a software reset. Also use this register to check that release from the reset state has been completed before attempting to use the SD/MMC host interfaces and before attempting access to the other registers.

Bit	Bit Name	Initial Value	R/W	Description
63 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2, 1	—	11	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
0	SDRST	1	R/W	Software Reset of SD Interface Unit 0: Reset 1: Reset released

49.2.21 Version Register (VERSION)

The version register (VERSION) indicates the version of the SD/MMC host interfaces.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15, 14	UR7, UR6	11	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
13, 12	UR5, UR4	00	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
11 to 8	UR3 to UR0	H'C	R	Version of Renesas' IP
7 to 0	IP7 to IP0	H'10	R	Version of introductory IP

49.2.22 Host Interface Mode Setting Register (HOST_MODE)

The host interface mode setting register (HOST_MODE) selects the width for access to the data bus.

Bit	Bit Name	Initial Value	R/W	Description
63 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8	BUSWIDTH	0	R/W	Width for Access to SD_BUF*1*2 Read or write access to SD_BUF0 can be performed with the specified width for access. 0: 16-bit access 1: 32-bit access This bit is enabled while the WMODE bit is set to 1.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1	ENDIAN	0	R/W	SD_BUF0 data swap
0	WMODE	0	R/W	Width for Access to SD_BUF*1*2 Read or write access to SD_BUF0 can be performed with the specified width for access. 0: 64-bit access 1: 16-bit or 32-bit access

Note 1. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.
Note 2. When using the built-in DMAC of this module, fix the bus width to 64 bits.

49.2.23 SD Interface Mode Setting Register (SDIF_MODE)

The SD interface mode setting register specifies DDR mode.

Bit	Bit Name	Initial Value	R/W	Description
63 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9	—	0*1	R/W	Reserved The write value should always be 0.
8	NOCHKCR	0*1	R/W	CRC Check Mask (test command for MMC supported) Enables or disables checking of the CRC16 and CRC status. 0: Enables the CRC check. 1: Disables the CRC check (the CRC16 value is ignored at read, and the CRC status is not detected at write)
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	DDR	0*1	R/W	DDR Mode Select*2 0: Normal mode (default, high speed, or SDR) 1: DDR mode Set this bit to 0 when the SD clock division ratio is specified as 1:1 (bits DIV[7:0] in SD_CLK_CTRL are set to 11111111).

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.
Note 2. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.

49.2.24 SD Status Register (SD_STATUS)

The effective bit of the SD status register controls the output value on the SD0_RST# pin.

Bit	Bit Name	Initial Value	R/W	Description
63 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1	SD_RST	0	R/W	Controls the output value on the SD0_RST# pin. 0: The output value on the SD0_RST# pin is 0. 1: The output value on the SD0_RST# pin is 1.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.

49.2.25 DMAC Transfer Mode Register (DM_CM_DTRAN_MODE)

The DMAC Transfer Mode Register (DM_CM_DTRAN_MODE) sets the operation mode of the module built-in DMAC.
High 32 bits (bit 63 to 32) are read only 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
17, 16	CH_NUM	00	R/W	DMAC channel selector 00 : SD down stream 01 : SD up stream Other settings are prohibited.
15 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5, 4	BUS_WIDTH [1:0]	11	R/W	Bus width selector 11: 64-bit Other settings are prohibited.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.

49.2.26 DMAC Transfer Control Register (DM_CM_DTRAN_CTRL)

The DMAC Transfer Control Register (DM_CM_DTRAN_CTRL) controls the module built-in DMAC operation.
High 32 bits (bit 63-32) are read only 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8	—	0	R/W	Reserved The write value should always be 0.
7 to 1	—	All 0	R	Reserved
0	DM_START	0	R/W	DMAC Start Writing 1 to this bit starts DMAC operation. This bit is automatically cleared when DMA transfer is started.

49.2.27 DMAC Reset Register (DM_CM_RST)

High 32 bits (bit 63 to 32) are read only 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 10	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
9	DTRANRST1	1	R/W	Soft resets of the module built-in DMAC channel 1 0: Reset 1: Reset released
8	DTRANRST0	1	R/W	Soft resets of the module built-in DMAC channel 0 0: Reset 1: Reset released
7 to 1	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
0	SEQRST	1	R/W	Soft resets of the sequencer 0: Reset 1: Reset released

Note: When using the software reset by this register, see section 52.3.6, Software Reset and follow the procedures for using the STBREQ and STBACK registers.

49.2.28 DMAC Interrupt Register 1 (DM_CM_INFO1)

The DMAC interrupt register 1 (DM_CM_INFO1) indicates the status of the module built-in DMAC and a sequencer. To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
20	DTRANEND1	0*1	R/W	Module built-in DMAC Channel 1 Transfer End [Setting conditions] 1. When transfer of DMAC channel 1 is completed 2. When an error occurs on DMAC channel 1 [Clearing condition] When 0 is written to DTRANEND1
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	DTRANEND0	0*2	R/W	Module built-in DMAC Channel 0 Transfer End [Setting conditions] 1. When transfer of DMAC channel 0 is completed 2. When an error occurs on DMAC channel 0 [Clearing condition] When 0 is written to DTRANEND0
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	SEQEND	0*3	R/W	Sequencer Operation End [Setting conditions] 1. When operation of a sequencer is completed 2. When a sequencer error occurs [Clearing condition] When 0 is written to SEQEND

Note 1. The initial value is applied at a reset and when the DTRANRST1 bit in DM_CM_RST is 0.
Note 2. The initial value is applied at a reset and when the DTRANRST0 bit in DM_CM_RST is 0.
Note 3. The initial value is applied at a reset and when the SEQRST bit in DM_CM_RST is 0.

49.2.29 DM_CM_INFO1 Interrupt Mask Register (DM_CM_INFO1_MASK)

The DM_CM_INFO1 interrupt mask register (DM_CM_INFO1_MASK) enables or disables the DM_CM_INFO1 interrupt. When 0 is set in DM_CM_INFO1_MASK while the corresponding flag in DM_CM_INFO1 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 21	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
20	DTRANEND1_MASK	1	R/W	DTRANEND1 interrupt masked
19 to 17	—	All 1	R/W	Reserved The write value should always be 1.
16	DTRANEND0_MASK	1	R/W	DTRANEND0 interrupt masked
15 to 1	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
0	SEQEND_MASK	1	R/W	SEQEND interrupt masked

49.2.30 DMAC Interrupt Register 2 (DM_CM_INFO2)

The DMAC interrupt register 2 (DM_CM_INFO2) indicates the status of the module built-in DMAC and a sequencer. To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
19, 18	—	00	R	Reserved The write value should always be 0.
17	DTRANERR1	0*1	R/W	Module built-in DMAC Channel 1 Error [Setting condition] When an error occurs on DMAC channel 1 [Clearing condition] When 0 is written to DTRANERR1
16	DTRANERR0	0*2	R/W	Module built-in DMAC Channel 0 Error [Setting condition] When an error occurs on the DMAC channel 0 [Clearing condition] When 0 is written to DTRANERR0
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	SEQERR	0*3	R/W	Sequencer error [Setting condition] When a sequencer error occurs [Clearing condition] When 0 is written to SEQERR

Note 1. The initial value is applied at a reset and when the DTRANRST1 bit in DM_CM_RST is 0.
Note 2. The initial value is applied at a reset and when the DTRANRST0 bit in DM_CM_RST is 0.
Note 3. The initial value is applied at a reset and when the SEQRST bit in DM_CM_RST is 0.

49.2.31 DM_CM_INFO2 Interrupt Mask Register (DM_CM_INFO2_MASK)

The DM_CM_INFO2 interrupt mask register (DM_CM_INFO2_MASK) enables or disables the DM_CM_INFO2 interrupt. When 0 is set in DM_CM_INFO2_MASK while the corresponding flag in DM_CM_INFO2 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 20	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
19, 18	—	11	R	Reserved The write value should always be 1.
17	DTRANERR1_MASK	1	R/W	DTRANERR1 interrupt masked
16	DTRANERR0_MASK	1	R/W	DTRANERR0 interrupt masked
15 to 1	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
0	SEQERR_MASK	1	R/W	SEQERR interrupt masked

49.2.32 DMAC Transfer Address Register (DM_DTRAN_ADDR)

The DMAC Transfer Address Register (DM_DTRAN_ADDR) sets the transfer destination and source address of the module built-in DMAC.
Higher-order 32 bits (bit 63 to 32) are read only 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 3	DADDR	All 0	R/W	Destination address / Source address (8 byte unit) Note that the value of DM_DTRAN_ADDR + transfer data length is less than or equal to 2^{32}.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.

49.3 Operation

49.3.1 SD Interface

(1) SD Data Format

When data is read from the SD card, the procedure is as follows.

1. The SD/MMC host interface receives data from the SD card via the SDDAT signal. (SDDAT signal: see Figure 49.2, Figure 49.3 and Figure 49.5.)
2. The receive data is stored in SD_BUF of the SD/MMC host interfaces. (SD_BUF store data: see Figure 49.7)
3. The data stored in SD_BUF is read from SD_BUF0. (Reading from SD_BUF0: see Table 49.4)

When data is written to the SD card, the above procedure will be reversed.
When accessing SD_BUF0, caution should be taken for the transfer order in SDDAT and the store order in SD_BUF. In addition, data stored in SD_BUF0 can be replaced in bytes with the EXT_SWAP. (See Figure 49.7)

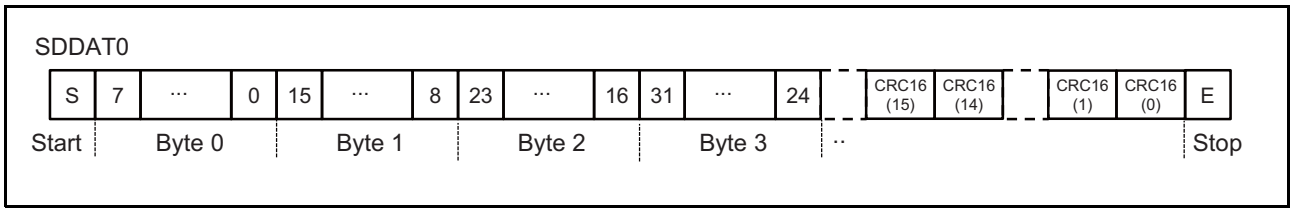


Figure 49.2 SDDAT in 1-Bit Width Mode

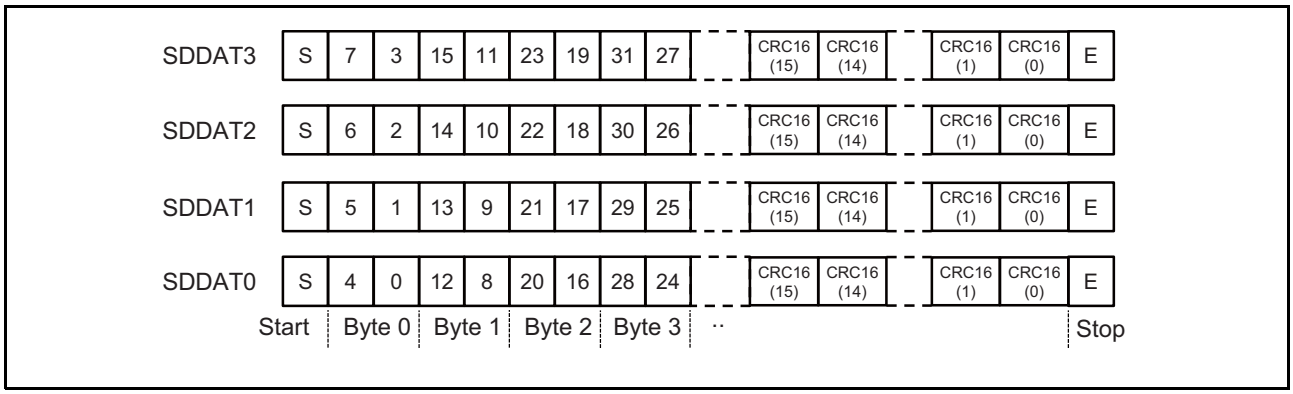


Figure 49.3 SDDAT in 4-Bit Width Mode

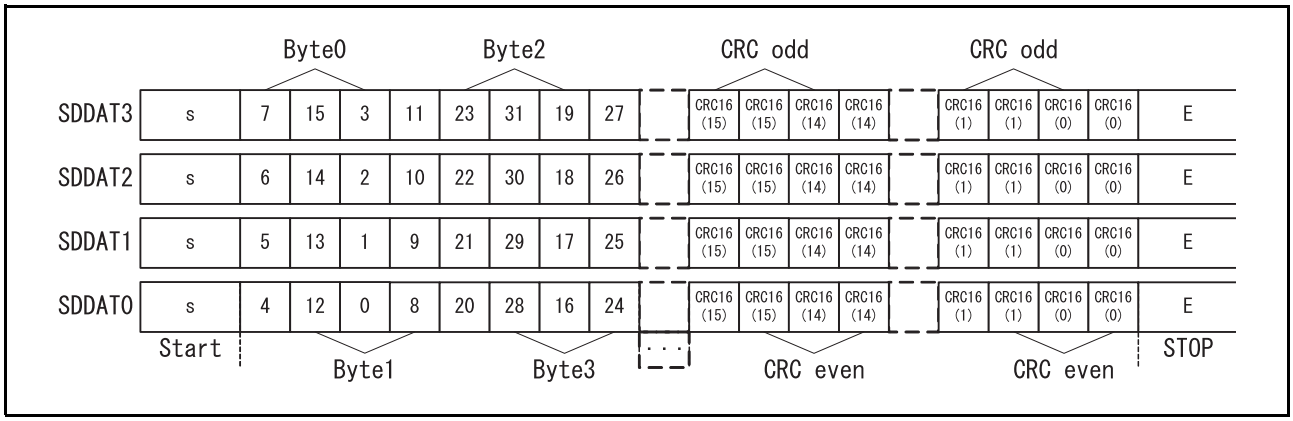


Figure 49.4 SDDAT in 4-Bit Width DDR Mode

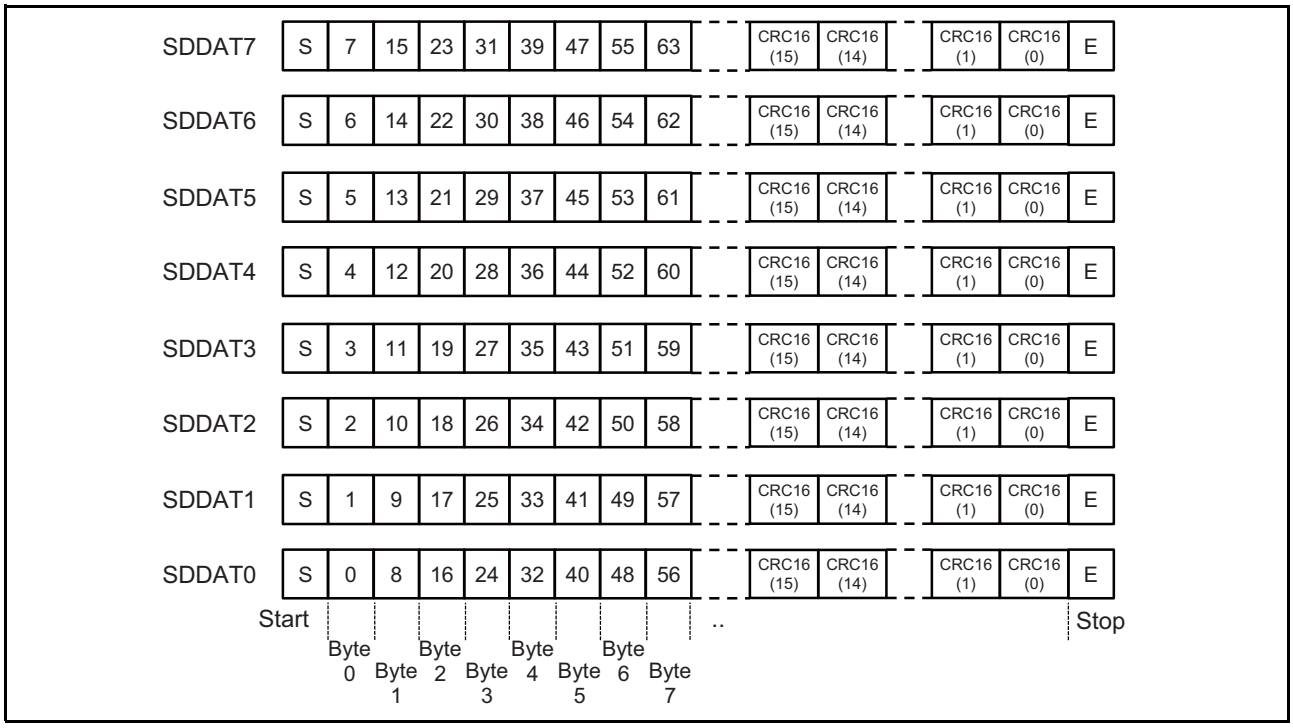


Figure 49.5 SDDAT in 8-Bit Width Mode

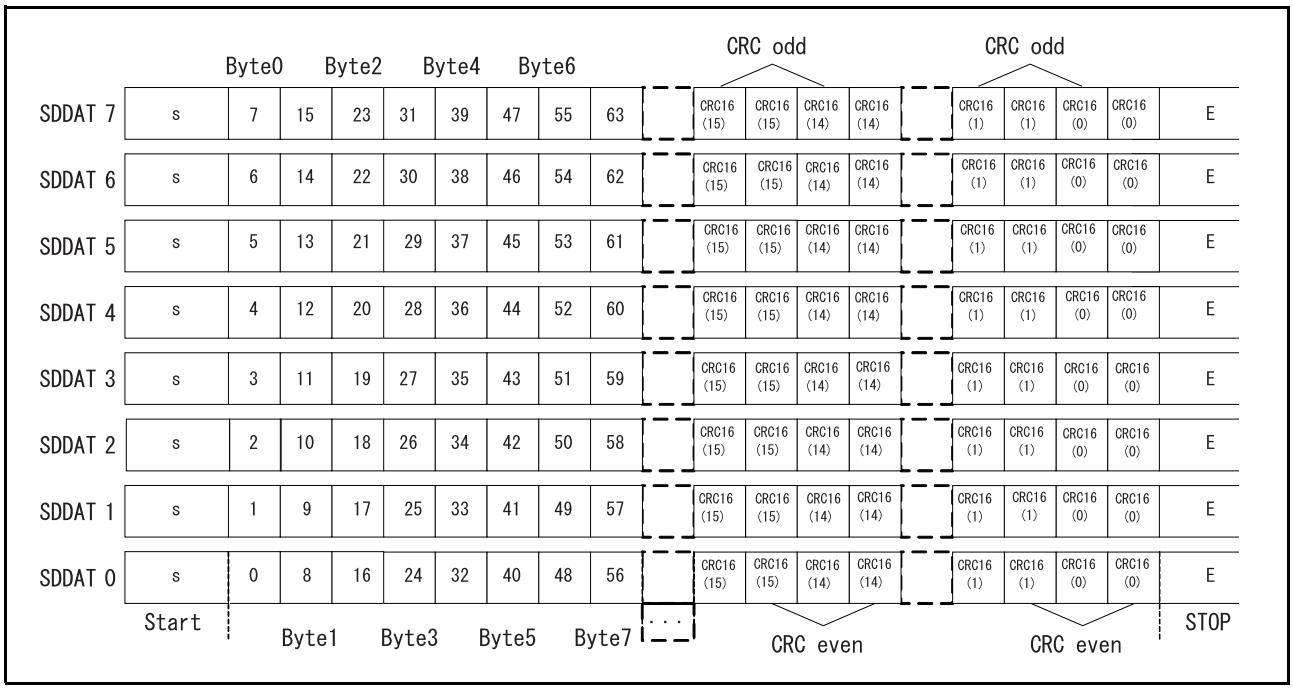


Figure 49.6 SDDAT in 8-Bit Width DDR Mode

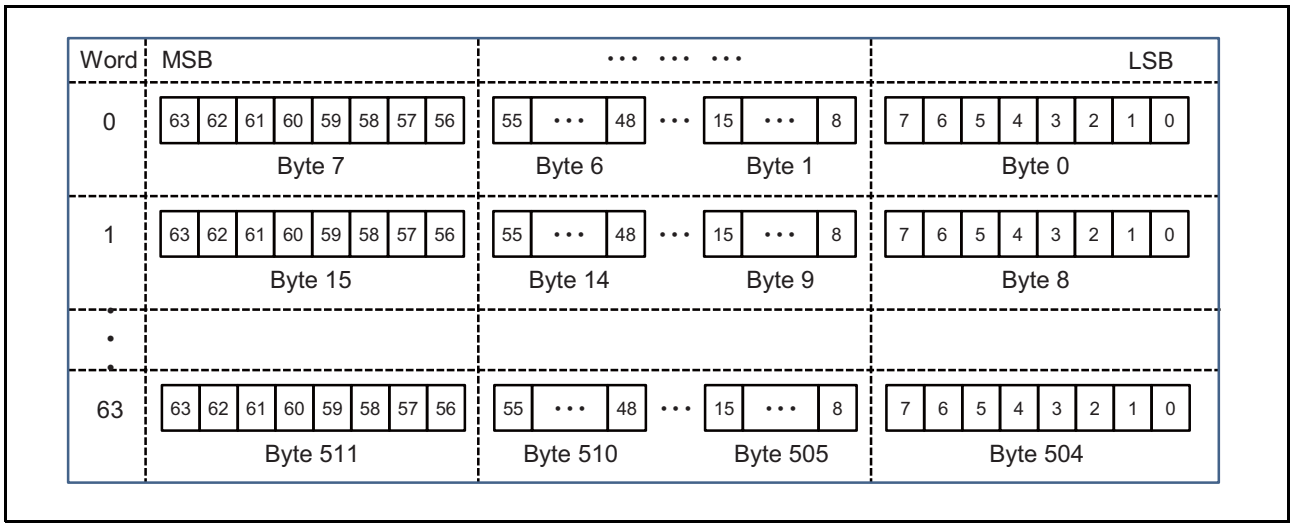


Figure 49.7 SD_BUF Store Data

Table 49.4 Reading from SD_BUF0

WMODE*1	BUSWIDTH*1	ENDIAN*1	Read Data*2
0	0	0	H'0123456789ABCDEF
0	0	1	H'EFCDAB8967452301
1	1	0	H'89ABCDEF (1st) H'01234567 (2nd)
1	1	1	H'EFCDAB89 (1st) H'67452301 (2nd)
1	0	0	H'CDEF (1st) H'89AB (2nd) H'4567 (3rd) H'0123 (4th)
1	0	1	H'EFCD (1st) H'AB89 (2nd) H'6745 (3rd) H'2301 (4th)

Note 1. The name of a bit in HOST_MODE.
Note 2. When the data stored in SD_BUF is H'0123456789ABCDEF

(2) Bus Signal Voltage Switch

Change the electric potential of the bus signal in the following procedure after checking that the SD card supports 1.8 V.

(i) Issuing CMD11
Perform command sequence processing of CMD11.

(ii) Stopping the SD clock (a)
Set the SCLKEN bit in the SD_CLK_CTRL to 0 to stop* the output of the SD clock. When the SDCLKOFFEN bit in the SD_CLK_CTRL register is 1, the SDCLKOFFEN bit is also set to 0.
Note: When the SDCLKOFFEN bit in the SD_CLK_CTRL register is 1, the SD clock has automatically been stopped.

(iii) Checking the value of SDDAT
Check that the DAT0 bit in the SD_INFO2 register is 0.

(iv) Changing the supply voltage of the host device
Change the voltage which is supplied through the power supply pin of the given channel (PVcc_SD0 for channel 0 or PVcc_SD1 for channel 1) from 3.3 V to 1.8 V. In addition, set the POC bit for the given channel (POC2 for channel 0 or POC3 for channel 1) of the dedicated pin POC control register (PPOC) for the GPIO to 0. Also, make the GPIO settings for 1.8-V power supply in the bits of SD/MMC host interface dedicated pin driving ability control registers 0 to 2 (PSDMMC0 and PSDMMC1 for channel 0 or PSDMMC2 for channel 1).
For details, see the following sections:
section 51.3.30, Dedicated Pin POC Control Register (PPOC),
section 51.3.31, SD/MMC Host Interface Dedicated Pin Driving Ability Control Register 0 (PSDMMC0),
section 51.3.32, SD/MMC Host Interface Dedicated Pin Driving Ability Control Register 1 (PSDMMC1),
section 51.3.33, SD/MMC Host Interface Dedicated Pin Driving Ability Control Register 2 (PSDMMC2).

(v) Starting supply of the SD clock (b)
After the SD clock has been stopped ((a) above) and 5 ms or more has elapsed, set the SCLKEN bit in the SD_CLK_CTRL register to 1 and allow the output of the SD clock. The SDCLKOFFEN bit must be 0.

(vi) Checking the value of SDDAT
After supplying the SD clock has been started ((b) above) and 1 ms or more has elapsed, check that the DAT0 bit in the SD_INFO2 register is 1. It is possible to set the SDCLKOFFEN bit in the SD_CLK_CTRL register to 1 and allow SD Clock (SDCLK) Output Automatic Control Enable.

49.3.2 Card Detect/Write Protect

(1) Card Detect

The SD/MMC host interface has two types of card detect functions as described in the following.

- Card detect with ISDCD

Figure 49.8 shows the timing chart of card detect using ISDCD. ISDCD is connected to the card socket and pulled up on the host device. The resistance of the pull-up resistor is decided by the specification of the SD host device.

[Card insertion]
ISDCD is pulled down when a card is inserted. At this time, if ISDCD has been pulled down for the Mcycle period (set in SD_OPTION), INFO4 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

[Card removal]
ISDCD is pulled up when a card is removed. At this time, if ISDCD has been pulled up for the Mcycle period (set in SD_OPTION), INFO3 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

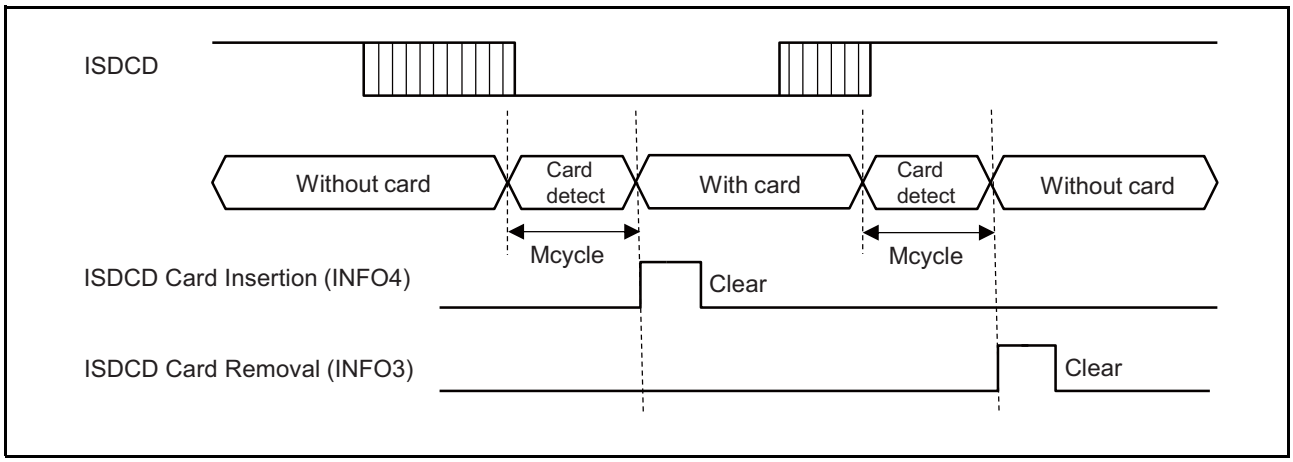


Figure 49.8 Example of Card Detect with ISDCD

- SD card detect with SDDAT3

Figure 49.9 shows the timing chart when the SD card is detected using SDDAT3. In addition, SDDAT3 is pulled down on the host device. The resistance of the pull-down resistor is decided by the specification of the SD host device.

[Card insertion]
When an SD card is inserted, SDDAT3 is pulled up. Accordingly, INFO9 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

[Card removal]
When an SD card is removed, SDDAT3 is pulled down. Accordingly, INFO8 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

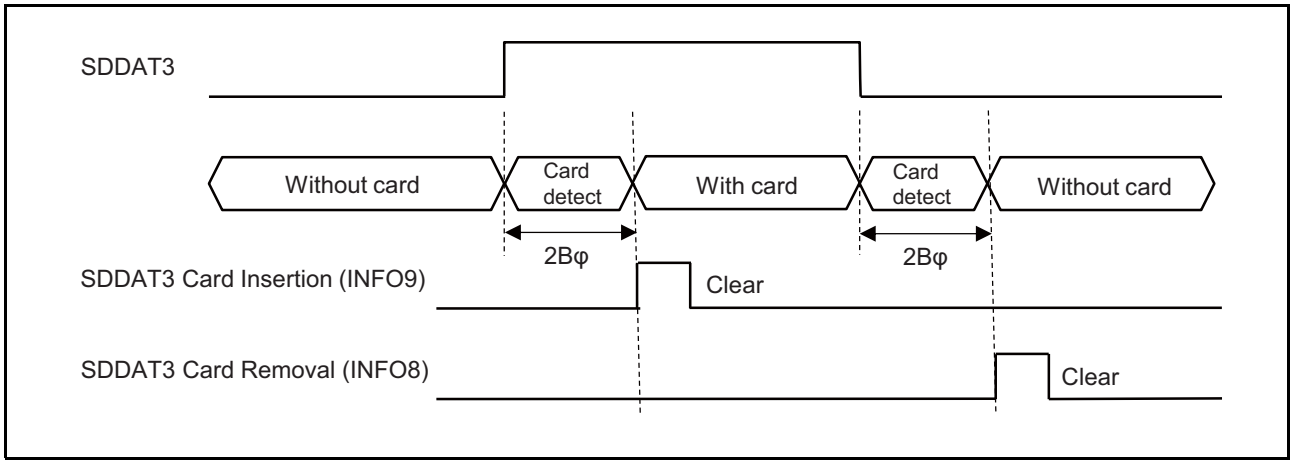


Figure 49.9 SD Card Detect with SDDAT3

(2) Write Protect

The SD/MMC host interface has two types of write protect functions.

- Write protect with ISDWP

ISDWP is connected to the card socket, and pulled up or pulled down by the card insertion. The selection of pulling up or pulling down and the resistance value is decided by the specifications of the SD host device. As the ISDWP state is reflected to INFO7 in SD_INFO1, the write protect is decided after the SD card is inserted.

- Write protect with command

The card's internal write protection and the card lock/unlock operation are realized by the command.

49.3.3 Interrupt Request

(1) Interrupt Request

The SD/MMC host interface has the interrupt requests shown in Table 49.5 shows the relationship between the interrupt flag registers and the interrupt mask registers. When a bit in an interrupt mask register is set to 0, an interrupt occurs by setting the corresponding bit in the interrupt flag register to 1.
To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Table 49.5 Interrupt Request

Interrupt Request	Interrupt Flag Register		Interrupt Mask Register	
	Register Name	Bit Name	Register Name	Bit Name
Card access interrupt	SD_INFO1	INFO2	SD_INFO1_MASK	IMASK2
		INFO0		IMASK0
	SD_INFO2	ILA	SD_INFO2_MASK	IMASK
		BWE		BMASK1
		BRE		BMASK0
		ERR6		EMASK6
		ERR5		EMASK5
		ERR4		EMASK4
		ERR3		EMASK3
		ERR2		EMASK2
		ERR1		EMASK1
		ERR0		EMASK0
SDIO access interrupt	SDIO_INFO1	EXWT	SDIO_INFO1_MASK	MEXWT
		EXPUB52		MEXPUB52
		IOIRQ		IOMSK
Card detect interrupt	SD_INFO1	INFO9	SD_INFO1_MASK	IMASK9
		INFO8		IMASK8
		INFO4		IMASK4
		INFO3		IMASK3
DMAC interrupt	DM_CM_INFO1	DTRANEND1	DM_CM_INFO1_MASK	DTRANEND1_MASK
		DTRANEND0		DTRANEND0_MASK
		SEQEND		SEQEND_MASK
	DM_CM_INFO2	DTRANERR1	DM_CM_INFO2_MASK	DTARERR1_MASK
		DTRANERR0		DTARERR0_MASK
		SEQERR		SEQERR_MASK

49.3.4 Communications Errors and Timeouts

- Communications Errors and Timeouts

Table 49.6 and Table 49.7 show the relationships between the SD card interrupt flag register and SD error status register for communications errors and timeouts, respectively. When a bit in the SD card interrupt flag register is set to 1, the corresponding bit in the SD error status register is set to 1. The values of the SD error status register are cleared by writing to SD_CMD or writing 0 to the SDRST bit in SOFT_RST.

Table 49.6 Communications Errors

Communication Error	SD Interrupt Flag Register		SD Error Status Register		
	Register Name	Bit Name	Register Name	Bit Name	Description
END error	SD_INFO2	ERR2	SD_ERR_STS1	E5	When an error occurs in the CRC status length
				E4	When an error occurs in read data length
				E3	When an error occurs in the response length to a command issued within a command sequence
				E2	When an error occurs in the response length (other than a response to a command issued within a command sequence)
CRC error		ERR1		E11	When an error occurs in the CRC status
				E10	When a CRC error occurs in the read data
				E9	When a CRC error occurs in the response to a command issued within a command sequence
				E8	When a CRC error occurs in the response (other than a response to a command issued within a command sequence)
CMD error		ERR0		E1	The command index of the transmitted command differed from the command index of the received response (for a command issued within a command sequence)
				E0	The command index of the transmitted command differed from the command index of the received response (for a command issued other than within a command sequence)

Table 49.7 Timeouts

Timeout	SD Interrupt Flag Register		SD Error Status Register		Description
	Register Name	Bit Name	Register Name	Bit Name	
Response timeout	SD_INFO2	ERR6	SD_ERR_STS2	E1	When the response to a command issued within a command sequence is not received even after 640 cycles of SDCLK have elapsed
				E0	When the response (other than a response to a command issued within a command sequence) is not received even after 640 cycles of SDCLK have elapsed
Data timeout (other than response timeout)		ERR3		E6	When the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle* after the CRC status
				E5	When the CRC status is not received t even after Ncycle* has elapsed after data writing
				E4	When read data is not received even after Ncycle* has elapsed after read command
					When read data for the next block are not received even after Ncycle* has elapsed after the reception of read data
					When read data for the next block are not received even after Ncycle* has elapsed after release of the read wait state
				E3	When the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle* after CMD12 has been issued within a command sequence
				E2	When the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle* after R1b response

Note: *Ncycle is set by bit 7 to bit 4 in SD_OPTION.

49.4 Usage Example

49.4.1 Command without Data Transfer

(1) Flowchart

Figure 49.10 and Figure 49.11 show flowchart examples.

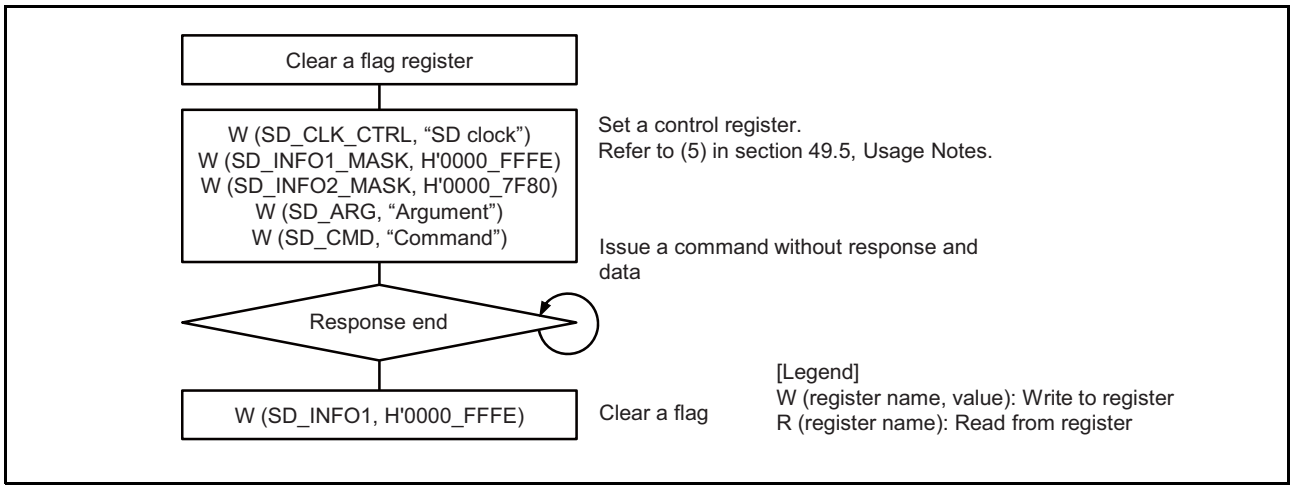


Figure 49.10 Flow Example of Command without Response and Data

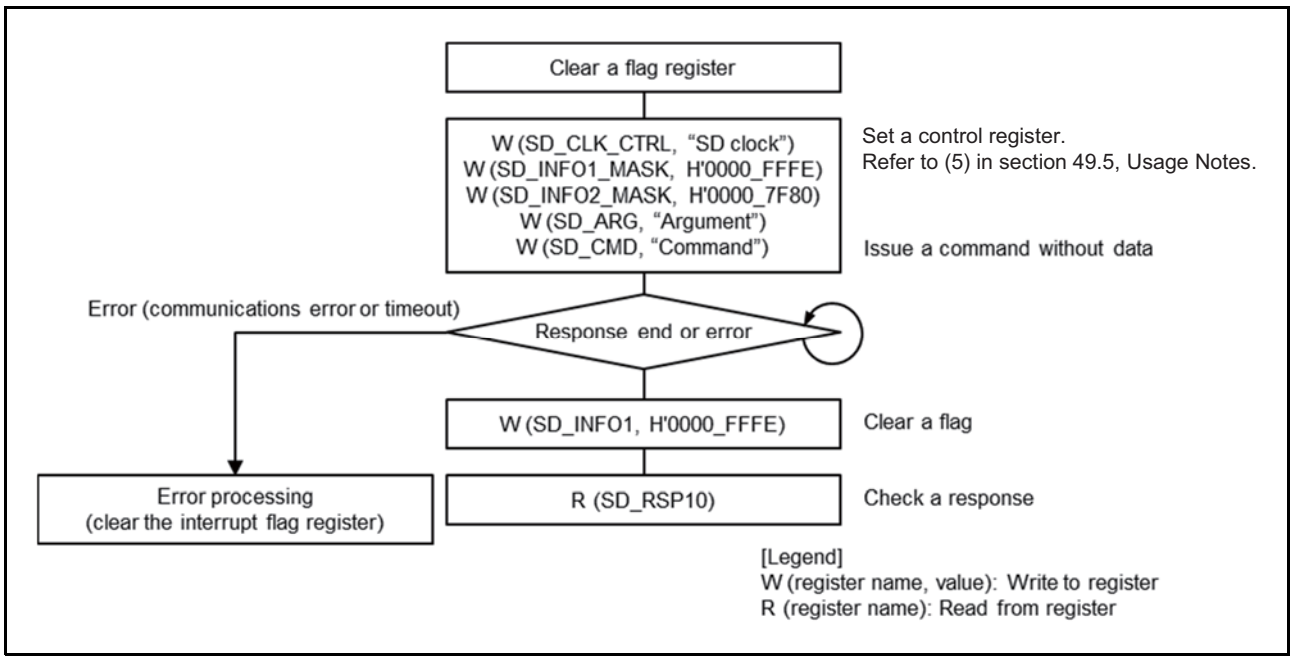


Figure 49.11 Flow Example of Command without Data

(2) Operation for Command without Data Transfer

The legend below is used for description of register read/write.

W (register name, value):Write to register
R (register name):Read from register

The operation is described as below.

(a) Command without response and data

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), interrupt mask, and so on. (SD_CLK_CTRL, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue
   Set CMD Argument in SD_ARG and write to SD_CMD.
   Accordingly, CMD is issued, and the operation is started.
4. Flag clear
   When transmission of a command is completed, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0.

(b) Command without data

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), interrupt mask, and so on. (SD_CLK_CTRL, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue
   Set CMD Argument in SD_ARG and write to SD_CMD.
   Accordingly, CMD is issued, and the operation is started.
4. Flag clear
   When a response is received, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0.
5. Read a response from SD_RSP10.
   Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

49.4.2 Single Block Read

(1) Flowchart

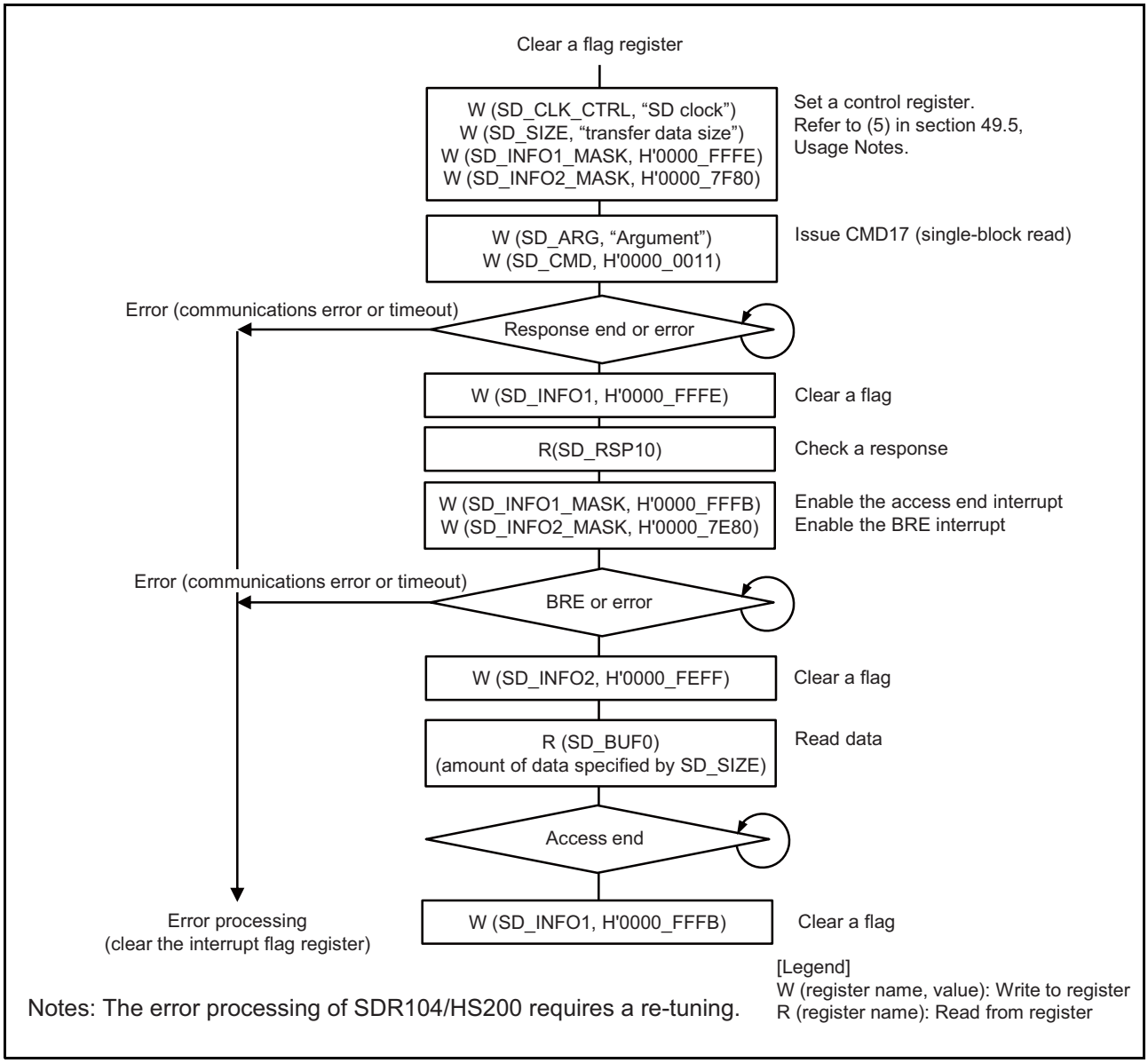


Figure 49.12 Single Block Read Flowchart Example

(2) Operation for Single Block Read

The operation of the single block read is described below.

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue (CMD17)
   Set CMD17 Argument in SD_ARG and write H'0000 0011 to SD_CMD.
   Accordingly, CMD17 is issued, and the single block read operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP10.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP or the IOABT bit in SDIO_MODE to 1. Furthermore, this causes CMD12 and CMD52 to not be issued. If the command sequence is halted, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt.
5. Data receive from SD card and data read
   Write H'0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000 7E80 to SD_INFO2_MASK to enable the BRE interrupt. When the data receive from the SD card is completed, the BRE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BRE bit to 0 and read the amount of data specified by SD_SIZE from SD_BUF0.
   However, a communications error or timeout may be generated if data are being received while reading of SD_BUF0 is in progress.
6. Operation complete
   When the data read from SD_BUF0 is completed, INFO2 (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO2 to 0 to end the single block read operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

49.4.3 Single Block Write

(1) Flowchart

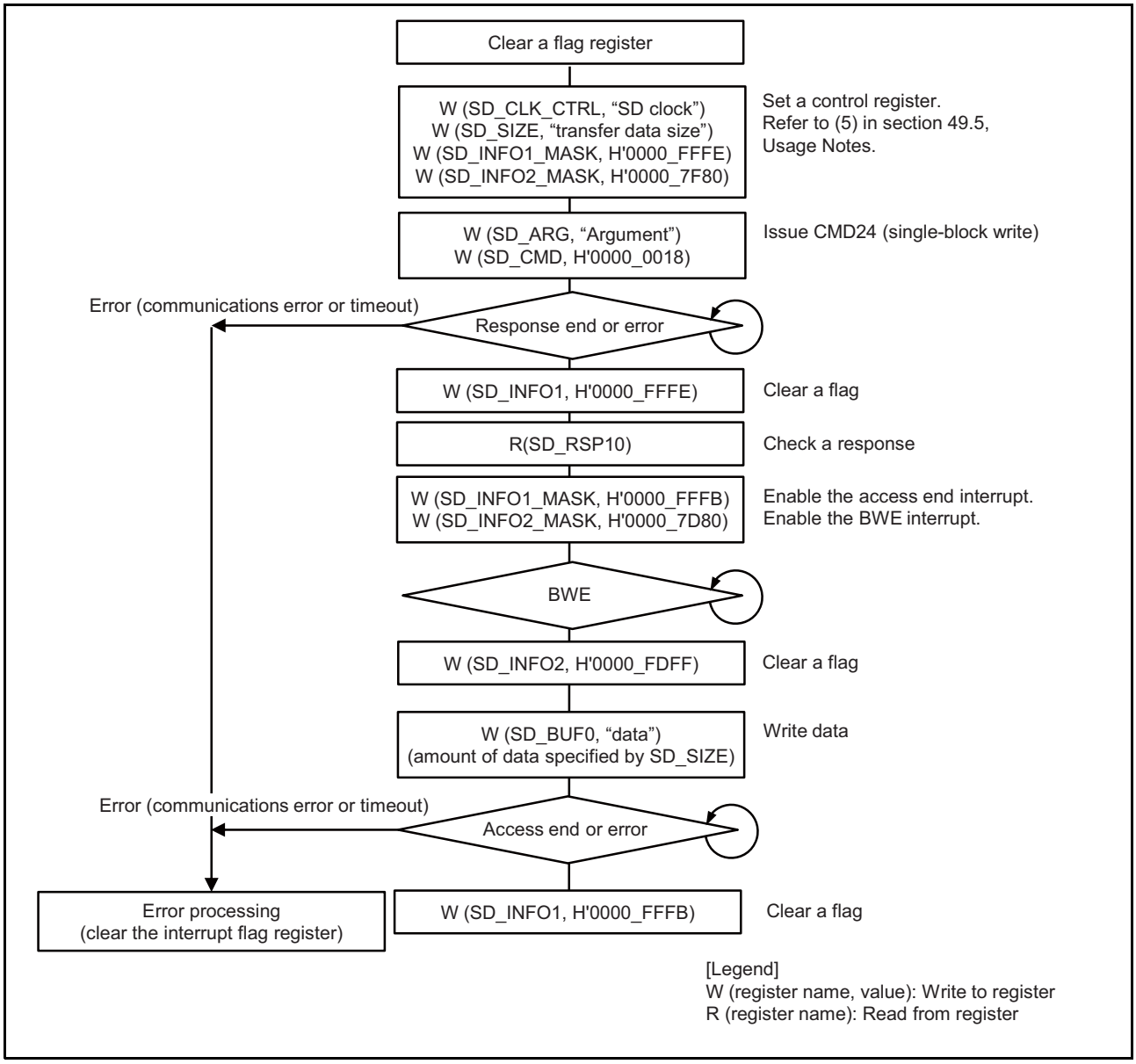


Figure 49.13 Single Block Write Flowchart Example

(2) Operation for Single Block Write

The operation of the single block write is described below.

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue (CMD24)
   Set CMD24 Argument in SD_ARG and write H'0000 0018 to SD_CMD.
   Accordingly, CMD24 is issued, and the single block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP10.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP or the IOABT bit in SDIO_MODE to 1. Furthermore, this causes CMD12 and CMD52 to not be issued. If the command sequence is halted, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt.
5. Data write and data transmit to SD card
   Write H'0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000 7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card.
   However, a communications error or timeout may be generated if data are being transmitted after writing to SD_BUF0.
6. Operation complete
   When the CRC status and busy state are received from the SD card, INFO2 (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO2 bit to 0 to end the single block write operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

49.4.4 Multiple Block Read

(1) Flowchart

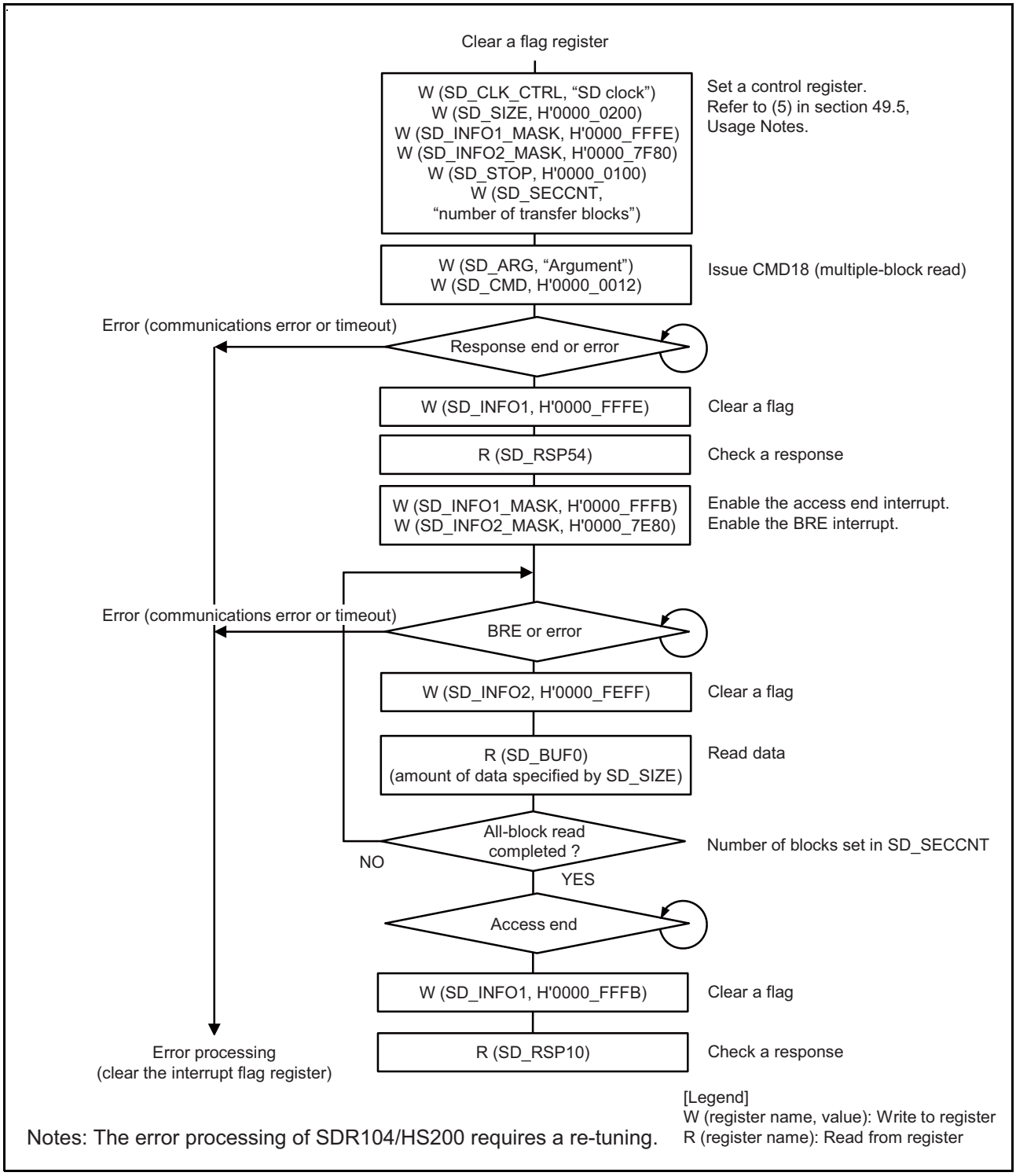


Figure 49.14 Multiple Block Read Flowchart Example

(2) Operation for Multiple Block Read

The operation of the multiple block read is described below.

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD18)
   Set CMD18 Argument in SD_ARG and write H'0000 0012 to SD_CMD.
   Accordingly, CMD18 is issued, and the multiple block read operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data receive from SD card and data read
   Write H'0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000 7E80 to SD_INFO2_MASK to enable the BRE interrupt. When one-block data receive from the SD card is completed, the BRE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BRE bit to 0 and read the amount of data specified by SD_SIZE from SD_BUF0. Doing this repeats transfer of the number of blocks set in SD_SECCNT. However, a communications error or timeout may be generated if data are being received while reading of SD_BUF0 is in progress. CMD12 is automatically issued to stop multi-block transfer with the number of blocks which is set to SD_SECCNT and the response is received. At this point, CMD12 Argument is automatically set to H'0000 000.
6. Operation complete
   When all-block data read and the CMD12 response receive are completed, INFO2 (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO2 to 0 to read the response. This is the end of multiple block read operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

49.4.5 Multiple Block Write (when Using Internal Timer)

(1) Flowchart

Figure 49.15 shows the flowchart when using an internal timer.

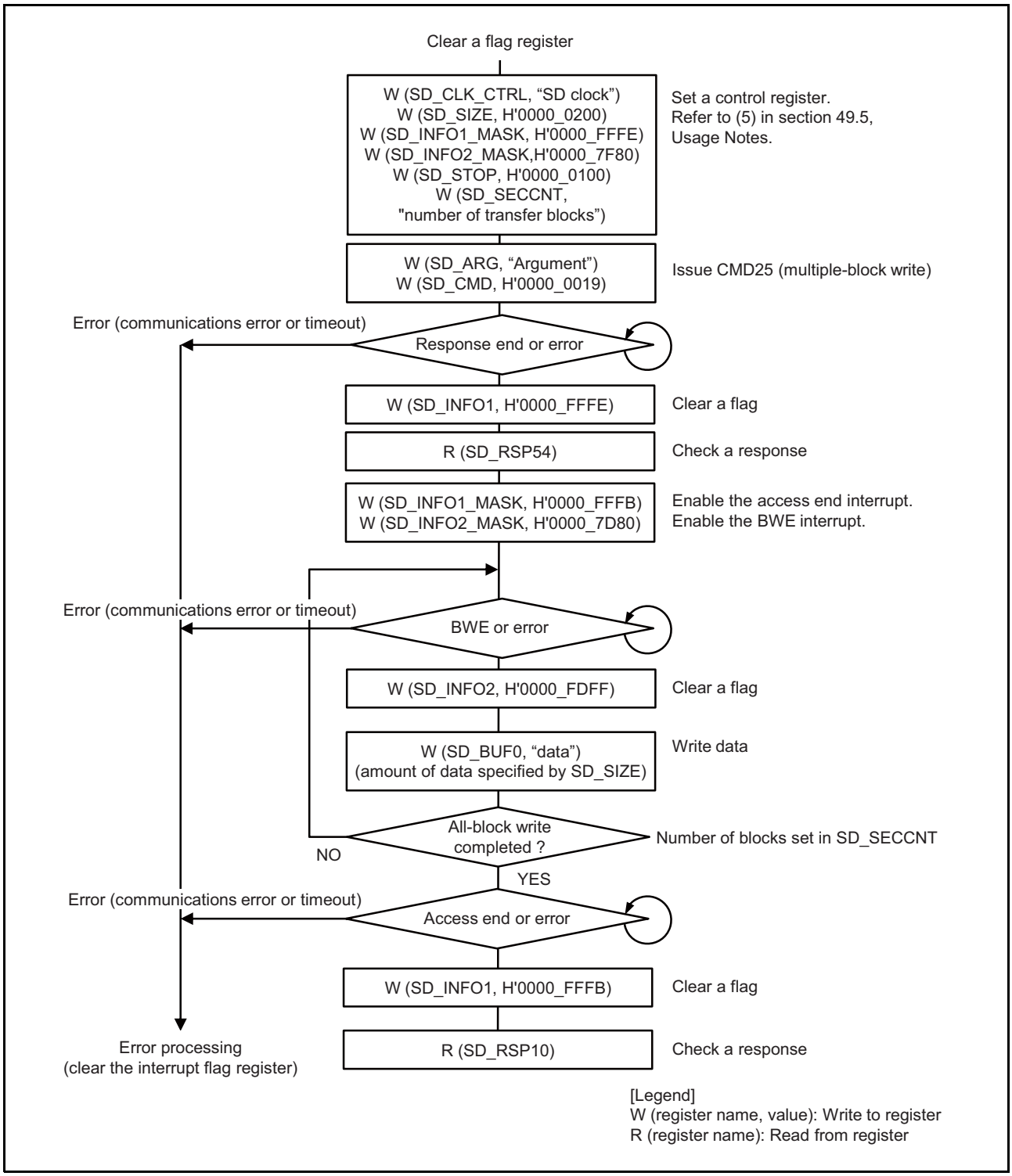


Figure 49.15 Multiple Block Write Flowchart Example (when Using Internal Timer)

(2) Operation for Multiple Block Write

The operation of the multiple block write is described below.

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD25)
   Set CMD25 Argument in SD_ARG and write H'0000 0019 to SD_CMD.
   Accordingly, CMD25 is issued, and the multiple block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) bit in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card
   Write H'0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000 7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card. Doing this repeats transfer of the number of blocks set in SD_SECCNT.
   However, a communications error or timeout may be generated if data are being transmitted while writing to SD_BUF0 is in progress. CMD12 is automatically issued to stop multi-block transfer with the number of blocks which is set to SD_SECCNT and the response is received. At this point, CMD12 Argument is automatically set to H'0000 0000.
6. Operation complete
   When all-block data transmit and the CRC status receive are completed, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO2 bit to 0 to read the response. This is the end of multiple block write operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

49.4.6 Multiple Block Write (when Using External Timer)

(1) Flowchart

The flowchart when using an external timer instead of an internal timer of this module is shown below.

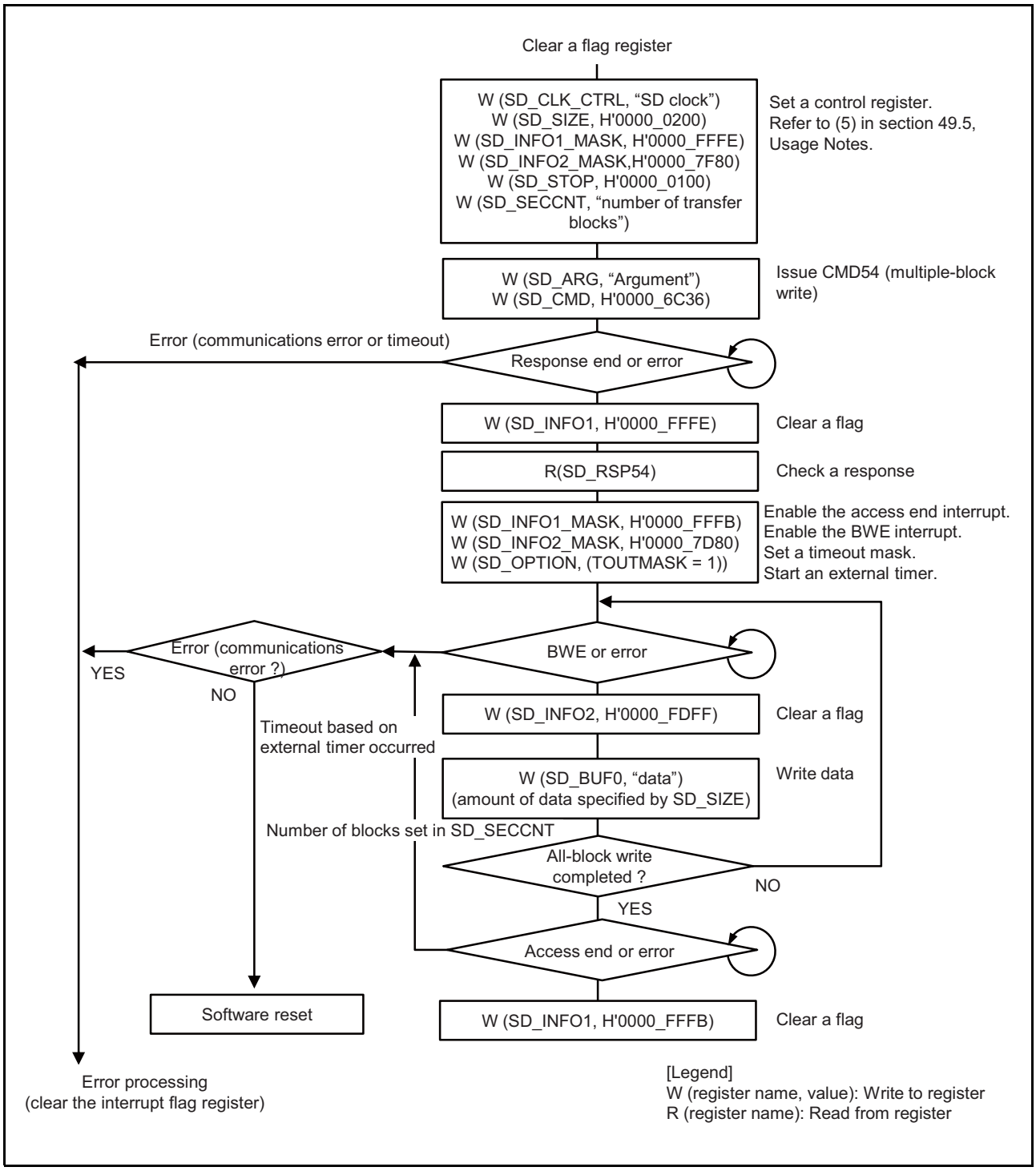


Figure 49.16 Multiple Block Write Flowchart Example (when Using External Timer)

(2) Operation for Multiple Block Write

The operation of the multiple block write is described below.

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD54)
   Set CMD54 Argument in SD_ARG and write H'0000 6C36 to SD_CMD.
   Accordingly, CMD54 is issued, and the multiple block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) bit in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card
   Write H'0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000 7D80 to SD_INFO2_MASK to enable the BWE interrupt. Set the TOUTMASK bit in SD_OPTION to disable timeout and start an external timer.
   When SD_BUF0 is ready for the data to be written, the BWE bit in the SD_INFO2 resister is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card. Doing this repeats transfer of the number of blocks set in SD_SECCNT. However, a communications error may be generated if data are being transmitted while writing to SD_BUF0 is in progress.
6. Operation complete
   When all-block data transmit and the CRC status receive are completed, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO2 bit to 0 to read the response. This is the end of multiple block write operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs at response reception (a communications error or timeout) or at data transmission. Perform a software reset if a timeout occurs at data transmission based on an external timer.

49.4.7 IO_RW_DIRECT Commend (CMD52)

(1) Flowchart

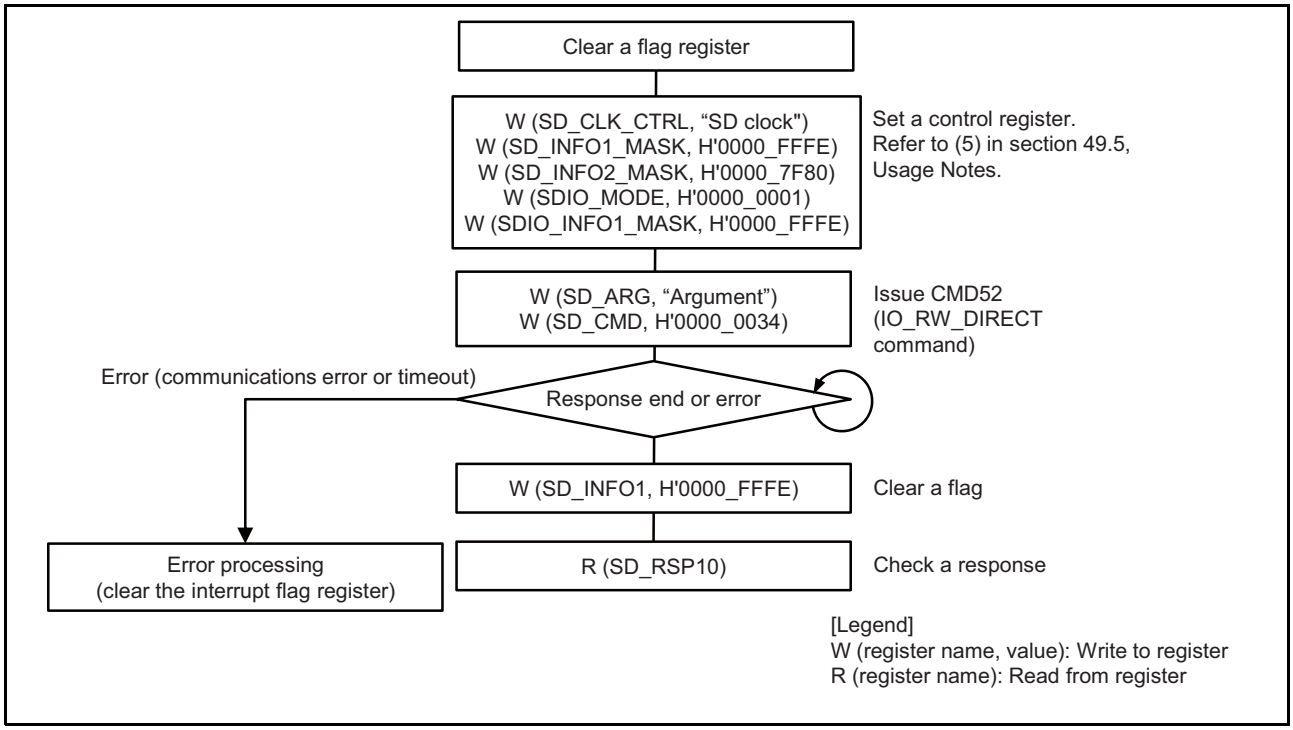


Figure 49.17 IO_RW_DIRECT Command (CMD52) Flowchart Example

49.4.8 IO_RW_EXTENDED (CMD53/Multiple Block Read)

(1) Flowchart

Figure 49.18 shows a flowchart example for CMD53 (multiple block read).

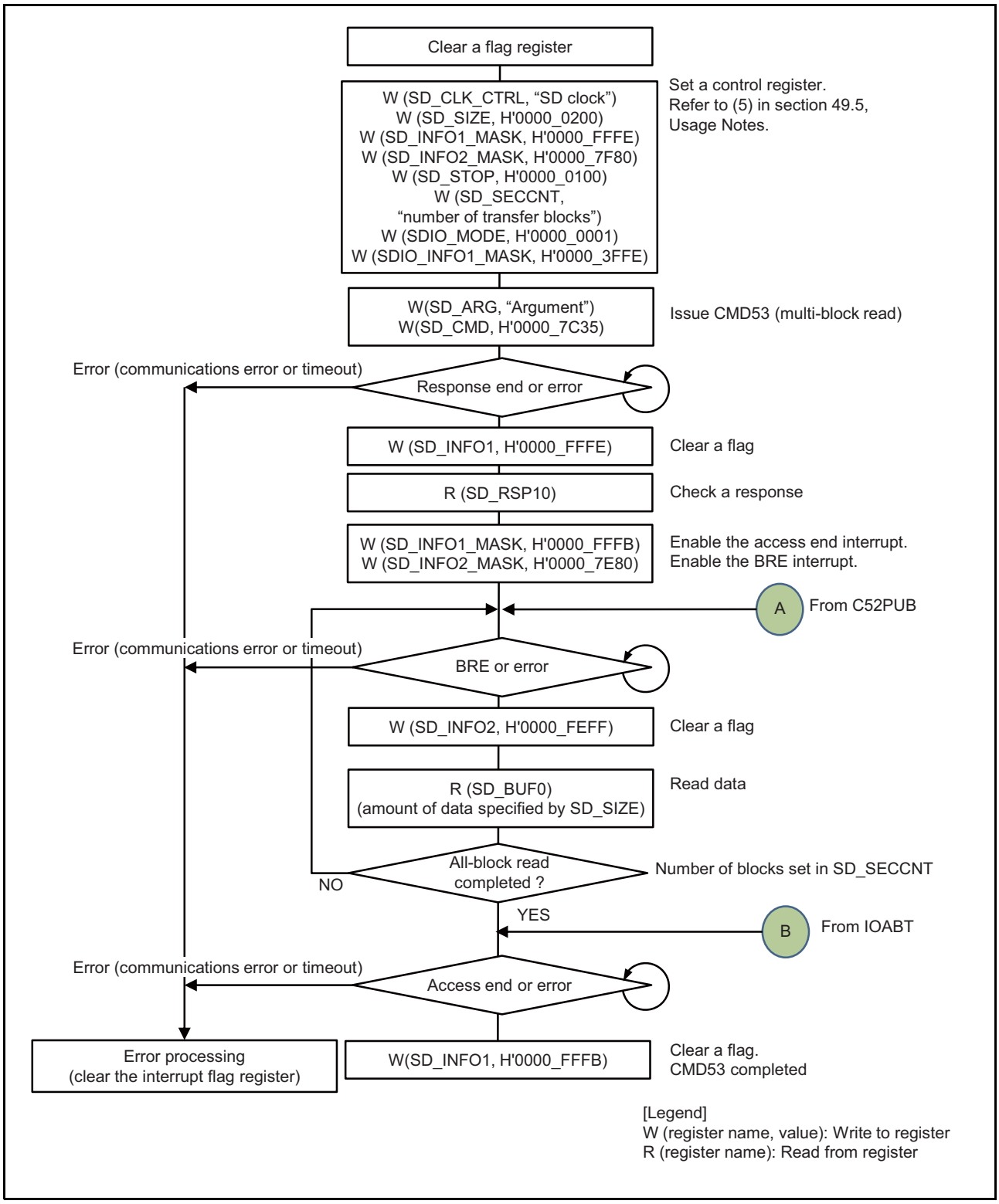


Figure 49.18 CMD53 (Multiple Block Read) Flowchart Example

Figure 49.19 shows a flowchart example when CMD52 (SDIO abort) is issued at CMD53 (multiple block read).

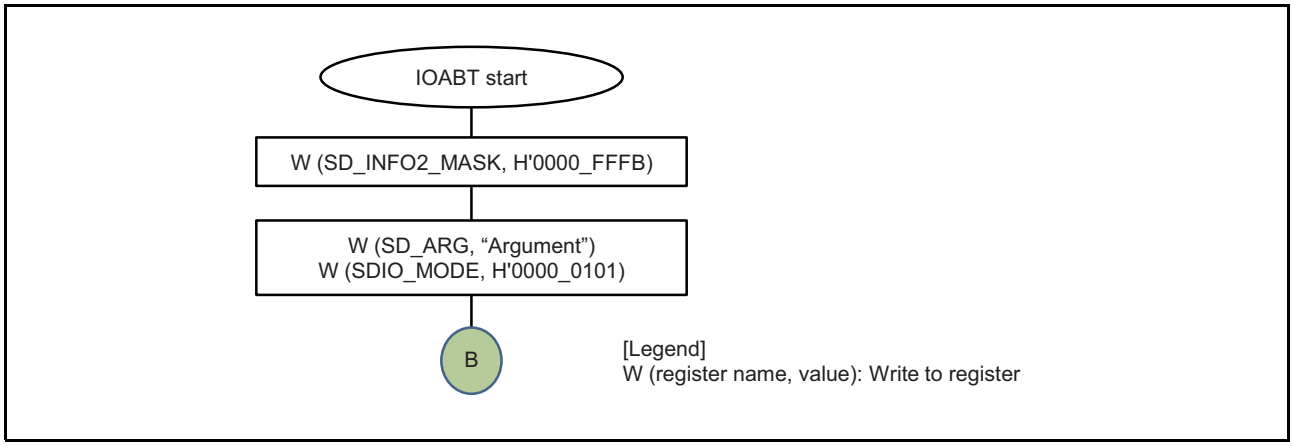


Figure 49.19 Flowchart Example when CMD52 (SDIO Abort) is Issued at CMD53 (Multiple Block Read)

Figure 49.20 shows a flowchart example when CMD52 (SDIO none abort) is issued at CMD53 (multiple block read) while the SD/MMC host interface is in the read wait state.

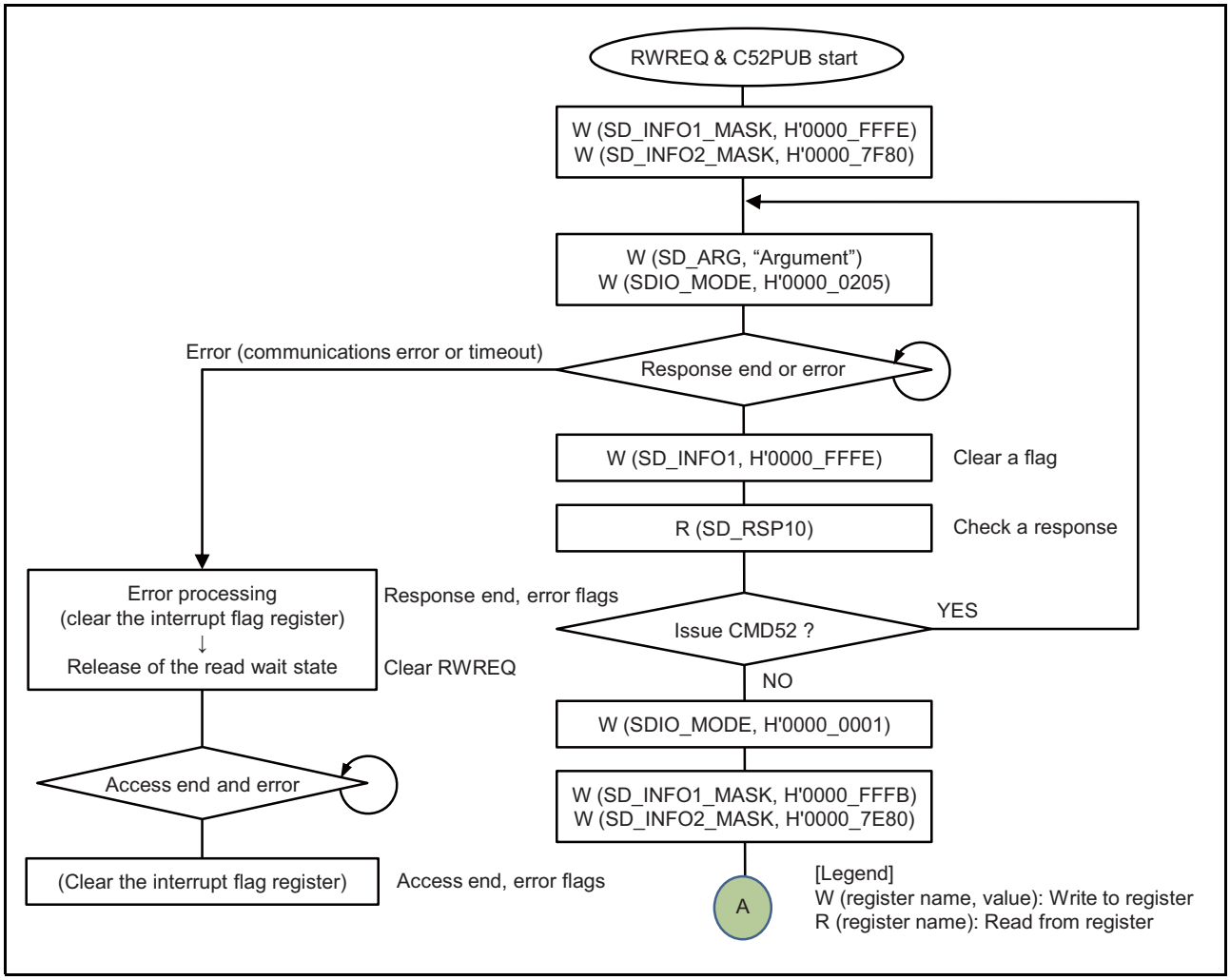


Figure 49.20 Flowchart Example when CMD52 (SDIO None Abort) is Issued after Read Wait State is Entered at CMD53 (Multi Block Read)

49.4.9 IO_RW_EXTENDED (CMD53/Multiple Block Write)

(1) Flowchart

Figure 49.21 shows a flowchart example for CMD53 (multiple block write).

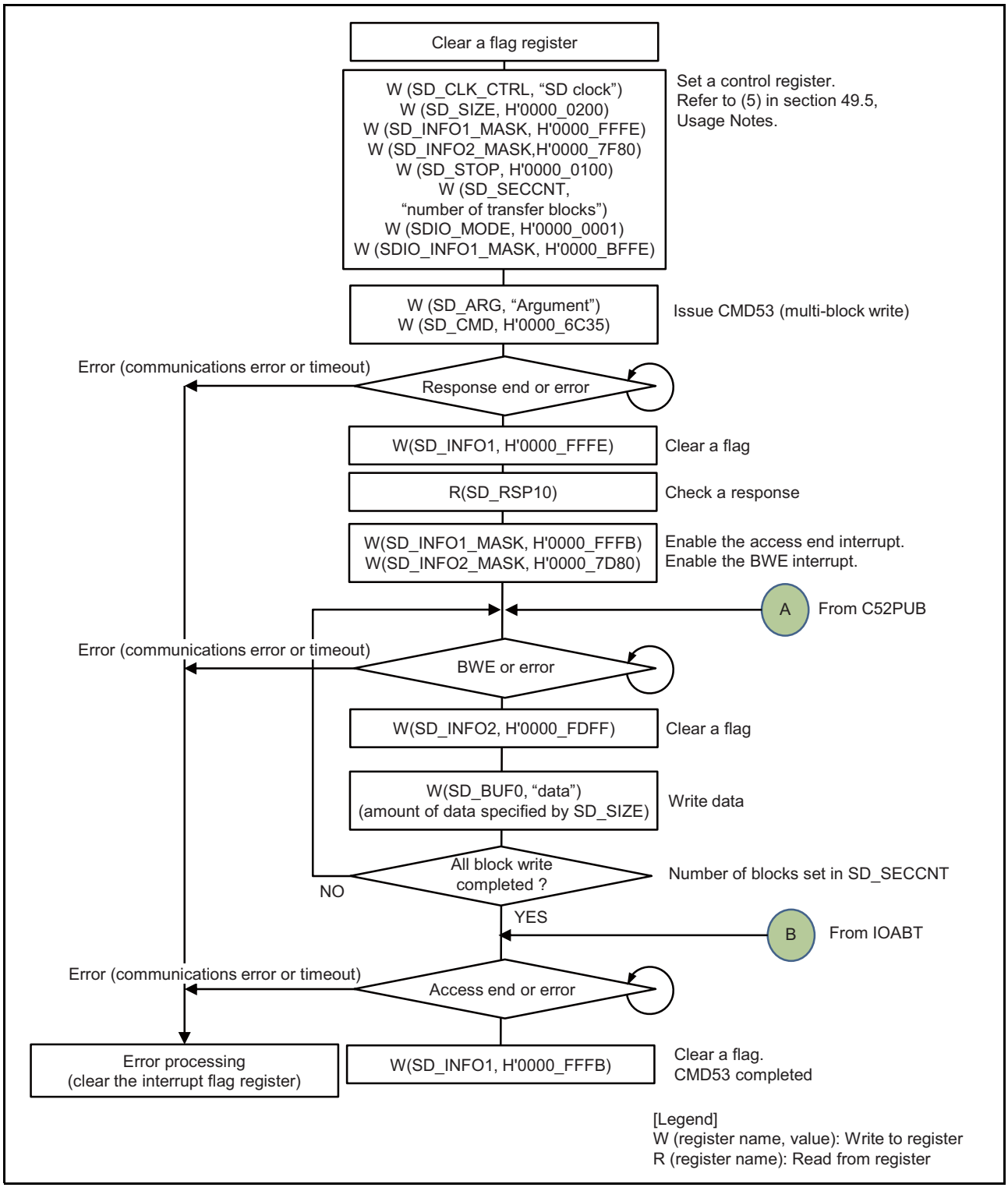


Figure 49.21 CMD53 (Multiple Block Write) Flowchart Example

Figure 49.22 shows a flowchart example when CMD52 (SDIO abort) is issued at CMD53 (multiple block write).

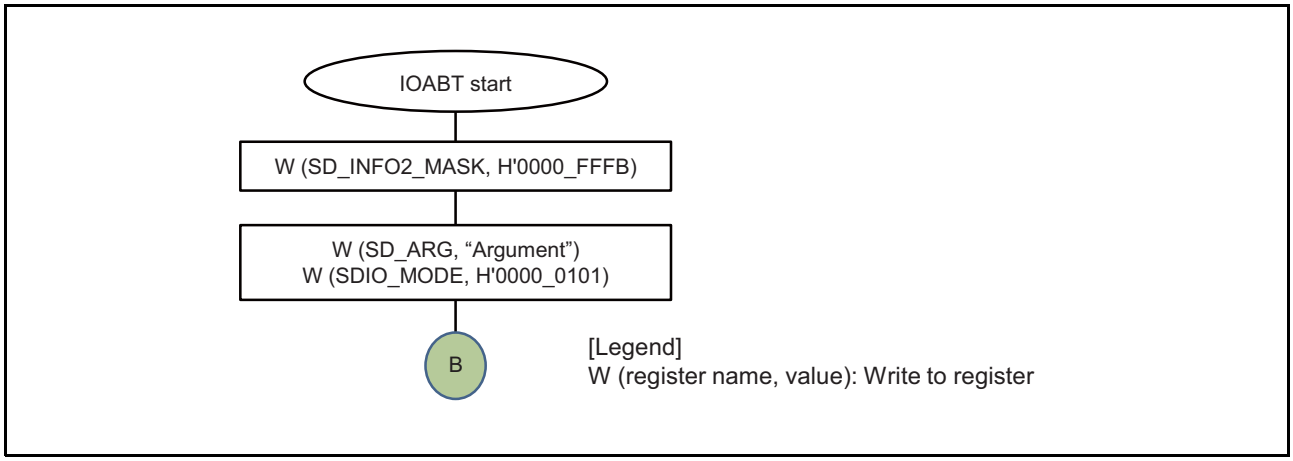


Figure 49.22 Flowchart Example when CMD52 (SDIO Abort) is Issued at CMD53 (Multiple Block Write)

Figure 49.23 shows a flowchart example when CMD52 (SDIO none abort) is issued at CMD53 (multiple block write).

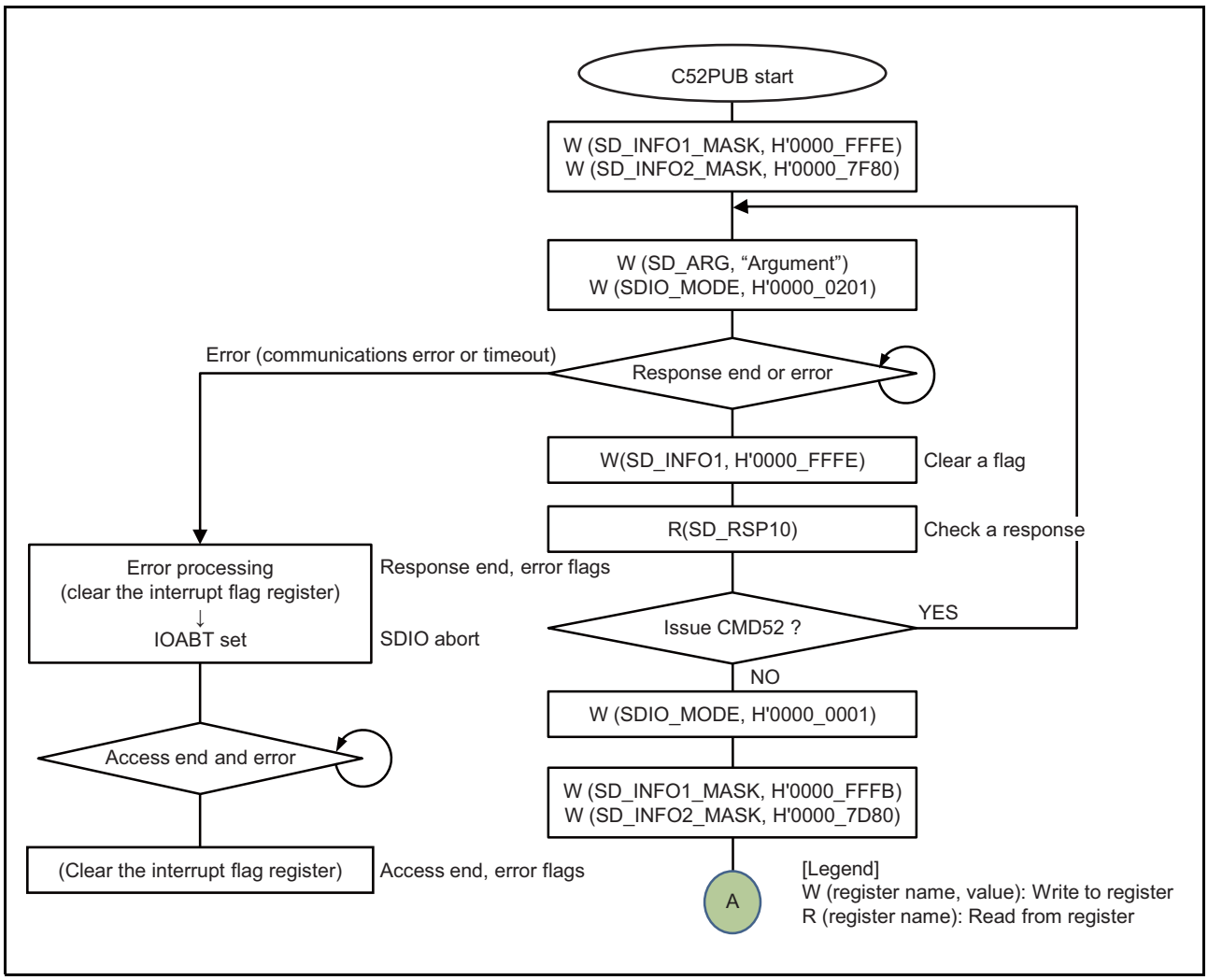


Figure 49.23 Flowchart Example when CMD52 (SDIO None Abort) is Issued at CMD53 (Multiple Block Write)

49.4.10 DMA Transfer

(1) SD_BUF DMA Transfer

Figure 49.24 shows a flowchart example for SD_BUF DMA read when CMD18 (multiple block read) is issued.

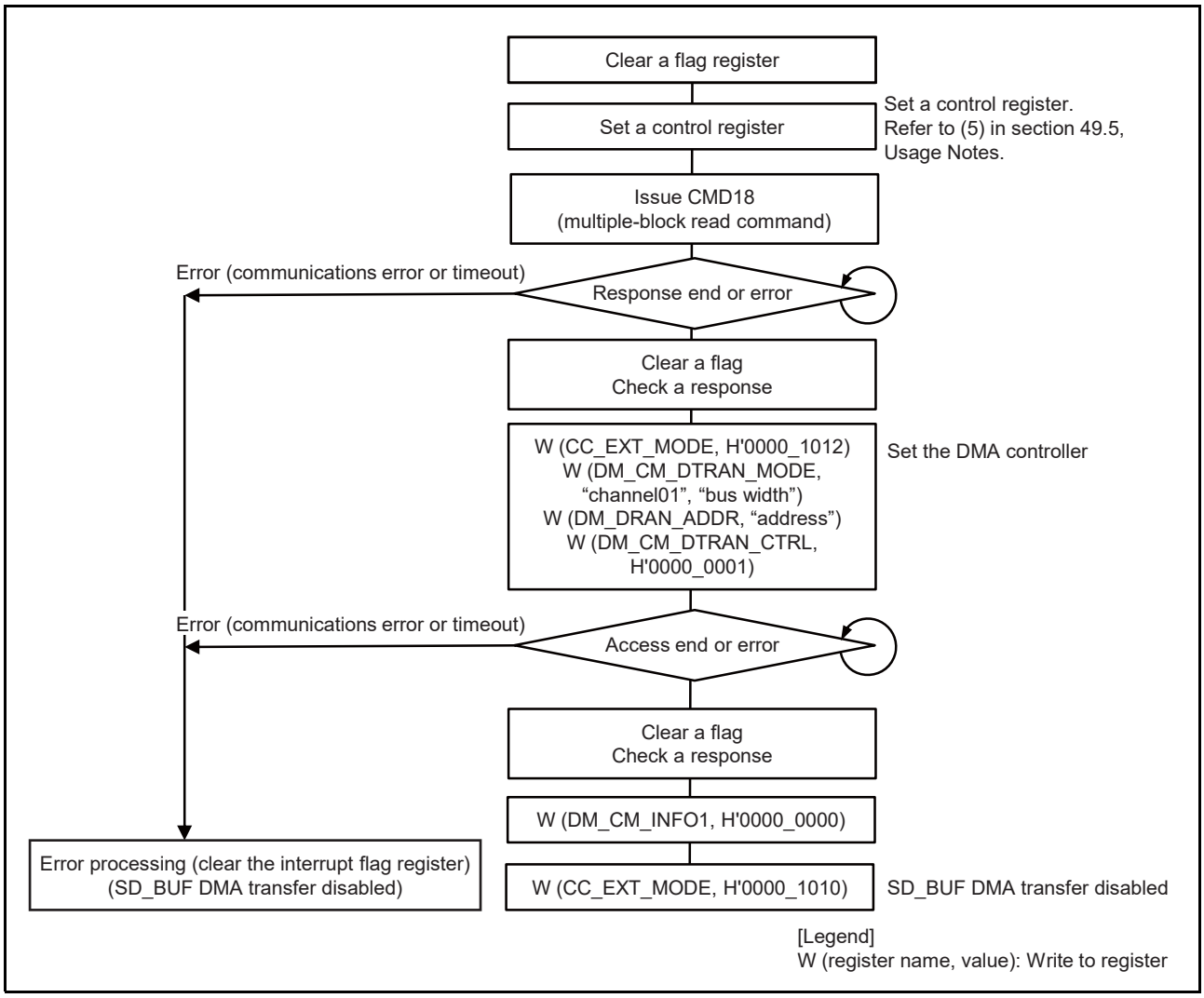


Figure 49.24 SD_BUF DMA Read Flowchart Example

Figure 49.25 shows a flowchart example for SD_BUF DMA write when CMD25 (multiple block write) is issued.

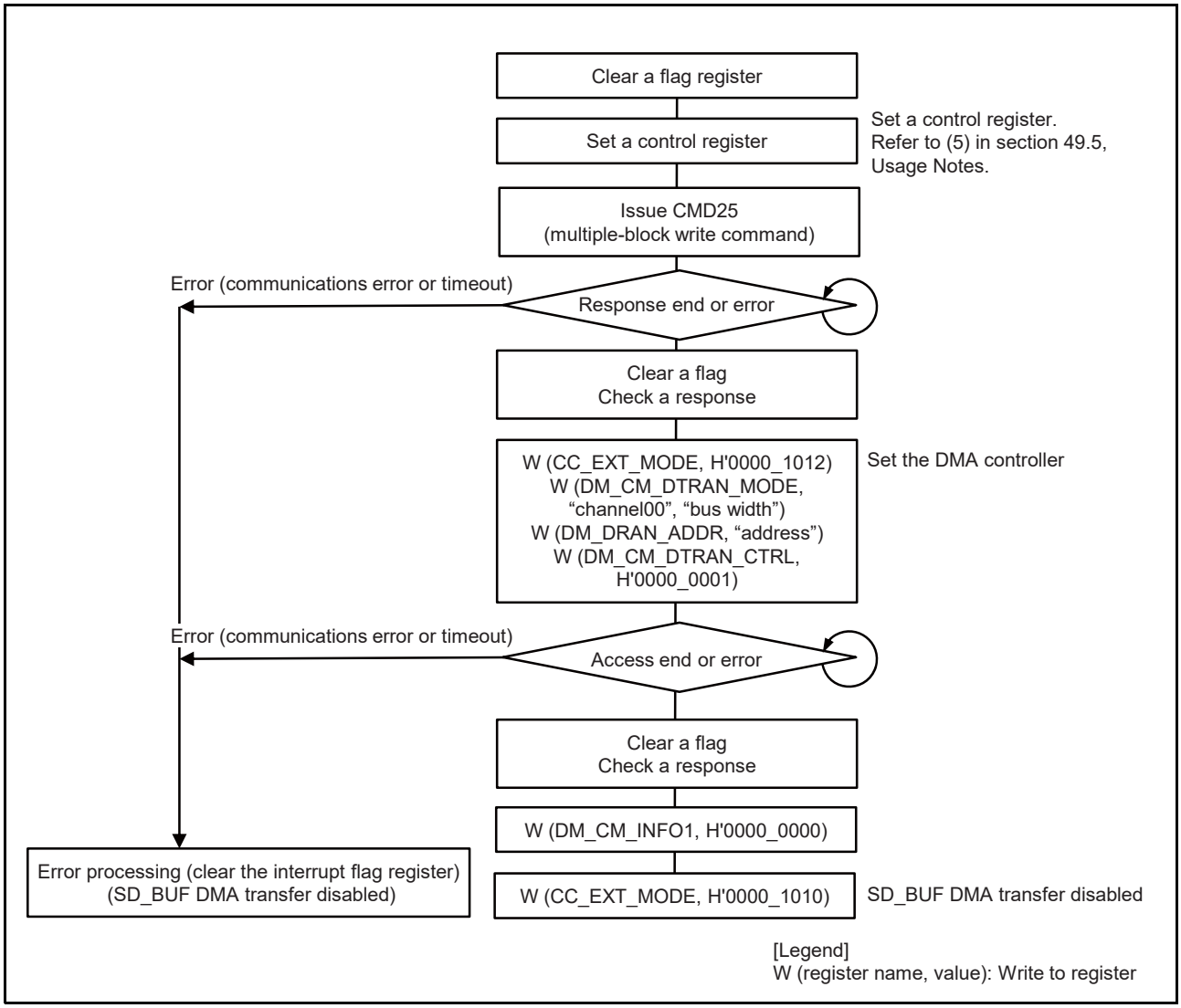


Figure 49.25 SD_BUF DMA Write Flowchart Example

49.4.11 High-Priority Interrupt (without Data Transfer)

(1) Flowchart

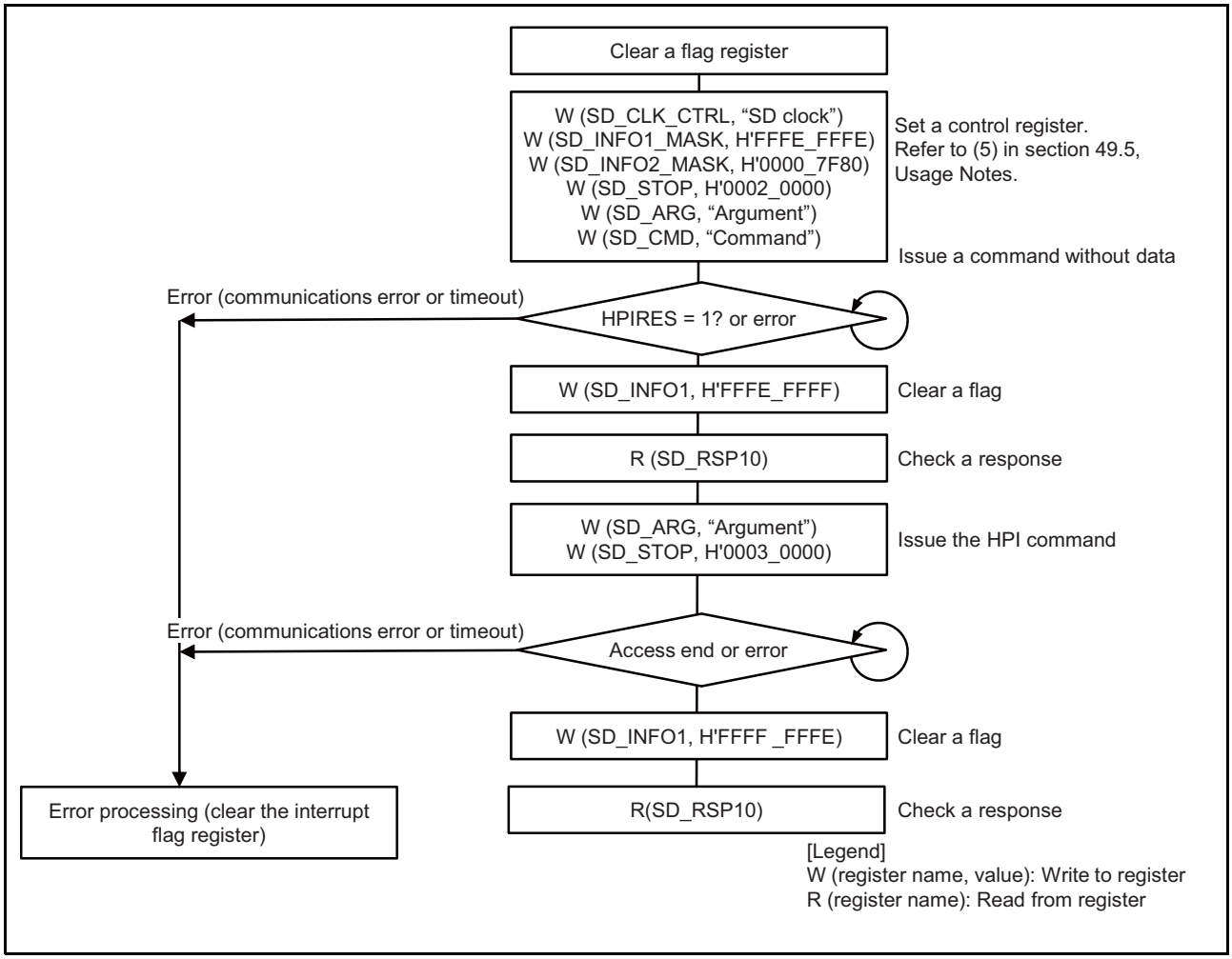


Figure 49.26 Example of the High-Priority Interrupt (without Data Transfer) Flowchart

(2) Operation for High-Priority Interrupt without Data Transfer

The operation of the high-priority interrupt (HPI) without data transfer is described below.

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), HPI enable, interrupt mask, and so on. (SD_CLK_CTRL, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue
   Set the CMD Argument in SD_ARG and write to SD_CMD.
   Accordingly, CMD is issued, and the operation is started.
4. Flag clear
   On receiving the response, the HPIRES bit in SD_INFO1 is set to 1 to generate an interrupt. Clear the HPIRES bit to 0.
5. Read the response from SD_RSP10.
6. HPI command issue
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
7. Operation complete
   When reception of the response to the HPI command is completed and the busy state is released, the INFO0 bit in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO0 bit to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

49.4.12 High-Priority Interrupt (at Single Block Write)

(1) Flowchart

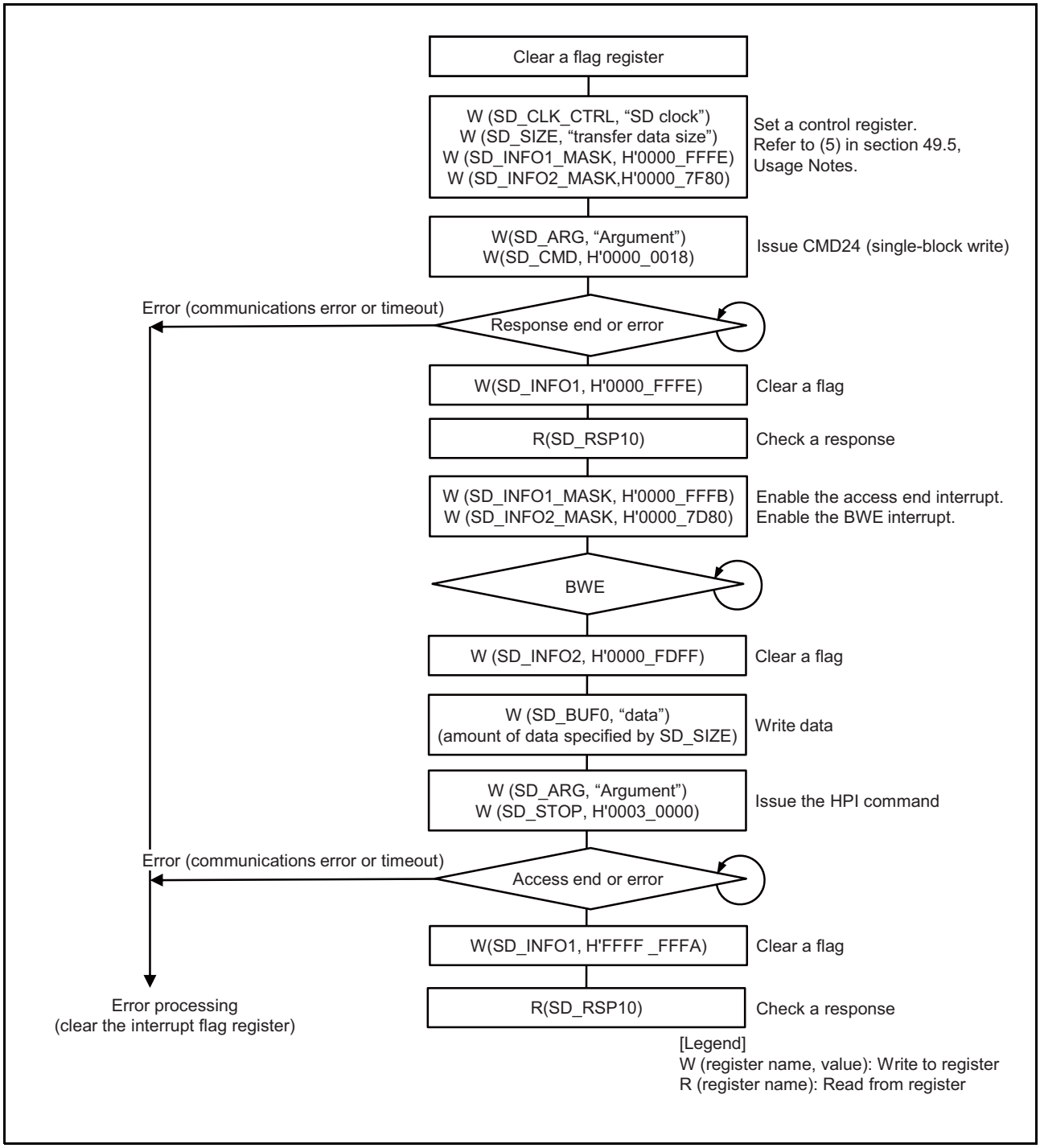


Figure 49.27 Example of the High-Priority Interrupt (at Single Block Write) Flowchart

(2) Operation for HPI at Single Block Write

The operation of HPI at single block write is described below.

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue (CMD24)
   Set CMD24 Argument in SD_ARG and write H'0000 0018 to SD_CMD.
   Accordingly, CMD24 is issued, and the single block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP10.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP or the IOABT bit in SDIO_MODE to 1. Furthermore, this causes CMD12 and CMD52 to not be issued. If the command sequence is halted, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt.
5. Data write and data transmit to SD card
   Write H'0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000 7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card.
6. HPI command issue
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
7. Operation complete
   When reception of the response to the HPI command is completed and the busy state is released, the INFO2 and INFO0 bits in SD_INFO1 are set to 1 to generate an interrupt. Clear the INFO2 and INFO0 bits to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

49.4.13 High-Priority Interrupt (at Multiple Block Write)

(1) Flowchart

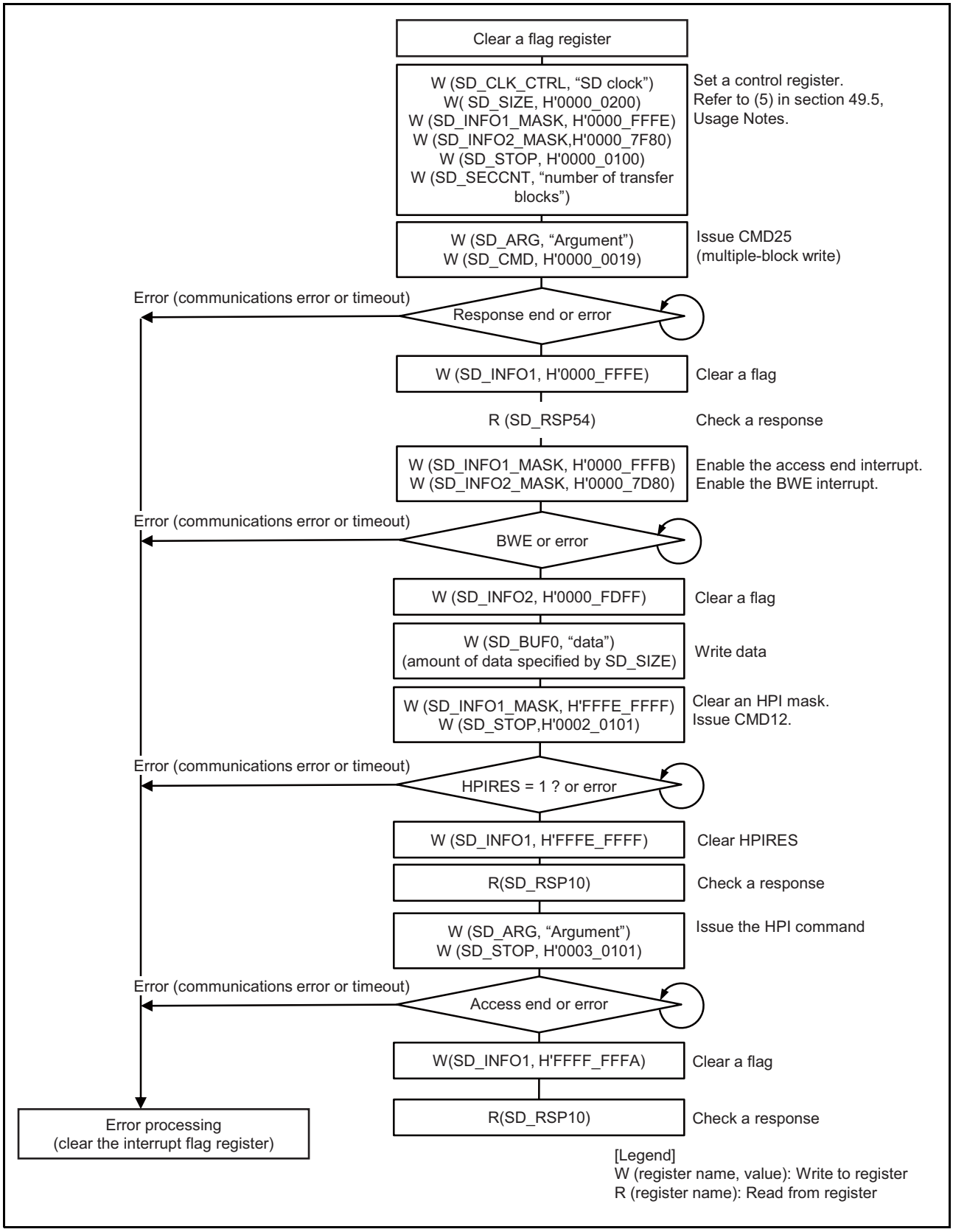


Figure 49.28 Example of the High-Priority Interrupt (at Multiple Block Write) Flowchart (a)

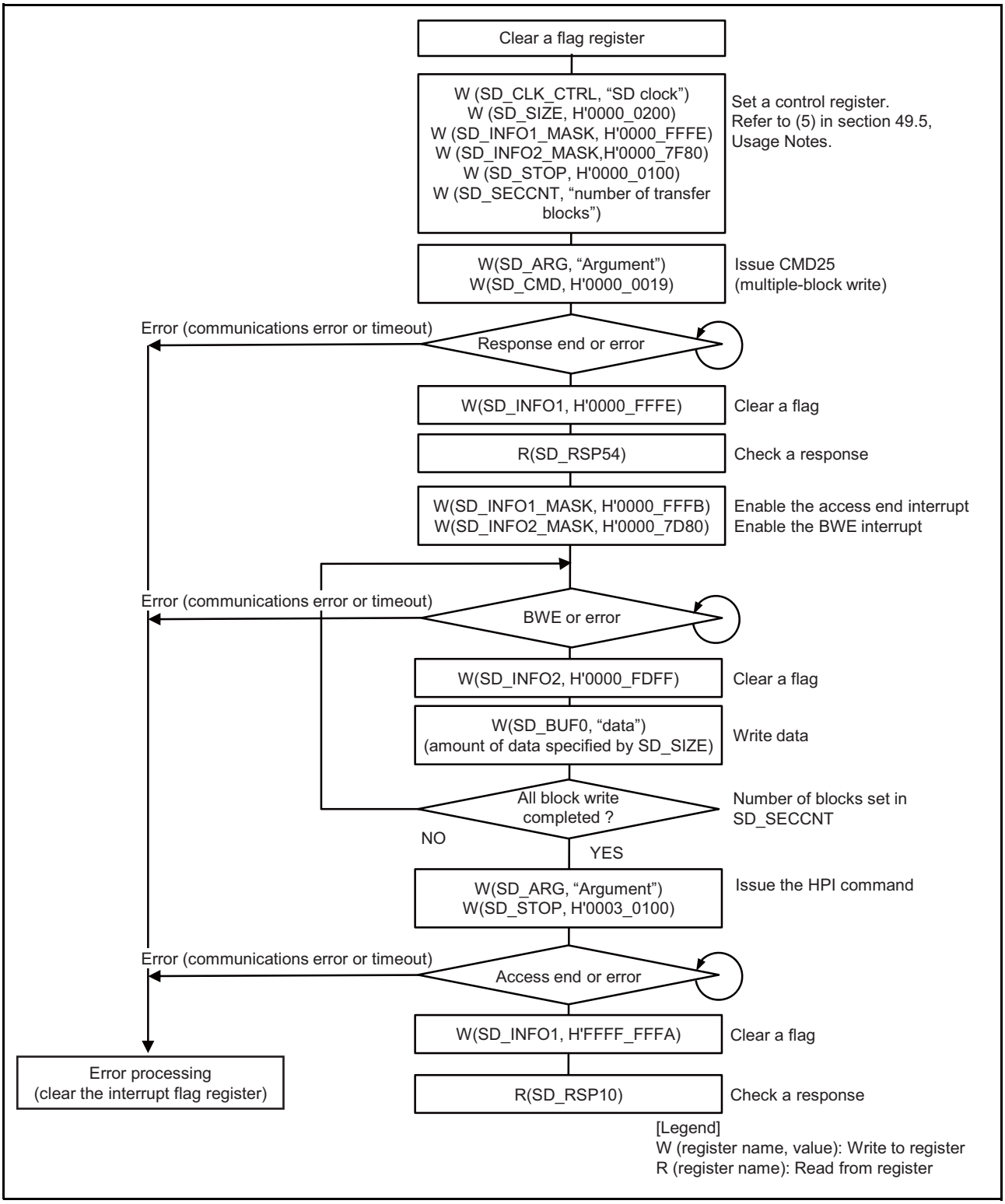


Figure 49.29 Example of the High-Priority Interrupt (at Multiple Block Write) Flowchart (b)

(2) Operation for HPI at Multiple Block Transfer

The operation of HPI at the multiple block write is described below.

(a) When not all the data has been written to SD_BUF

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD25)
   Set CMD25 Argument in SD_ARG and write H'0000 0019 to SD_CMD.
   Accordingly, CMD25 is issued, and the multiple block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) bit in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card
   Write H'0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000 7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card.
   Then, the CRC status and busy state are received from the SD card.
   However, a communications error or timeout may be generated if data are being transmitted while writing to SD_BUF0 is in progress.
6. Command issue (CMD12)
   Write H'FFFE FFFF to SD_INFO1_MASK to enable the HPIRES interrupt.
   Write H'0002 0101 to SD_STOP, which causes CMD12 to be issued.
7. Response check
   When the response is received, the HPIRES bit in SD_INFO1 is set to 1 to generate an interrupt. Clear the HPIRES bit to 0 to read the response from SD_RSP10.
8. HPI command issue (CMD25)
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
9. Operation complete
   When reception of the response to the HPI command is completed and the busy state is released, the INFO2 and INFO0 bits in SD_INFO1 are set to 1 to generate an interrupt. Clear the INFO2 and INFO0 bits to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

(b) When all the data has been written to SD_BUF

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD25)
   Set CMD25 Argument in SD_ARG and write H'0000 0019 to SD_CMD.
   Accordingly, CMD25 is issued, and the multiple block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) bit in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card
   Write H'0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000 7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card.
   Then, the CRC status and busy state are received from the SD card. Doing this repeats transfer of the number of blocks set in SD_SECCNT.
   However, a communications error or timeout may be generated if data are being transmitted while writing to SD_BUF0 is in progress.
6. HPI command issue
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
7. Operation complete
   When reception of the response to the HPI command is completed and the busy state is released, the INFO2 and INFO0 bits in SD_INFO1 are set to 1 to generate an interrupt. Clear the INFO2 and INFO0 bits to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

49.4.14 Example of SD_CMD Register Setting

Table 49.8 lists the example of SD_CMD (SD interface) register setting.

Table 49.8 Example of SD_CMD Register Setting (SD)

Type	Command	Example of SD_CMD Register Setting	Remark
CMD	CMD0	H'0000 0000	
	CMD2	H'0000 0002	
	CMD3	H'0000 0003	
	CMD4	H'0000 0004	
	CMD5	H'0000 0705 or H'0000 0005	
	CMD6	H'0000 1C06 or H'0000 0006	
	CMD7	H'0000 0007	When the card is placed in the deselected state, the response timeout flag will be set since there is no response.
	CMD8	H'0000 0408 or H'0000 0008	
	CMD9	H'0000 0009	
	CMD10	H'0000 000A	
	CMD11	H'0000 040B or H'0000 000B	
	CMD12	H'0000 000C	
	CMD13	H'0000 000D	
	CMD15	H'0000 000F	
	CMD16	H'0000 0010	
	CMD17	H'0000 0011	
	CMD18	H'0000 0012	With auto CMD12 enabled (other than in SDR104 mode)
		H'0000 7C12	With auto CMD12 disabled (only in SDR104 mode)
	CMD19	H'0000 1C13 or H'0000 0013	Setting prohibited in DDR50 mode
	CMD20	H'0000 0514 or H'0000 0014	
	CMD23	H'0000 0417 or H'0000 0017	
	CMD24	H'0000 0018	
	CMD25	H'0000 0019	With auto CMD12 enabled (other than in SDR104 mode)
		H'0000 6C19	With auto CMD12 disabled (only in SDR104 mode)
	CMD27	H'0000 001B	
	CMD28	H'0000 001C	
	CMD29	H'0000 001D	
	CMD30	H'0000 001E	
	CMD32	H'0000 0020	
	CMD33	H'0000 0021	
	CMD38	H'0000 0026	
	CMD42	H'0000 002A	
	CMD48	H'0000 1C30	
	CMD49	H'0000 0C31	
	CMD52	H'0000 0434 or H'0000 0034	

Table 49.8 Example of SD_CMD Register Setting (SD)

Type	Command	Example of SD_CMD Register Setting	Remark
CMD	CMD53	H'0000 1C35	Single read
		H'0000 0C35	Single write
		H'0000 7C35	Multiple read
		H'0000 6C35	Multiple write
		H'0000 0035	The value on the left can be set irrespective of whether single or multi. However, the CF39 bit in SD_ARG must be set as follows. Read: 0 Write: 1
	CMD55	H'0000 0037	
	CMD56	H'0000 0038	
	CMD58	H'0000 7C3A	
	CMD59	H'0000 6C3B	
ACMD	ACMD6	H'0000 0046	
	ACMD13	H'0000 004D	
	ACMD22	H'0000 0056	
	ACMD23	H'0000 0057	
	ACMD41	H'0000 0069	
	ACMD42	H'0000 006A	
	ACMD51	H'0000 0073	

Table 49.9 lists the example of SD_CMD (MMC interface) register setting.

Table 49.9 Example of SD_CMD Register Setting (MMC)

Type	Command	Example of SD_CMD Register Setting	Remark
CMD	CMD0	H'0000 0000	
	CMD1	H'0000 0701	
	CMD2	H'0000 0002	
	CMD3	H'0000 0003	
	CMD4	H'0000 0004	
	CMD5	H'0000 0505	
	CMD6	H'0000 0506	In the response busy state
		H'0000 0406	Not in the response busy state
	CMD7	H'0000 0007	When the card is placed in the deselected state, the response timeout flag will be set since there is no response.
	CMD8	H'0000 1C08	
	CMD9	H'0000 0009	
	CMD10	H'0000 000A	
	CMD12	H'0000 000C	
	CMD13	H'0000 000D	
	CMD14	H'0000 1C0E	SDIF_MODE must be set to H'0100 (with CRC16 disabled).
	CMD15	H'0000 000F	
	CMD16	H'0000 0010	
	CMD17	H'0000 0011	
	CMD18	H'0000 7C12	Pre-defined
	CMD19	H'0000 0C13	SDIF_MODE must be set to H'0100 (with CRC16 disabled).
	CMD21	H'0000 1C15	Setting prohibited in DDR mode
	CMD23	H'0000 0017	
	CMD24	H'0000 0018	
	CMD25	H'0000 6C19	Pre-defined
	CMD26	H'0000 0C1A	
	CMD27	H'0000 001B	
	CMD28	H'0000 001C	
	CMD29	H'0000 001D	
	CMD30	H'0000 001E	
	CMD31	H'0000 1C1F	
	CMD35	H'0000 0423	
	CMD36	H'0000 0424	
	CMD38	H'0000 0026	
	CMD39	H'0000 0427	
	CMD40	H'0000 0428	
	CMD42	H'0000 002A	
	CMD49	H'0000 0C31	
	CMD53	H'0000 7C35	
	CMD54	H'0000 6C36	
	CMD55	H'0000 0037	
	CMD56	H'0000 0038	

49.5 Usage Notes

(1) SD_BUF Illegal Write Access

When writing data to SD_BUF0 after the single block write or multi block write command is issued, the data of the size specified by SD_SIZE must be written to.
If the data of the size which exceeds the size specified by SD_SIZE is written to, the ERR4 bit in SD_INFO2 is set to 1. In addition, the data written to SD_BUF0 may not be transmitted and it causes the SCLKDIVEN bit in SD_INFO2 to hold the value of 0. In such cases, clearing the SDRST bit in SOFT_RST to 0 and then restoring its value to 1 clears the SCLKDIVEN bit to 1.
However, for the single byte (in the case of 16- or 32-bit access) or three bytes (in the case of 32-bit access) when the number of bytes setting in SD_SIZE is odd, or the fraction of bytes when the number of bytes setting in SD_SIZE is even (in the case of 32-bit access), since the portion of dummy data writing is regarded as excess data and ignored, it is not within the scope of the above description (the fraction of bytes: the two bytes that are not in a four-byte unit).

(2) Block Number Limitation for Multiple Block Read

When performing a multiple block read of one or two blocks, depending on the timing with which the response register is read, the response value may not be read properly. This must be avoided by either of the following countermeasures.

1) When receiving one or two blocks of data, use single block reading.
2) Read the response to CMD18 from SD_RSP54.

<Mechanism of incorrect reading>

Figure 49.30 shows the processing flows of SD/MMC host interface (hardware) operation and software operation when a multiple block read is performed on two blocks. As shown in the incorrect operation of Figure 49.30, when an interrupt is generated on reception of the CMD18 response and the timing with which the SD card response register (SD_RSP10) is read by the interrupt is delayed, the data during the CMD12 response reception or the CMD12 response may be read. In the case of a multiple block read of three or more blocks, CMD12 is not issued until the block of data has been read, so this problem does not arise. Furthermore, in the case of a multiple block write, since the CMD25 response is read before the block of data is sent, the problem does not arise.

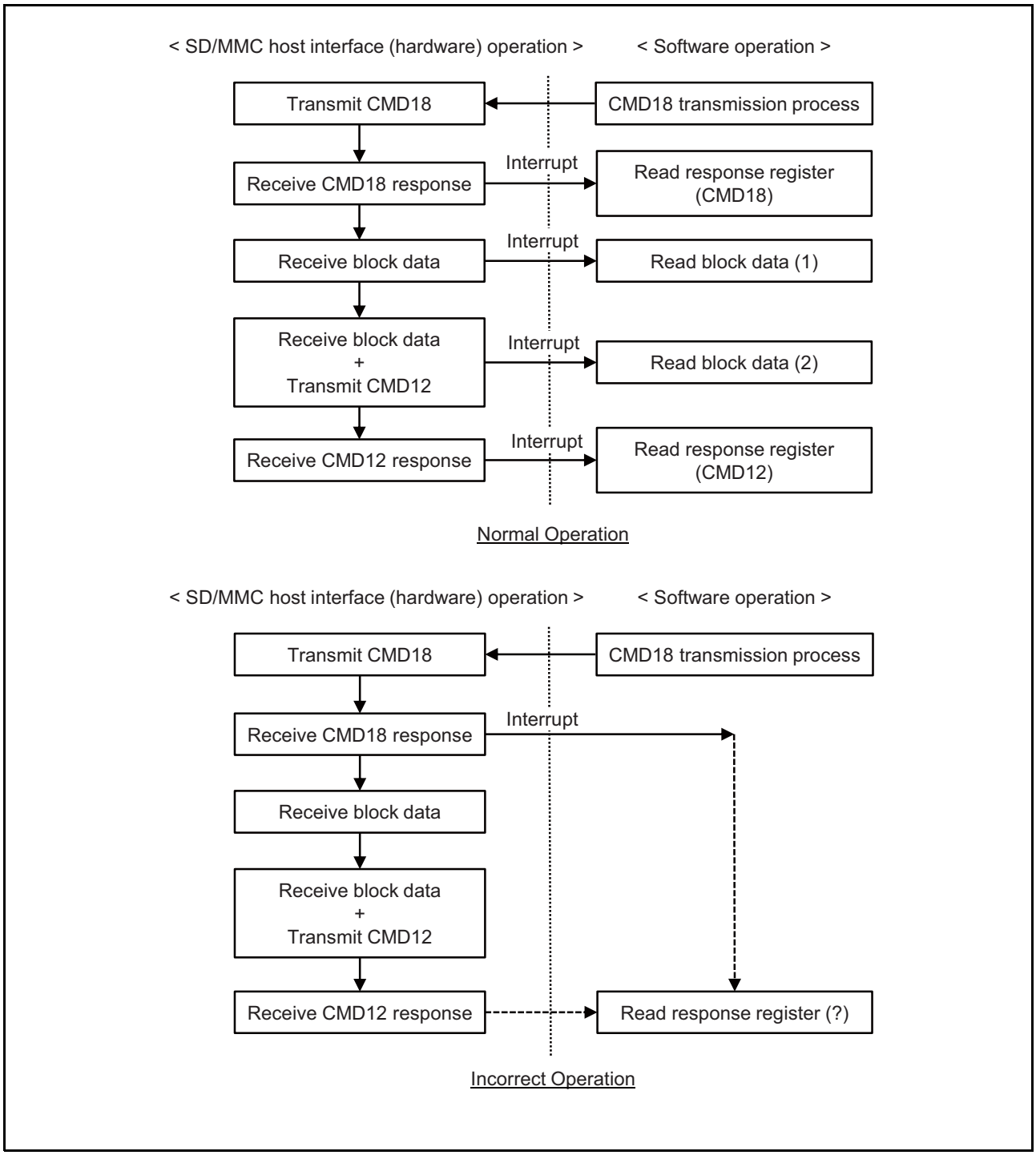


Figure 49.30 Flowcharts for Multiple Block Read Operation (Two Blocks)

(3) Automatic Control of SDCLK Output

In the SD Card standard, 74 cycles of SDCLK must be output before initialization of the card. For this reason, use automatic control of SDCLK output after 74 cycles of SDCLK have been output. Furthermore, if automatic control of SDCLK output was in use, SDCLK output is stopped on completion of the sequence for a communications error or timeout. Thus, in cases where state transitions within the SD card are necessary and so on after completion of the sequence, release automatic control of SDCLK output and restart supply of SDCLK to the SD card.

(4) Control of the C52PUB Setting for Multiple Block Write

If the C52PUB bit in SDIO_MODE is set to 1 during a sequence of multiple block write due to CMD53, CMD52 is not issued until SD_BUF becomes empty. For this reason, set the C52PUB bit after suspending writing to SD_BUF by following the appropriate procedure below.

- When DMA transfer is not in use

1. Before setting the C52PUB bit, suspend writing to SD_BUF by making the setting in SD_INFO2 to disable BWE interrupts.
2. Set the C52PUB bit in SDIO_MODE to 1 (so that CMD52 is issued when SD_BUF becomes empty).
3. After the INFO0 interrupt processing in SD_INFO1 due to the issuing of CMD52 has been completed, restart writing to SD_BUF by making the setting in SD_INFO2 to enable BWE interrupts.

- When DMA transfer is in use

1. Every time DMA transfer of the value set in SD_SIZE × n blocks (where n = 1, 2, ...) proceeds, suspend writing to SD_BUF by DMA transfer before the C52PUB bit is set.
2. Set the C52PUB bit in SDIO_MODE to 1 (so that CMD52 is issued when SD_BUF becomes empty).
3. After the INFO0 interrupt processing in SD_INFO1 due to the issuing of CMD52 has been completed, restart writing to SD_BUF by DMA transfer.

(5) Notes on SD_CLK_CTRL Register Settings

When the SCLKDIVEN bit in SD_INFO2 is 0, SD_CLK_CTRL cannot be written to. Before writing to SD_CLK_CTRL, be sure to check that the SCLKDIVEN bit in SD_INFO2 is 1.

(6) Restrictions on specifications

1. The SDIO suspend/resume is not supported.
2. The SPI bus is not supported.
3. The shared bus and 8-bit SD bus for embedded SDIO are not supported.
4. The stream transfer for MMC cards is not supported.

(7) STP bit setting during multiple block read

During execution of multiple block read with automatic CMD12 execution by setting the SEC bit in SD_STOP to 1, even if the STP bit in SD_STOP is set to 1 to forcibly stop the execution, the command sequence may not stop depending on the timing of setting the STP bit.
To avoid this, when setting the STP bit in SD_STOP to 1 during multiple block transfer, clear the SEC bit in SD_STOP to 0 at the same time. (Even when the SCLKDIVEN bit in SD_INFO2 is 0, change the SEC bit from 1 to 0.)
When the command sequence has not stopped because the SEC bit was not cleared to 0, the command sequence can be stopped by clearing the SDRST bit in SOFT_RST to 0.
When forcibly terminating the CMD53 multiple block transfer through the IOABT bit in SDIO_MODE, be sure to leave the SEC bit in SD_STOP as 1.

49.6 Sampling Clock Controller (SCC)

49.6.1 Features

This module controls a sampling clock (hereafter referred to as the SCC sampling clock) that is used for SD UHS-I/ SDR104 and MMC HS200. When this module is used with the LSI, SD UHS-I/SDR104 and MMC HS200 can be supported.

49.6.2 SCC Block Diagram

Figure 49.31 shows a block diagram of the sampling clock controller.

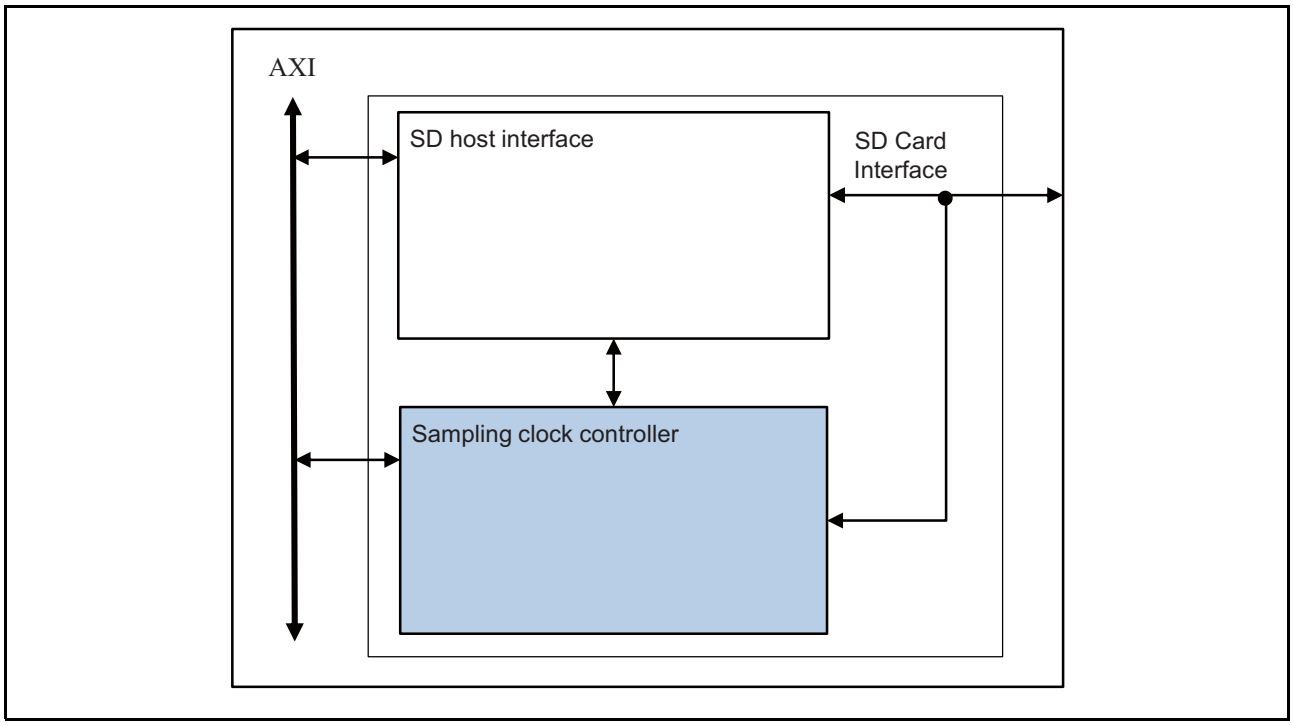


Figure 49.31 Block Diagram of the Sampling Clock Controller

49.6.3 SCC Register Configuration

Table 49.10 shows the SCC registers. The base addresses of each channel are as follows.

Channel 0: H'E8227000
Channel 1: H'E8229000

Table 49.10 Register Configuration

Name	Abbreviation	Address ([9:0])	Access Size
Initial setting register	SCC_DTCNTL	H'000	32
Sampling clock position setting register	SCC_TAPSET	H'008	32
Hardware adjustment register 1	SCC_DT2FF	H'010	32
Sampling clock selection register	SCC_CKSEL	H'018	32
Sampling clock position correction register	SCC_RVSCNTL	H'020	32
Sampling clock position correction request register	SCC_RVSREQ	H'028	32
Sampling data comparison register	SCC_SMPCMP	H'030	32

49.7 SCC Register Descriptions

49.7.1 Initial Setting Register (SCC_DTCNTL)

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
23 to 16	TAPNUM7 to TAPNUM0	H'08	R/W	When bits DIV7 to DIV0 in the SD_CLK_CTRL register are H'FF (1: 1 mode), set these bits to H'08.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	TAPEN	0	R/W	SCC Sampling Clock Operation Enable 0: SCC sampling clock operation is disabled. 1: SCC sampling clock operation is enabled.

49.7.2 Sampling Clock Position Setting Register (SCC_TAPSET)

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7 to 0	TAPSET7 to TAPSET0	H'00	R/W	SCC Sampling Clock Position ▪ Set the tuning result in the range from 0 to TAPNUM-1.

49.7.3 Sampling Clock Selection Register (SCC_CKSEL)

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	DTSEL	0	R/W	Sampling Clock Selection 0: An SCC sampling clock is not used (for other than SDR104 and HS200). 1: An SCC sampling clock is used (for SDR104 or HS200). • For SDR104 or HS200, set DTSEL to 1. DIV[7:0] in the SD_CLK_CTRL register to H'FF (1:1 mode). • When this bit is switched, stop the SD clock output from the SD/MMC host interface (set SCLKEN in SD_CLK_CTRL to 0).

49.7.4 Sampling Clock Position Correction Register (SCC_RVSCNTL)

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 8	TAPSEL7 to TAPSEL0	H'00	R	SCC Sampling Clock Position Display • Displays the SCC sampling clock position selected by hardware. • After RVSEN has been set to 1, the value may differ from that of TAPSET.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1	RVSW	0	R/W	This bit is read as 0. The write value should be 1.
0	RVSEN	0	R/W	SCC Sampling Clock Position Correction Enable 0: SCC sampling clock position correction is disabled. 1: SCC sampling clock position correction is enabled. • When RVSEN is set to 1 after tuning has been performed, this module corrects the SCC sampling clock position each time of a command sequence of the SD/MMC host interface. • However, when RVSERR is 1, this module does not correct the SCC sampling clock position. • While tuning is being performed, set RVSEN to 0.

49.7.5 Sampling Clock Position Correction Request Register (SCC_RVSREQ)

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2	RVSERR	0	R/W	SCC Sampling Clock Position Correction Error 0: There is no correction error. 1: There is a correction error. • If this bit is set to 1 after a command sequence, write 0 to this bit and perform tuning again. • Ignore this bit while tuning is being performed. • Writing 1 to this bit is disabled and writing 0 to this bit is only enabled.
1	REQTAPUP	0	R/W	SCC Sampling Clock Position Positive Direction Correction Request 0: There is no correction request. 1: There is a correction request. • If this bit is set to 1 after a command sequence, write 0 to this bit and rewrite TAPSET in the positive direction (when TAPSEL = TAPNUM-1, set 0 to TAPSET). • When RVSEN is 1, this bit is disabled (this bit is not set to 1). • Writing 1 to this bit is disabled and writing 0 to this bit is only enabled.
0	REQTAPDWN	0	R/W	SCC Sampling Clock Position Negative Direction Correction Request 0: There is no correction request. 1: There is a correction request. • If this bit is set to 1 after a command sequence, write 0 to this bit and rewrite TAPSET in the negative direction (when TAPSEL = 0, set TAPNUM-1 to TAPSET). • When RVSEN is 1, this bit is disabled (this bit is not set to 1). • Writing 1 to this bit is disabled and writing 0 to this bit is only enabled.

49.7.6 Hardware Adjustment Register 1 (SCC_DT2FF)

This register makes a setting that SD_DATA, which has been fetched by the sampling clock at each TAP position, is used in the appropriate timing.

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 8	DT2NESET7 to DT2NESET0	H'05	R/W	Hardware Adjustment 1 When the sampling clock controller is to be used, set these bits to H'03.
7 to 0	DT2NSSET7 to DT2NSSET0	H'02	R/W	Hardware Adjustment 2 When the sampling clock controller is to be used, set these bits to H'00.

49.7.7 Sampling data comparison register (SCC_SMPCMP)

Data comparison register indicates the result of the comparison of the sampling data. The subject of the comparison is the before and the behind TAP.

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
24 to 16	CMPNGU	All 0	R/W	Comparison of sampling data with the previous TAP Clock. Bit 16-23 is the comparison result of data 0-7. Bit 24 is the comparison result of CMD. 0: Match 1: Mismatch < Clear conditions.> The start of the command sequence Write to SCC_TAPSET register
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8 to 0	CMPNGD	All 0	R/W	Comparison of sampling data with the after TAP Clock. Bit 0-7 is the comparison result of data 0-7. Bit 8 is the comparison result of CMD. 0: Match 1: Mismatch < Clear conditions.> The start of the command sequence Write to SCC_TAPSET register

49.8 Usage Example of SCC

49.8.1 Tuning

SCC is tuned by using operation of single-block reading.
As shown in Figure 49.32, check whether the single-block read command normally ends when the sampling clock position is changed from 0 to TAPNUM-1 and save the result. After checking, confirm that there exists the range which has three or more continuous normal ends (OK). Then, the median value within the continuous range is determined as the final adjustment value.
Figure 49.33 and Table 49.11 show the detailed tuning flow and the method how to select the sampling clock position (example when TAPNUM = 8).

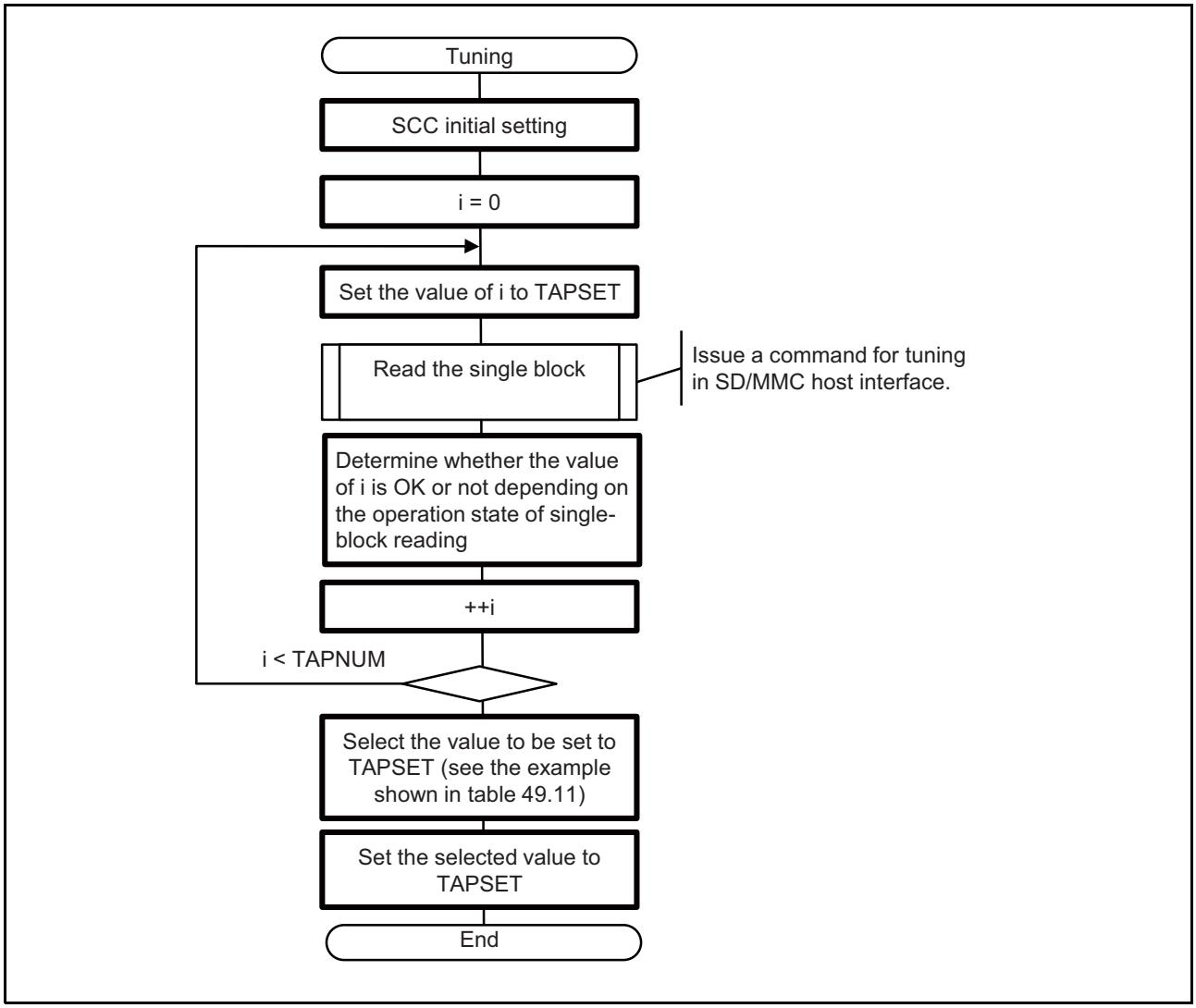


Figure 49.32 Example of Tuning Flow (Outline)

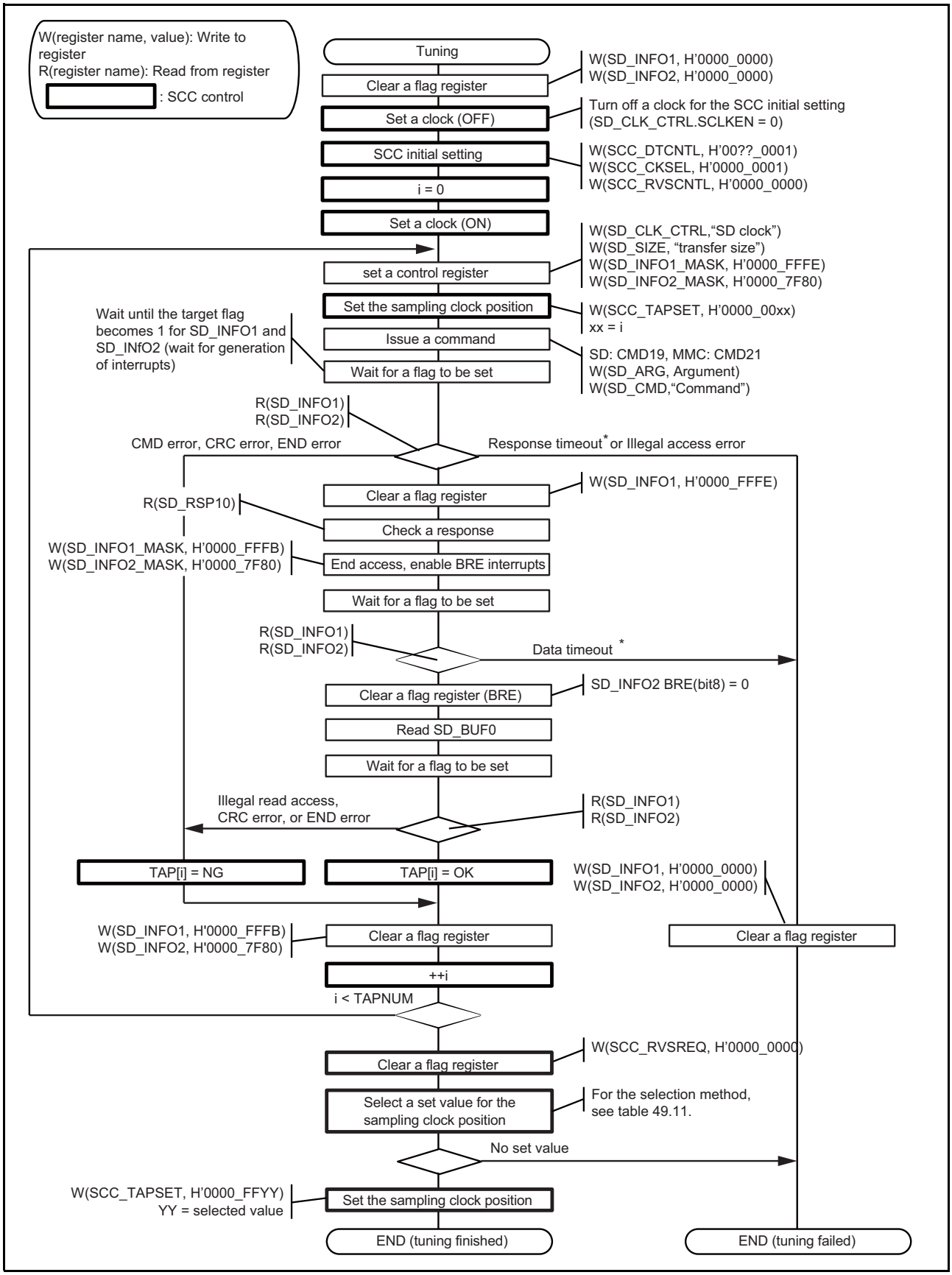


Figure 49.33 Example of Tuning Flow (Detailed)

*: If all of the TAP[i] has become "NG".

Table 49.11 Example of the Method How to Select the Sampling Clock Position (when TAPNUM = 8)

Item	i	Case 1	Case 2	Case 3	Case 4	Case 5*1
TAP[i]	0	NG	OK	NG	OK	OK
	1	OK	OK	NG	NG	OK
	2	OK	NG	OK	NG	OK
	3	OK(←)	NG	OK	NG	OK
	4	OK	NG	NG	NG	OK
	5	OK	OK	NG	OK	OK
	6	NG	OK	NG	OK(←)	OK
Max. value→	7	NG	OK(←)	NG	OK	OK
	(0)	NG	OK	NG	OK	OK
	(1)	OK	OK	NG	NG	OK
	(2)	OK	NG	OK	NG	OK
	(3)	OK	NG	OK	NG	OK
	(4)	OK	NG	NG	NG	OK
	(5)	OK	OK	NG	OK	OK
	(6)	NG	OK	NG	OK	OK
	(7)	NG	OK	NG	OK	OK
Selected value		i = 3	i = 7	Fail	i = 6 or 7	i = 0 to 7

Symbols: (←): Example of selection, (x): repeated display of index x of TAP[x]

*1 If all of the TAP [i] is OK, the sampling clock position is selected by identifying the change point of data.

Change point of the data can be found in the value of SCC_SMPCMP register. Usage example is section 49.8.3, Change point of the input data.

(a) The sampling clock position is selected by considering a margin in the range which has three or more continuous 'TAP[i] = OK'.

(b) The sampling clock position is repeated from 0 after the maximum value (TAPNUM-1). In case 2 above, that position is continued in the order of 5→6→7→0→1.

49.8.2 Sampling Clock Position Correction after Tuning

After tuning, correction of the sampling clock position may be required when a command is issued.
There are manual and automatic correction methods. After a command sequence, if the CMD, CRC, END error or time out occurs or the correction error occurs, tuning will be performed again. The following shows examples of manual and automatic correction methods.

(1) Manual correction of the sampling clock position

Figure 49.34 shows the flow of manual correction of the sampling clock position. Table 49.12 shows set values determined when correction is required (when TAPNUM = 8).

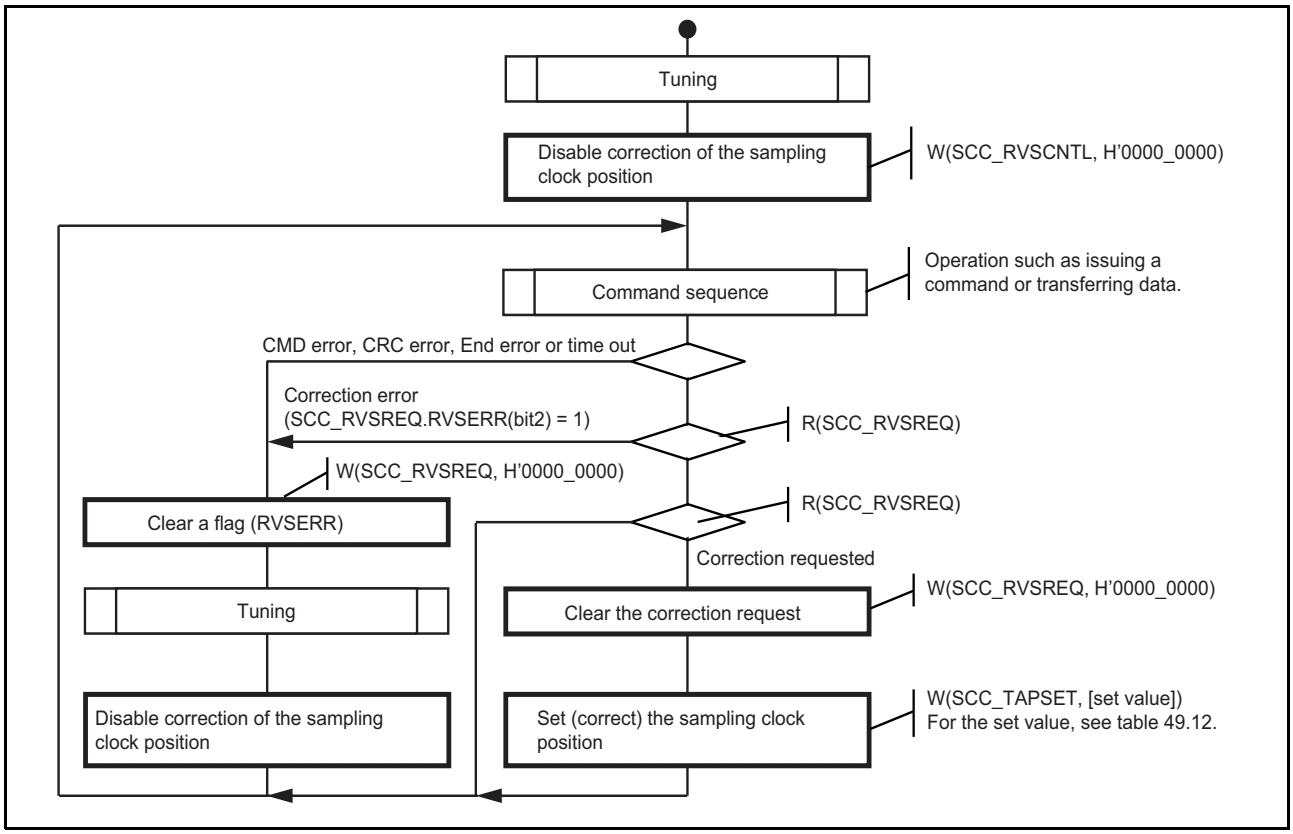


Figure 49.34 Flow of Manual Correction of the Sampling Clock Position (Example)

Table 49.12 Set Values for TAPSET when Correction is Required (when TAPNUM = 8)

No.	Current value of TAPSET	Value set to TAPSET when REQTAPUP = 1	Value set to TAPSET when REQTAPDWN = 1
1	0	1	7
2	1	2	0
3	2	3	1
4	3	4	2
5	4	5	3
6	5	6	4
7	6	7	5
8	7	0	6

Note: As is the case in the tuning selection method, the sampling clock position is 0 after the maximum value (TAPNUM-1).

(2) Automatic correction of the sampling clock position

Figure 49.35 shows the flow of automatic correction of the sampling clock position.

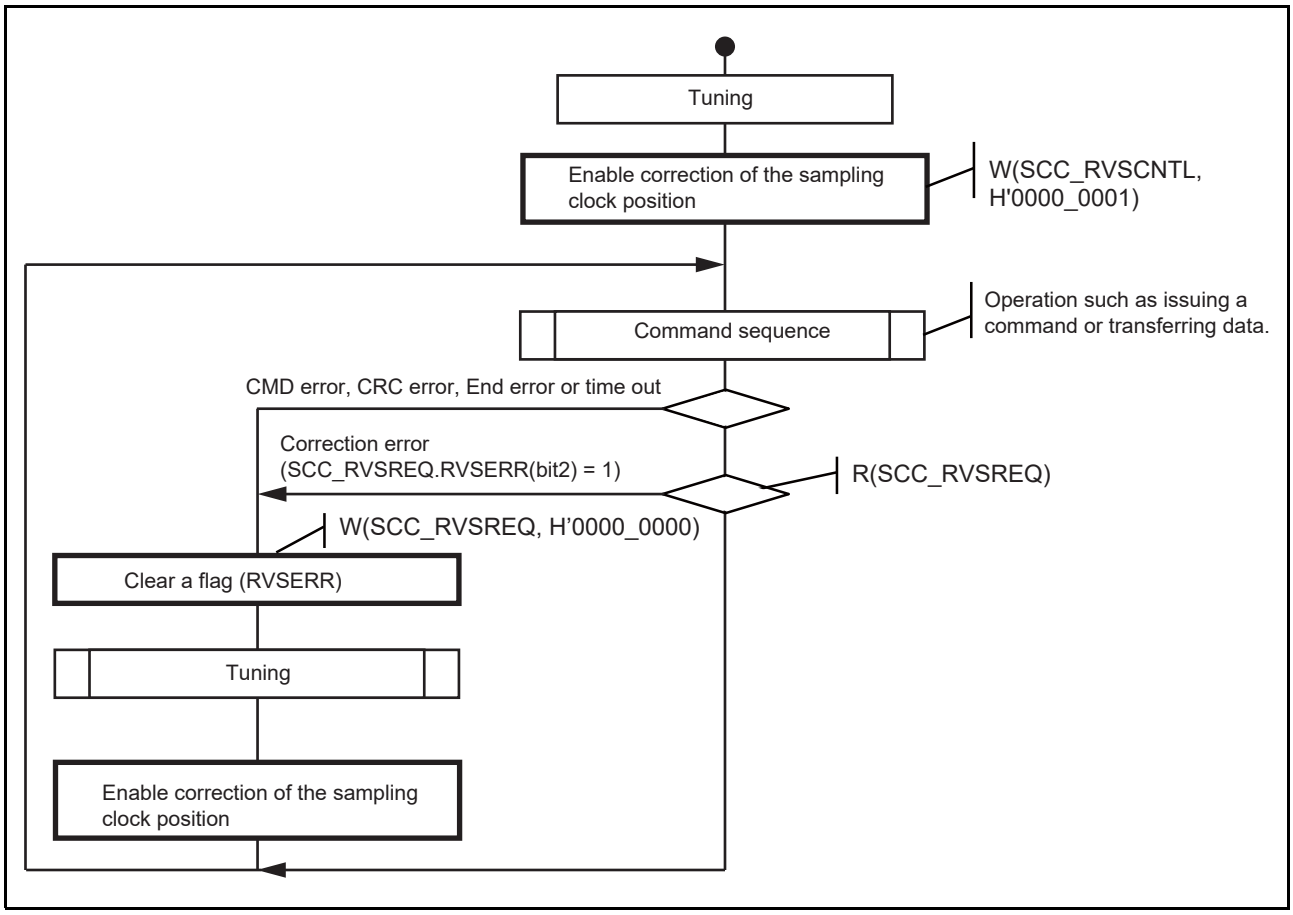


Figure 49.35 Flow of Automatic Correction of the Sampling Clock Position (Example)

49.8.3 Change point of the input data

Tuning is capture the data by the TAP clock selected. However, also captures the data by the previous TAP clock and the behind the TAP clock at the same time. This result is reflected in the sampling data comparison register (SCC_SMPCMP). Point of mismatch before and after the selected TAP clock is the changing point of the data. In this example, it is desirable to set as TAP6 or TAP7.

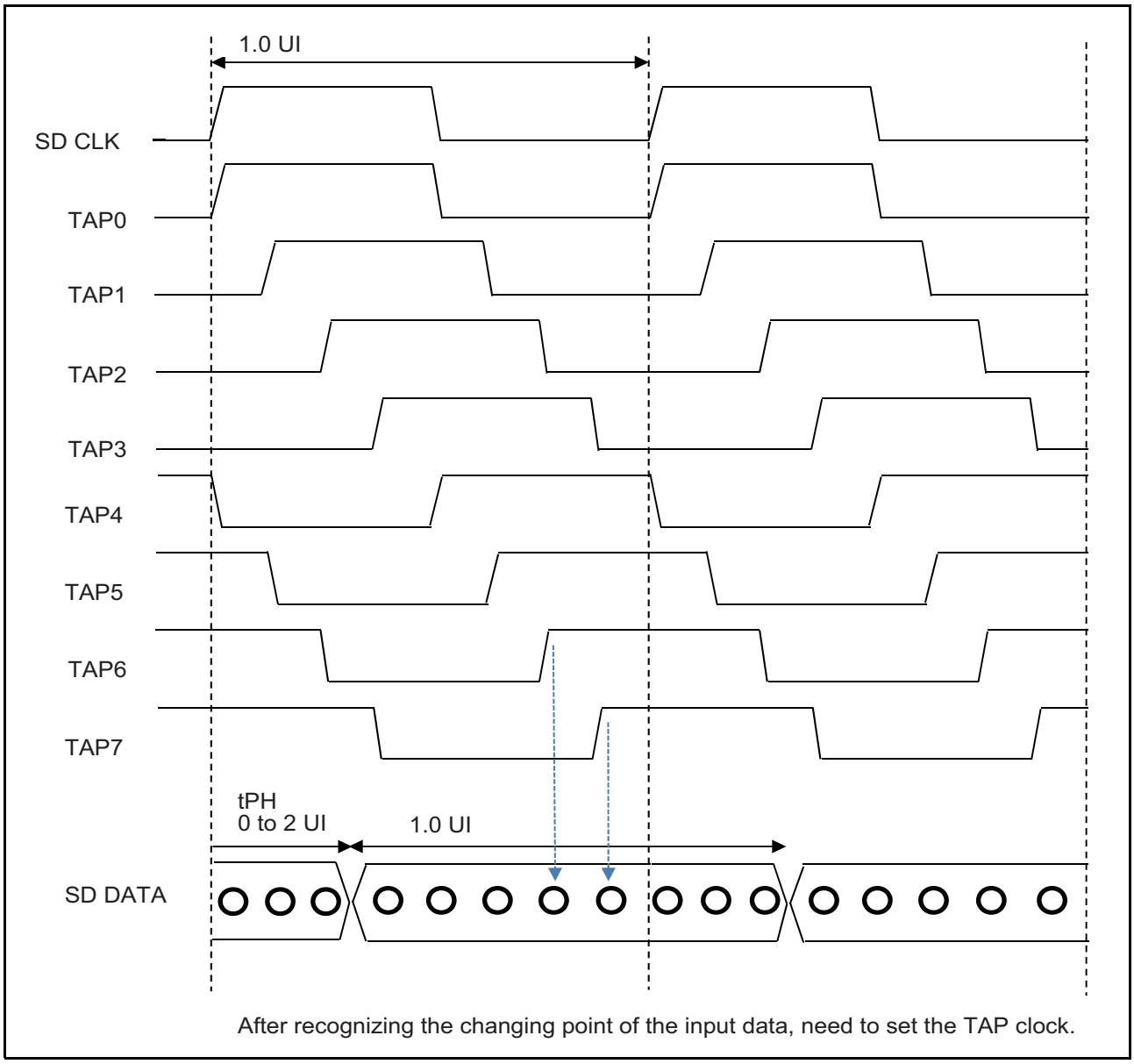


Figure 49.36 Example, All taps is OK.

Figure 49.36 is shown change point of data between TAP2 and TAP3. Change point of the data can be confirmed by sampling data comparison register (SCC_SMPCMP). When the tuning of TAP2 or TAP3, CMPNGU bit of sampling data comparison register indicates a mismatch. As the width of the input data is 1 (UI), select TAP6 or TAP7 which is the median of next TAP3 from TAP3.

50. On-Chip RAM

This LSI has an on-chip large-capacity RAM for display area and work area (128 Kbytes of this RAM are shared with the on-chip data retention RAM) and an on-chip data retention RAM, which can retain data in deep standby mode. These memory units can be used to store instructions or data.
The operation and write access to the large-capacity RAM (including on-chip data retention RAM) can be enabled or disabled through the RAM enable bits and RAM write enable bits.
The on-chip data retention RAM is assigned to page 0 in the on-chip large-capacity RAM. Retention or non-retention of data by the on-chip data retention RAM in deep standby mode is selectable on a per-page basis.

50.1 Features

- Page
  - The on-chip large-capacity RAM consists of five pages.
  - The on-chip data retention RAM consists of four pages. Page 0 has 16-Kbytes, page 1 has 16-Kbytes, page 2 has 32-Kbytes, and page 3 has 64-Kbytes.
- Memory map
  The on-chip RAM is located in the address spaces shown in Table 50.1 and Table 50.2. Pages 0 and 1 of the large-capacity RAM each have a capacity of 512 Kbytes, and pages 2 to 4 of the large-capacity RAM has a capacity of 1 Mbyte.

Table 50.1 Address Spaces of On-Chip Large-Capacity RAM

Page	Address
Page 0 (512KB)	H'80000000 to H'8007FFFF
Page 1 (512KB)	H'80080000 to H'800FFFFF
Page 2 (1024KB)	H'80100000 to H'801FFFFF
Page 3 (1024KB)	H'80200000 to H'802FFFFF
Page 4 (1024KB)	H'80300000 to H'803FFFFF

Table 50.2 Address Spaces of On-Chip Data Retention RAM

Page	Address
Page 0 (16KB)	H'80000000 to H'80003FFF
Page 1 (16KB)	H'80004000 to H'80007FFF
Page 2 (32KB)	H'80008000 to H'8000FFFF
Page 3 (64KB)	H'80010000 to H'8001FFFF

- Ports
  Each page of the on-chip large-capacity RAM has one read and write port and is connected to the AXI bus. The on-chip RAM for data retention, which has a port independent of the port in page 0, is shared with the read and write port in four pages.
- Method of arbitration
  When the same port of the on-chip large-capacity RAM is accessed from different masters simultaneously, the AXI bus performs arbitration in round-robin mode.
- Number of access cycles
  The number of cycles for access to read or write is one cycle of Bφ.

50.2 Usage Notes

50.2.1 Page Conflict

When the same page of the on-chip large-capacity RAM is accessed from different masters simultaneously, a conflict on the page occurs. Although each access is completed correctly, this kind of conflict degrades the memory access speed. Therefore, it is advisable to provide software measures to prevent such conflicts as far as possible. For example, no conflict will arise if different pages are accessed by each master.

50.2.2 Data Retention

Data in the large-capacity RAM (including on-chip data retention RAM) are retained in the states other than power-on reset and deep standby mode. In power-on reset and deep standby mode, these RAMs operate as described below.

(1) Power-on Reset

(a) On-Chip Large-Capacity RAM (Excluding On-Chip Data Retention RAM)

Data are retained on a power-on reset by disabling the setting of either the VRAME or VRAMWE bit.
Data are not retained when the setting of the VRAME and VRAMWE bits are both enabled.

(b) On-Chip Data Retention RAM

Data are retained on a power-on reset by disabling the setting of any of the VRAME, VRAMWE, or RRAMWE, excluding the case that deep standby mode is canceled by power-on reset.
Data are not retained when the setting of the VRAME, VRAMWE and RRAMWE bits are all enabled.

(2) Deep Standby Mode

(a) On-Chip Large-Capacity RAM (Excluding On-Chip Data Retention RAM)

Data are not retained.

(b) On-Chip Data Retention RAM

Data are retained in deep standby mode by enabling the setting of the RRAMKP bit, excluding the case that deep standby mode is canceled by power-on reset. In the case that deep standby mode is canceled by interrupt or pins for cancelling, power-on reset exception handling is executed, but the data are retained.

51. GPIO

51.1 Features

This LSI chip has 61 pins exclusively for use with single functions. It also has 22 ports: P0 to PL, and JP0.
The pins of the ports can be used as general-purpose input/output pins, each of which may also be assigned one of multiple other selectable functions.

Switching of the following pin functions by controlling the respective registers is possible for the specific function ports.

- Driving ability of the CKIO pin by controlling PCKIO
- Driving ability and operating voltage of the pins for the SPI multi I/O bus controller by controlling PSPIBSC and PPOC
- Selection of the HyperBus™*1 controller or Octa*2 memory controller pin by controlling PHMOM0
- Driving ability and operating voltage of the pins for the SD/MMC host interfaces by controlling PSDMMC0 to PSDMMC2 and PPOC

Note 1. HyperBus, HyperFlash, and HyperRAM are trademarks of Cypress Semiconductor Corporation.
Note 2. OctaFlash and OctaRAM are trademarks of Macronix International Co., Ltd.

The general-purpose input/output port pins are multiplexed with the peripheral on-chip module functions. The functions multiplexed on these pins can be selected as desired by setting of the registers.
Each port has the port direction register (PDR) that selects non-use, input, or output, the port output data register (PODR) that holds data for output, the port input register (PIDR) that indicates the pin states, and the port mode register (PMR) that specifies the pin function of each port.
Each port has the control register (PFS) that is used to select the input/output function and interrupt pin from the multiplexed port pins, and to assign the function to the selected pin.
The configuration of the I/O ports differs depending on the package. Table 51.1 shows the specifications of I/O ports, and Table 51.2 lists the port functions.

Table 51.1 Specifications of I/O Ports

		Package		Package		Package	
	Pin Name	324 Pins	Number of Pins	272, 256 Pins	Number of Pins	176 Pins	Number of Pins
Alternative pin	P0_0 to P0_6	P0_0 to P0_6	7	—	0	—	0
	P1_0 to P1_4	P1_0 to P1_4	5	—	0	—	0
	P2_0 to P2_3	P2_0 to P2_3	4	—	0	—	0
	P3_0 to P3_5	P3_0 to P3_5	6	P3_0 to P3_5	6	P3_1 to P3_5	5
	P4_0 to P4_7	P4_0 to P4_7	8	P4_0 to P4_7	8	P4_0 to P4_7	8
	P5_0 to P5_7	P5_0 to P5_7	8	P5_0 to P5_7	8	P5_0 to P5_7	8
	P6_0 to P6_7	P6_0 to P6_7	8	P6_0 to P6_7	8	P6_0 to P6_3	4
	P7_0 to P7_7	P7_0 to P7_7	8	P7_2, P7_6, P7_7	3	—	0
	P8_0 to P8_7	P8_0 to P8_7	8	P8_0	1	—	0
	P9_0 to P9_7	P9_0 to P9_7	8	—	0	—	0
	PA_0 to PA_7	PA_0 to PA_7	8	PA_0 to PA_7	8	—	0
	PB_0 to PB_5	PB_0 to PB_5	6	PB_0 to PB_5	6	—	0
	PC_0 to PC_7	PC_0 to PC_7	8	PC_0 to PC_7	8	—	0
	PD_0 to PD_7	PD_0 to PD_7	8	PD_0 to PD_7	8	PD_0 to PD_7	8
	PE_0 to PE_6	PE_0 to PE_6	7	PE_0 to PE_6	7	PE_0 to PE_6	7
	PF_0 to PF_7	PF_0 to PF_7	8	PF_0 to PF_7	8	—	0

	Pin Name	Package 324 Pins	Number of Pins	Package 272, 256 Pins	Number of Pins	Package 176 Pins	Number of Pins
Alternative pin	PG_0 to PG_7	PG_0 to PG_7	8	PG_0 to PG_7	8	PG_0 to PG_7	8
	PH_0 to PH_6	PH_0 to PH_6	7	PH_0 to PH_6	7	PH_0, PH_1, PH_3, PH_4	4
	PJ_0 to PJ_7	PJ_0 to PJ_7	8	PJ_0 to PJ_7	8	PJ_0 to PJ_5	6
	PK_0 to PK_5	PK_0 to PK_5	6	PK_0 to PK_5	6	PK_0 to PK_4	5
	PL_0 to PL_4	PL_0 to PL_4	5	PL_0 to PL_4	5	PL_0 to PL_4	5
	JP0_0, JP0_1	JP0_0, JP0_1	2	JP0_0, JP0_1	2	JP0_0, JP0_1	2
Dedicated Pin	CKIO	CKIO	1	CKIO	1	CKIO	1
	RES#	RES#	1	RES#	1	RES#	1
	NMI	NMI	1	NMI	1	NMI	1
	USB_X1	USB_X1	1	USB_X1	1	USB_X1	1
	USB_X2	USB_X2	1	USB_X2	1	USB_X2	1
	EXTAL	EXTAL	1	EXTAL	1	EXTAL	1
	XTAL	XTAL	1	XTAL	1	XTAL	1
	RTC_X1	RTC_X1	1	RTC_X1	1	RTC_X1	1
	RTC_X2	RTC_X2	1	RTC_X2	1	RTC_X2	1
	AUDIO_X1	AUDIO_X1	1	AUDIO_X1	1	AUDIO_X1	1
	AUDIO_X2	AUDIO_X2	1	AUDIO_X2	1	AUDIO_X2	1
	TMS/SWDIO	TMS/SWDIO	1	TMS/SWDIO	1	TMS/SWDIO	1
	TCK/SWDCLK	TCK/SWDCLK	1	TCK/SWDCLK	1	TCK/SWDCLK	1
	TRST#	TRST#	1	TRST#	1	TRST#	1
	SD1_CLK	SD1_CLK	1	SD1_CLK	1	SD1_CLK	1
	SD1_CMD	SD1_CMD	1	SD1_CMD	1	SD1_CMD	1
	SD1_DAT0	SD1_DAT0	1	SD1_DAT0	1	SD1_DAT0	1
	SD1_DAT1	SD1_DAT1	1	SD1_DAT1	1	SD1_DAT1	1
	SD1_DAT2	SD1_DAT2	1	SD1_DAT2	1	SD1_DAT2	1
	SD1_DAT3	SD1_DAT3	1	SD1_DAT3	1	SD1_DAT3	1
	SD0_CLK	SD0_CLK	1	SD0_CLK	1	SD0_CLK	1
	SD0_CMD	SD0_CMD	1	SD0_CMD	1	SD0_CMD	1
	SD0_DAT0	SD0_DAT0	1	SD0_DAT0	1	SD0_DAT0	1
	SD0_DAT1	SD0_DAT1	1	SD0_DAT1	1	SD0_DAT1	1
	SD0_DAT2	SD0_DAT2	1	SD0_DAT2	1	SD0_DAT2	1
	SD0_DAT3	SD0_DAT3	1	SD0_DAT3	1	SD0_DAT3	1
	SD0_DAT4	SD0_DAT4	1	SD0_DAT4	1	SD0_DAT4	1
	SD0_DAT5	SD0_DAT5	1	SD0_DAT5	1	SD0_DAT5	1
	SD0_DAT6	SD0_DAT6	1	SD0_DAT6	1	SD0_DAT6	1
	SD0_DAT7	SD0_DAT7	1	SD0_DAT7	1	SD0_DAT7	1
	SD0_RST#	SD0_RST#	1	SD0_RST#	1	SD0_RST#	1
	HM_CK/OM_SCLK	HM_CK/OM_SCLK	1	HM_CK/OM_SCLK	1	HM_CK/OM_SCLK	1
	HM_CK#	HM_CK#	1	HM_CK#	1	HM_CK#	1
	HM_CS0#/ OM_CS0#	HM_CS0#/ OM_CS0#	1	HM_CS0#/ OM_CS0#	1	HM_CS0#/ OM_CS0#	1
	HM_CS1#/ OM_CS1#	HM_CS1#/ OM_CS1#	1	HM_CS1#/ OM_CS1#	1	HM_CS1#/ OM_CS1#	1
	HM_RWDS/ OM_DQS	HM_RWDS/ OM_DQS	1	HM_RWDS/ OM_DQS	1	HM_RWDS/ OM_DQS	1

	Pin Name	Package 324 Pins	Number of Pins	Package 272, 256 Pins	Number of Pins	Package 176 Pins	Number of Pins
Dedicated Pin	HM_DQ0/OM_SIO0	HM_DQ0/OM_SIO0	1	HM_DQ0/OM_SIO0	1	HM_DQ0/OM_SIO0	1
	HM_DQ1/OM_SIO1	HM_DQ1/OM_SIO1	1	HM_DQ1/OM_SIO1	1	HM_DQ1/OM_SIO1	1
	HM_DQ2/OM_SIO2	HM_DQ2/OM_SIO2	1	HM_DQ2/OM_SIO2	1	HM_DQ2/OM_SIO2	1
	HM_DQ3/OM_SIO3	HM_DQ3/OM_SIO3	1	HM_DQ3/OM_SIO3	1	HM_DQ3/OM_SIO3	1
	HM_DQ4/OM_SIO4	HM_DQ4/OM_SIO4	1	HM_DQ4/OM_SIO4	1	HM_DQ4/OM_SIO4	1
	HM_DQ5/OM_SIO5	HM_DQ5/OM_SIO5	1	HM_DQ5/OM_SIO5	1	HM_DQ5/OM_SIO5	1
	HM_DQ6/OM_SIO6	HM_DQ6/OM_SIO6	1	HM_DQ6/OM_SIO6	1	HM_DQ6/OM_SIO6	1
	HM_DQ7/OM_SIO7	HM_DQ7/OM_SIO7	1	HM_DQ7/OM_SIO7	1	HM_DQ7/OM_SIO7	1
	HM_RESET#/ OM_RESET#	HM_RESET#/ OM_RESET#	1	HM_RESET#/ OM_RESET#	1	HM_RESET#/ OM_RESET#	1
	BSCANP	BSCANP	1	BSCANP	1	BSCANP	1
	QSPI0_SPCLK	QSPI0_SPCLK	1	QSPI0_SPCLK	1	QSPI0_SPCLK	1
	QSPI0_IO0	QSPI0_IO0	1	QSPI0_IO0	1	QSPI0_IO0	1
	QSPI0_IO1	QSPI0_IO1	1	QSPI0_IO1	1	QSPI0_IO1	1
	QSPI0_IO2	QSPI0_IO2	1	QSPI0_IO2	1	QSPI0_IO2	1
	QSPI0_IO3	QSPI0_IO3	1	QSPI0_IO3	1	QSPI0_IO3	1
	QSPI0_SSL	QSPI0_SSL	1	QSPI0_SSL	1	QSPI0_SSL	1
	RPC_RESET#	RPC_RESET#	1	RPC_RESET#	1	RPC_RESET#	1
	RPC_WP#	RPC_WP#	1	RPC_WP#	1	RPC_WP#	1
	RPC_INT#	RPC_INT#	1	RPC_INT#	1	RPC_INT#	1
	QSPI1_SPCLK	QSPI1_SPCLK	1	QSPI1_SPCLK	1	QSPI1_SPCLK	1
	QSPI1_IO0	QSPI1_IO0	1	QSPI1_IO0	1	QSPI1_IO0	1
	QSPI1_IO1	QSPI1_IO1	1	QSPI1_IO1	1	QSPI1_IO1	1
	QSPI1_IO2	QSPI1_IO2	1	QSPI1_IO2	1	QSPI1_IO2	1
	QSPI1_IO3	QSPI1_IO3	1	QSPI1_IO3	1	QSPI1_IO3	1
	QSPI1_SSL	QSPI1_SSL	1	QSPI1_SSL	1	QSPI1_SSL	1

Note: The dedicated pins of MIPI, LVDS, and USB are excluded from the list. Refer to the related sections for each pin.

Table 51.2 Port Functions

Port	Switching of Driving Ability	TTL	Schmitt Input	1.8V	1.8 V/3.3 V Voltage Switching
P0_0 to P0_6	—	√	√	—	—
P1_0 to P1_4	—	√	√	—	—
P2_0 to P2_3	—	√	√	—	—
P3_0 to P3_5	—	—	√	—	—
P4_0 to P4_7	—	—	√	—	—
P5_0 to P5_7	—	—	√	—	—
P6_0 to P6_7	—	—	√	—	—
P7_0 to P7_7	—	—	√	—	—
P8_0 to P8_7	—	—	√	—	—
P9_0 to P9_7	—	—	√	—	—
PA_0 to PA_7	—	—	√	—	—
PB_0 to PB_5	—	—	√	—	—
PC_0 to PC_7	—	—	√	—	—
PD_0 to PD_7	—	—	√	—	—
PE_0 to PE_6	—	—	√	—	—

Port	Switching of Driving Ability	TTL	Schmitt Input	1.8V	1.8 V/3.3 V Voltage Switching
PF_0 to PF_7	—	—	√	—	—
PG_0 to PG_7	√*1	—	√	—	—
PH_0 to PH_6	—	—	√	—	—
PJ_0 to PJ_7	√*1	—	√	—	—
PK_0 to PK_5	—	—	√	—	—
PL_0 to PL_4	—	—	√	—	—
JP0_0	—	√	—	—	—
JP0_1	—	—	—	—	—
CKIO	√	—	—	—	—
RES#	—	—	√	—	—
NMI	—	—	√	—	—
USB_X1	—	—	—	—	—
USB_X2	—	—	—	—	—
EXTAL	—	—	—	—	—
XTAL	—	—	—	—	—
RTC_X1	—	—	—	—	—
RTC_X2	—	—	—	—	—
AUDIO_X1	—	—	—	—	—
AUDIO_X2	—	—	—	—	—
TMS/SWDIO	—	√	—	—	—
TCK/SWDCLK	—	√	—	—	—
TRST#	—	—	√	—	—
SD1_CLK	—	—	—	—	√
SD1_CMD	—	—	—	—	√
SD1_DAT0	—	—	—	—	√
SD1_DAT1	—	—	—	—	√
SD1_DAT2	—	—	—	—	√
SD1_DAT3	—	—	—	—	√
SD0_CLK	—	—	—	—	√
SD0_CMD	—	—	—	—	√
SD0_DAT0	—	—	—	—	√
SD0_DAT1	—	—	—	—	√
SD0_DAT2	—	—	—	—	√
SD0_DAT3	—	—	—	—	√
SD0_DAT4	—	—	—	—	√
SD0_DAT5	—	—	—	—	√
SD0_DAT6	—	—	—	—	√
SD0_DAT7	—	—	—	—	√
SD0_RST#	—	—	—	—	√
HM_CK/OM_SCLK	—	—	—	√	—
HM_CK#	—	—	—	√	—
HM_CS0#/ OM_CS0#	—	—	—	√	—
HM_CS1#/ OM_CS1#	—	—	—	√	—
HM_RWDS/ OM_DQS	—	—	—	√	—

Port	Switching of Driving Ability	TTL	Schmitt Input	1.8V	1.8 V/3.3 V Voltage Switching
HM_DQ0/OM_SIO0	—	—	—	√	—
HM_DQ1/OM_SIO1	—	—	—	√	—
HM_DQ2/OM_SIO2	—	—	—	√	—
HM_DQ3/OM_SIO3	—	—	—	√	—
HM_DQ4/OM_SIO4	—	—	—	√	—
HM_DQ5/OM_SIO5	—	—	—	√	—
HM_DQ6/OM_SIO6	—	—	—	√	—
HM_DQ7/OM_SIO7	—	—	—	√	—
HM_RESET#/ OM_RESET#	—	—	—	√	—
BSCANP	—	—	√	—	—
QSPI0_SPCLK	—	—	—	—	√
QSPI0_IO0	—	—	—	—	√
QSPI0_IO1	—	—	—	—	√
QSPI0_IO2	—	—	—	—	√
QSPI0_IO3	—	—	—	—	√
QSPI0_SSL	—	—	—	—	√
RPC_RESET#	—	—	—	—	√
RPC_WP#	—	—	—	—	√
RPC_INT#	—	—	—	—	√
QSPI1_SPCLK	—	—	—	—	√
QSPI1_IO0	—	—	—	—	√
QSPI1_IO1	—	—	—	—	√
QSPI1_IO2	—	—	—	—	√
QSPI1_IO3	—	—	—	—	√
QSPI1_SSL	—	—	—	—	√

Note 1. The driving ability of some pins cannot be switched. For details, refer to section 51.3.5, GPIO Driving Ability Control Register (DSCR).
Note 2. The dedicated pins of MIPI, LVDS, and USB are excluded from the list. Refer to the related sections for each pin.

51.1.1 Port group index m n

Throughout this section, the individual port groups are identified by the index "m" (m = 0 to 9, A to H, J to M), and the port groups pins are identified by the index "n" (n = 0 to 7). For example PORTm.PMRn for the port mode control register and Pmn.PFS for the pin control register which are used for the Pmn pin.
The index “m = M” means the pin control register which is used for the JP0_0 and JP0_1 pins.

51.1.2 Base address

The addresses of the registers used to control the ports are given as addresses offset from the base address <PORTm_base>.

Table 51.3 Base Address

Name	Address
<PORTm_base>	H'FCFF E000

51.2 GPIO Port Configuration

Figure 51.1 to Figure 51.15 show the configuration of the GPIO ports.

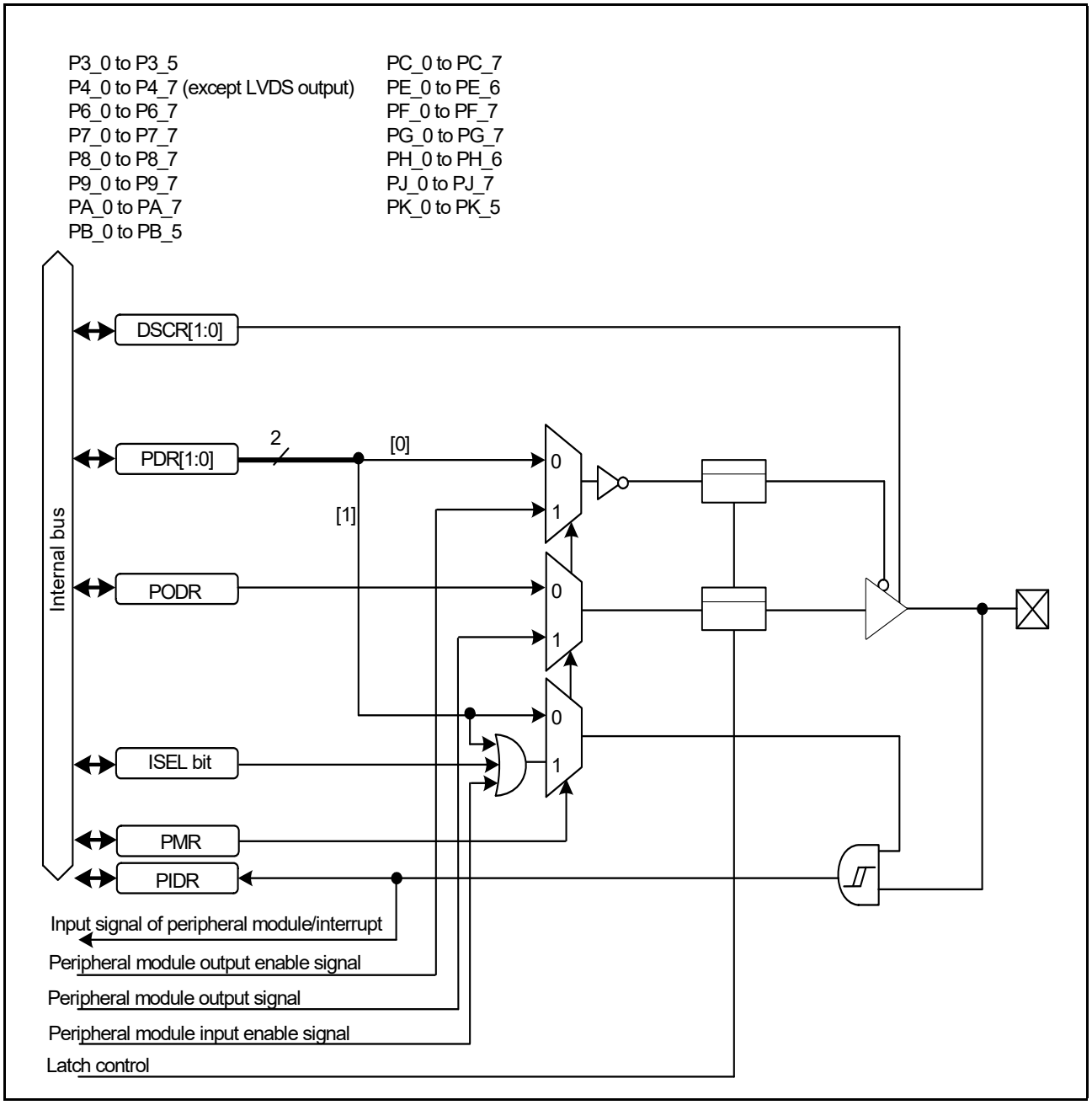


Figure 51.1 (1) Input/Output Buffer with Schmitt AND Input and Latch

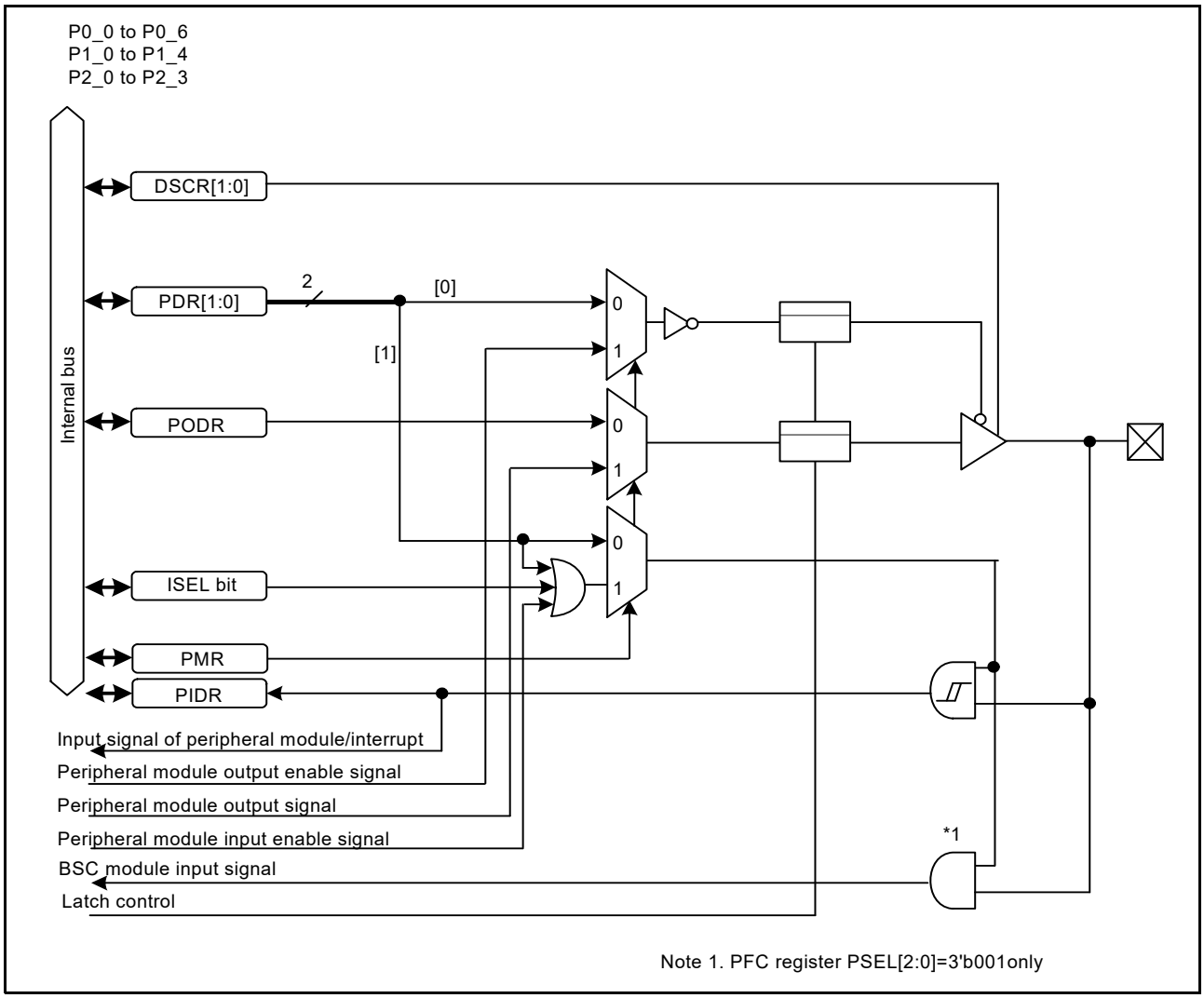


Figure 51.2 (2) Input/Output Buffer with TTL AND Input, Schmitt AND Input, and Latch

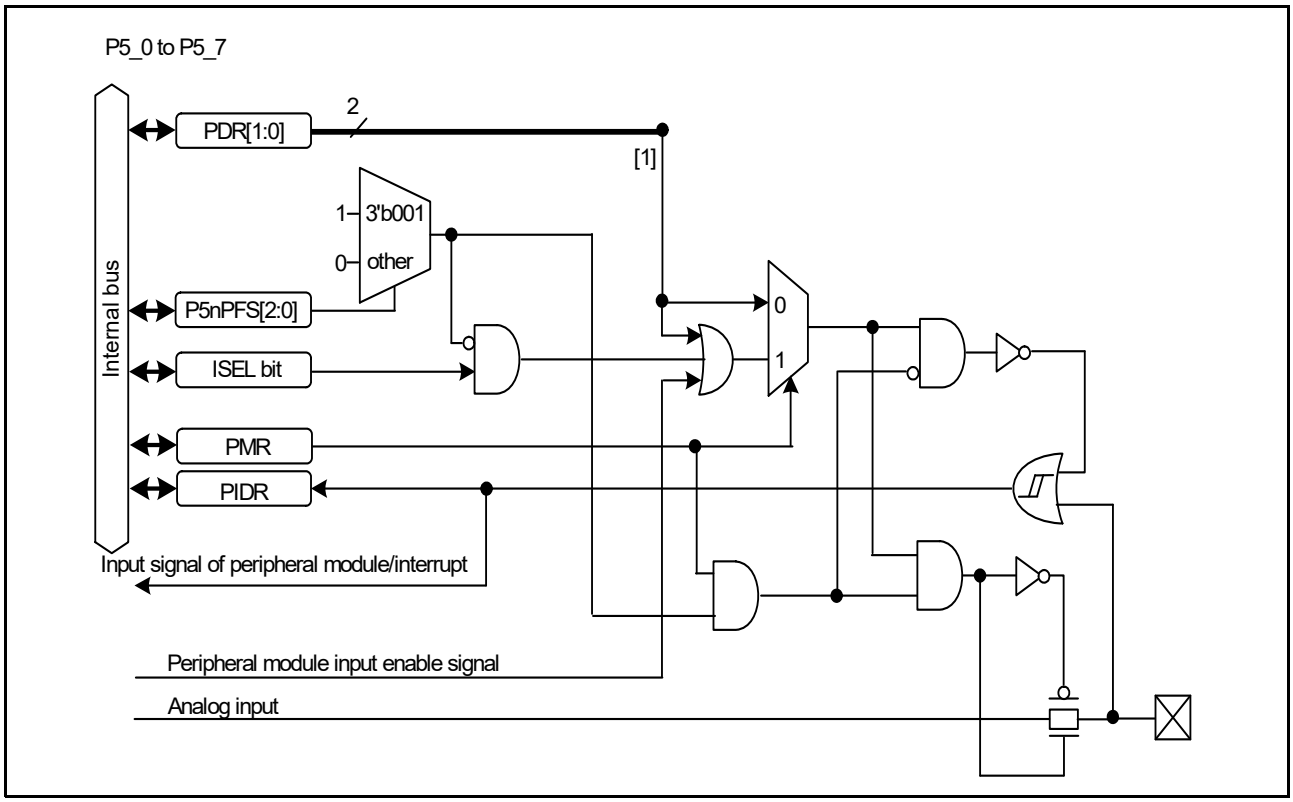
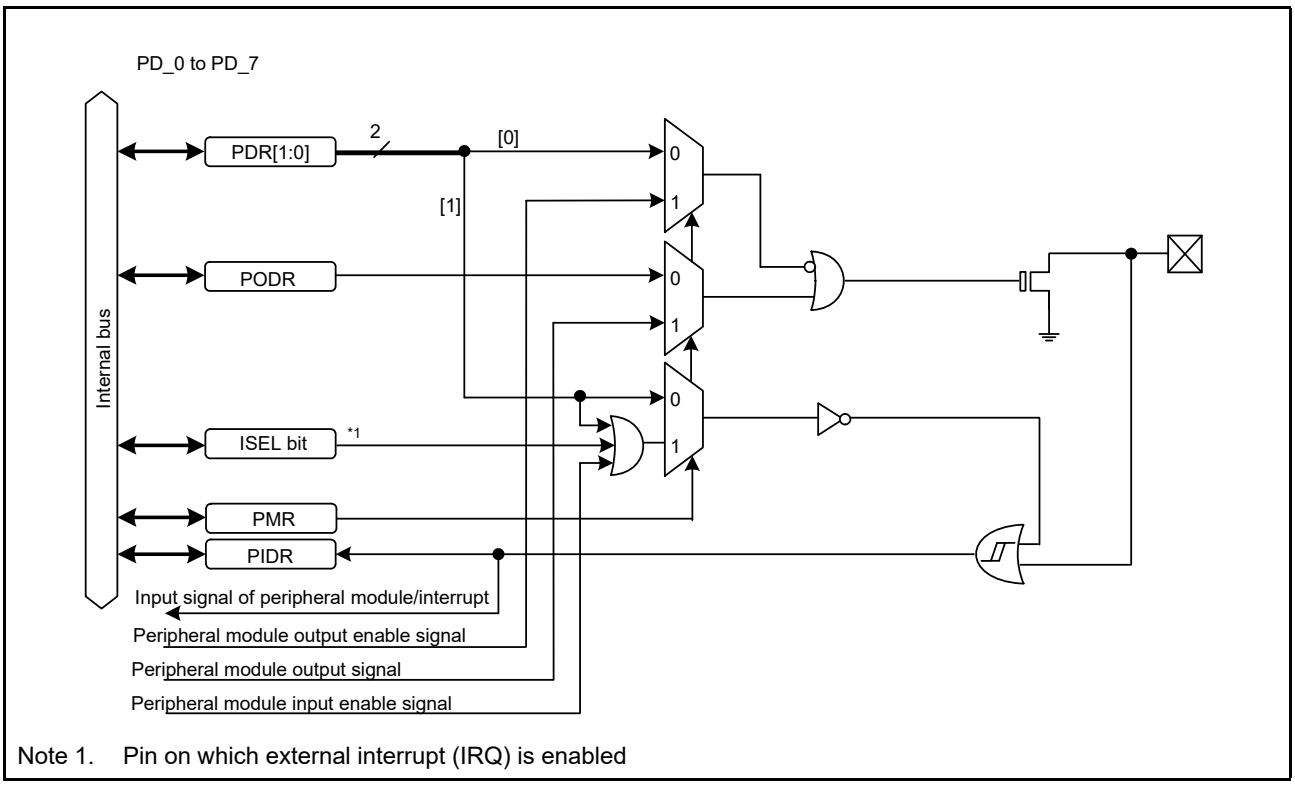
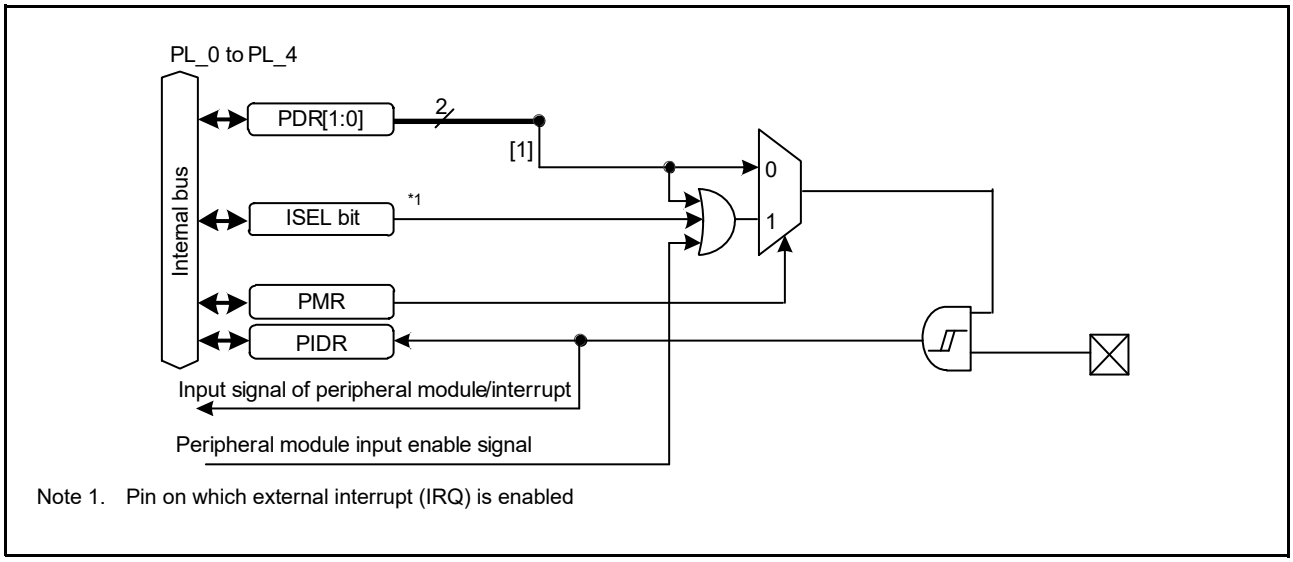


Figure 51.3 (3) Input Buffer Multiplexed with Schmitt OR Input and A/D Input



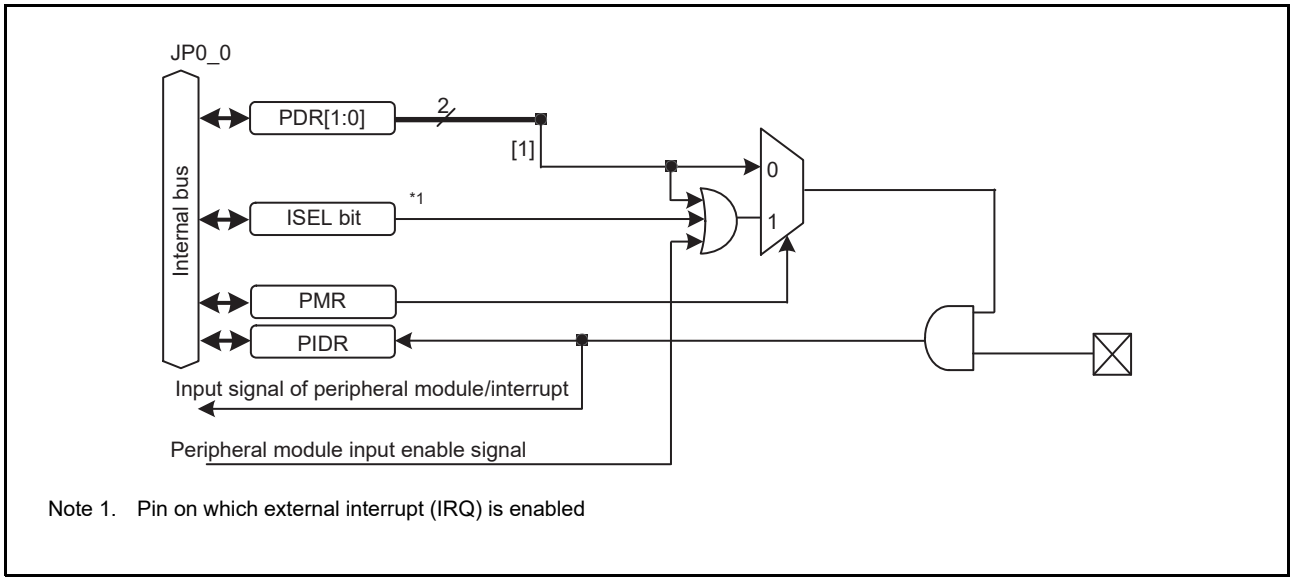
Note 1. Pin on which external interrupt (IRQ) is enabled

Figure 51.4 (4) Bidirectional Buffer with Schmitt OR Input and Open-Drain Output



Note 1. Pin on which external interrupt (IRQ) is enabled

Figure 51.5 (5) Schmitt AND Input Buffer



Note 1. Pin on which external interrupt (IRQ) is enabled

Figure 51.6 (6) TTL AND Input Buffer

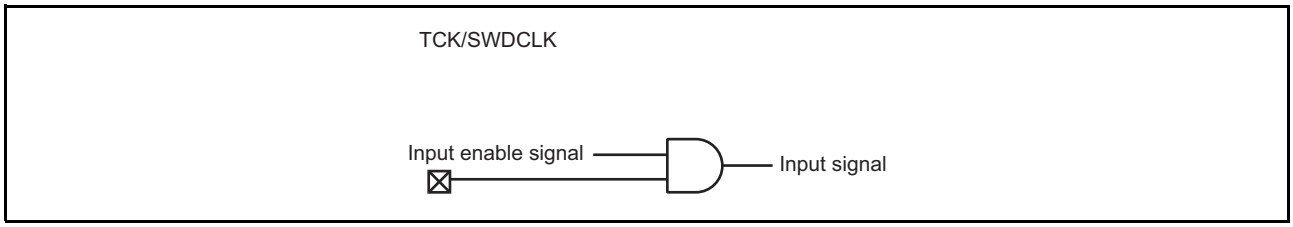


Figure 51.7 (7) TTL AND Input Buffer

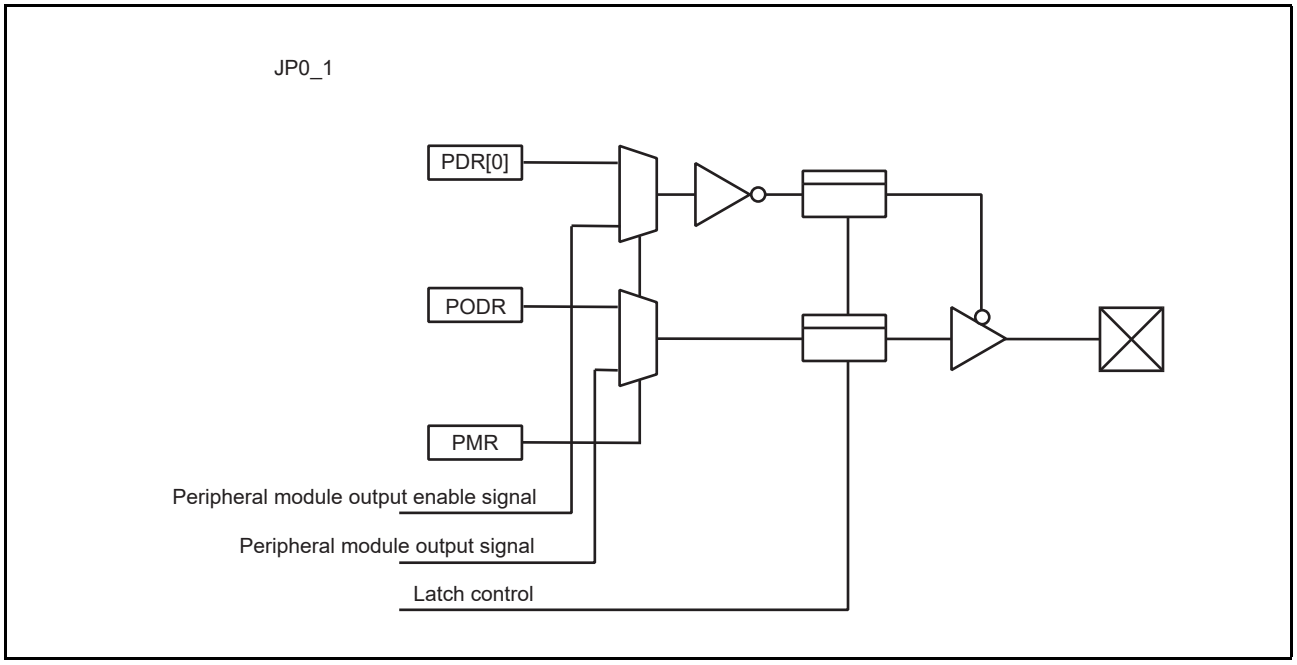


Figure 51.8 (8) Output Buffer with Latch and Enable Signal

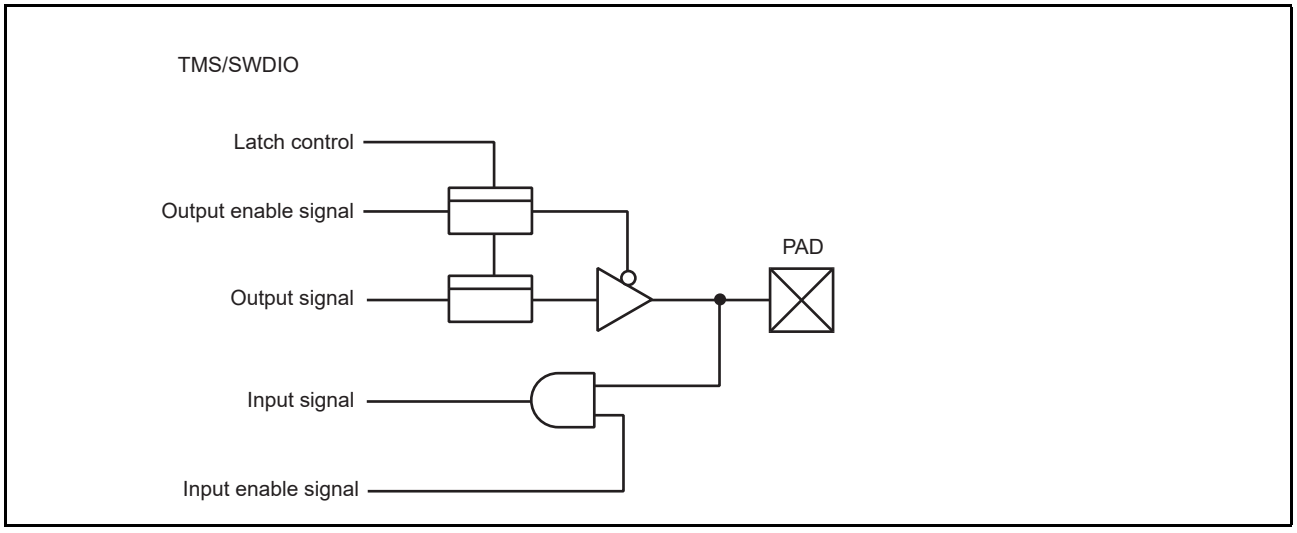


Figure 51.9 (9) Bidirectional Buffer with TTL AND Input and Latch

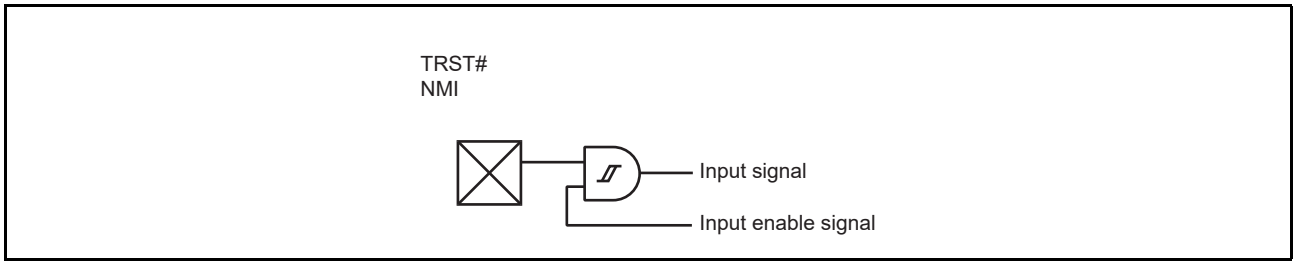


Figure 51.10 (10) Schmitt AND Input Buffer

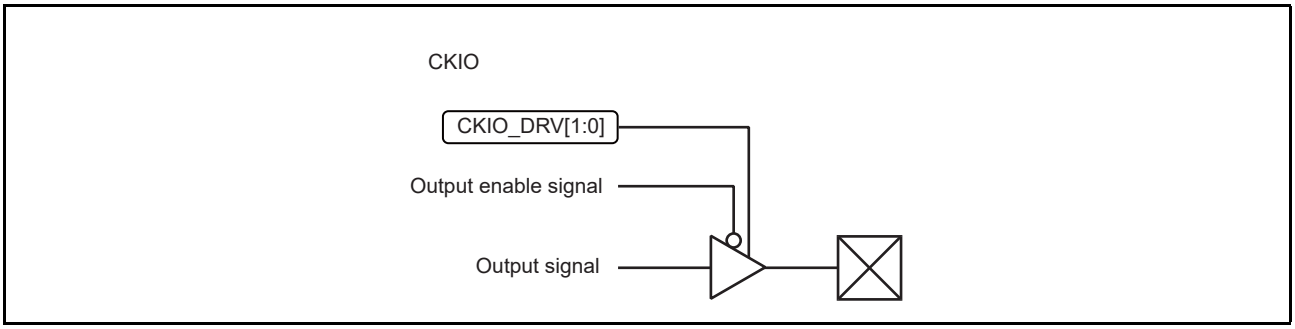


Figure 51.11 (11) Output Buffer with Enable Signal and Controllable Driving Ability

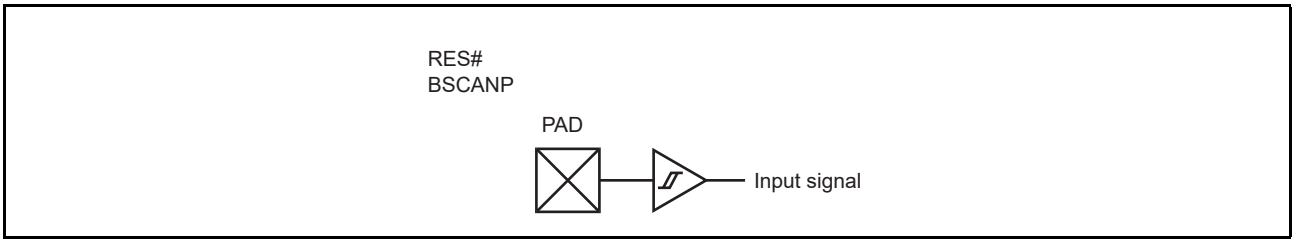


Figure 51.12 (12) Schmitt Input Buffer

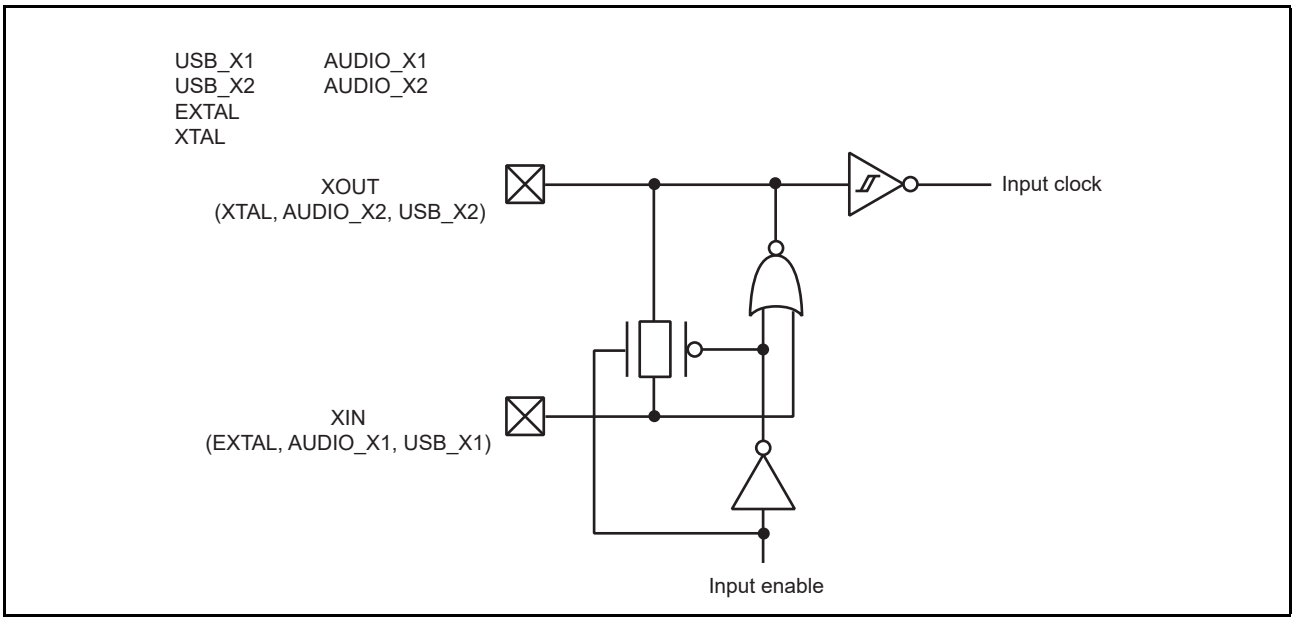


Figure 51.13 (13) Oscillation Buffer (1)

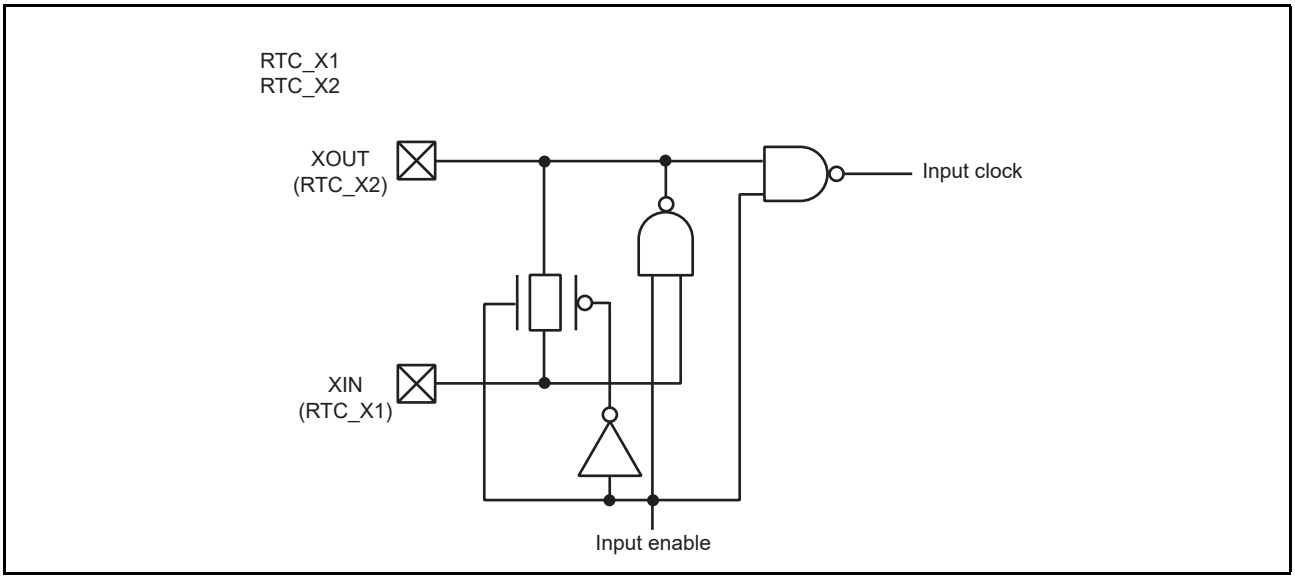


Figure 51.14 (14) Oscillation Buffer (2)

SD1_CLK	SD0_DAT1	HM_CK#	HM_DQ5/OM_SIO5	QSPI0_SSL	QSPI1_SSL
SD1_CMD	SD0_DAT2	HM_CS0#/OM_CS0#	HM_DQ6/OM_SIO6	RPC_RESET#	
SD1_DAT0	SD0_DAT3	HM_CS1#/OM_CS1#	HM_DQ7/OM_SIO7	RPC_WP#	
SD1_DAT1	SD0_DAT4	HM_RWDS/OM_DQS	HM_RESET#/OM_RESET#	RPC_INT#	
SD1_DAT2	SD0_DAT5	HM_DQ0/OM_SIO0	QSPI0_SPCLK	QSPI1_SPCLK	
SD1_DAT3	SD0_DAT6	HM_DQ1/OM_SIO1	QSPI0_IO0	QSPI1_IO0	
SD0_CLK	SD0_DAT7	HM_DQ2/OM_SIO2	QSPI0_IO1	QSPI1_IO1	
SD0_CMD	SD0_RST#	HM_DQ3/OM_SIO3	QSPI0_IO2	QSPI1_IO2	
SD0_DAT0	HM_CK/OM_SCLK	HM_DQ4/OM_SIO4	QSPI0_IO3	QSPI1_IO3	

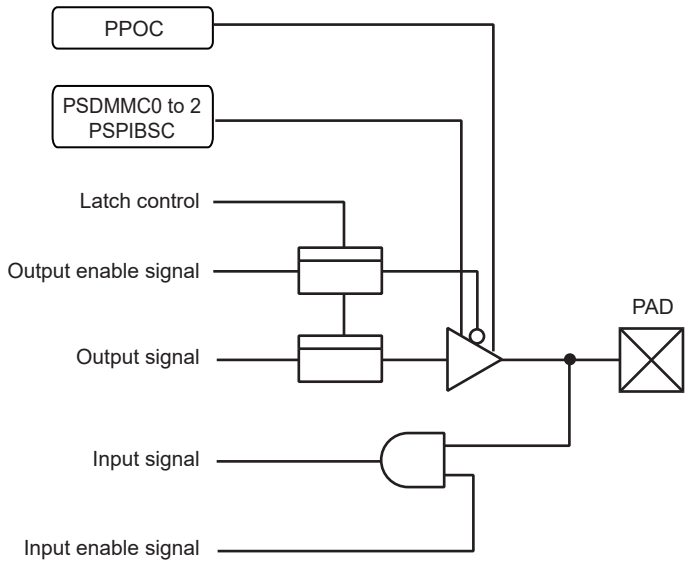


Figure 51.15 (15) Bidirectional Buffer with AND Input, Controllable Driving Ability, Operating Voltage Selection, and Latch

51.3 Register Description

The following registers are used for setting the individual pins.

Table 51.4 Register Configuration

Symbol	Register Name	R/W	8 Bit Access	16 Bit Access	32 Bit Access	Address
PORT0.PDR	Port Direction Register (PORT0)	R/W	—	√	—	<PORTm_base> + 0000H
PORT1.PDR	Port Direction Register (PORT1)	R/W	—	√	—	<PORTm_base> + 0002H
PORT2.PDR	Port Direction Register (PORT2)	R/W	—	√	—	<PORTm_base> + 0004H
PORT3.PDR	Port Direction Register (PORT3)	R/W	—	√	—	<PORTm_base> + 0006H
PORT4.PDR	Port Direction Register (PORT4)	R/W	—	√	—	<PORTm_base> + 0008H
PORT5.PDR	Port Direction Register (PORT5)	R/W	—	√	—	<PORTm_base> + 000AH
PORT6.PDR	Port Direction Register (PORT6)	R/W	—	√	—	<PORTm_base> + 000CH
PORT7.PDR	Port Direction Register (PORT7)	R/W	—	√	—	<PORTm_base> + 000EH
PORT8.PDR	Port Direction Register (PORT8)	R/W	—	√	—	<PORTm_base> + 0010H
PORT9.PDR	Port Direction Register (PORT9)	R/W	—	√	—	<PORTm_base> + 0012H
PORTA.PDR	Port Direction Register (PORTA)	R/W	—	√	—	<PORTm_base> + 0014H
PORTB.PDR	Port Direction Register (PORTB)	R/W	—	√	—	<PORTm_base> + 0016H
PORTC.PDR	Port Direction Register (PORTC)	R/W	—	√	—	<PORTm_base> + 0018H
PORTD.PDR	Port Direction Register (PORTD)	R/W	—	√	—	<PORTm_base> + 001AH
PORTE.PDR	Port Direction Register (PORTE)	R/W	—	√	—	<PORTm_base> + 001CH
PORTF.PDR	Port Direction Register (PORTF)	R/W	—	√	—	<PORTm_base> + 001EH
PORTG.PDR	Port Direction Register (PORTG)	R/W	—	√	—	<PORTm_base> + 0020H
PORTH.PDR	Port Direction Register (PORTH)	R/W	—	√	—	<PORTm_base> + 0022H
PORTJ.PDR	Port Direction Register (PORTJ)	R/W	—	√	—	<PORTm_base> + 0024H
PORTK.PDR	Port Direction Register (PORTK)	R/W	—	√	—	<PORTm_base> + 0026H
PORTL.PDR	Port Direction Register (PORTL)	R/W	—	√	—	<PORTm_base> + 0028H
PORTM.PDR	Port Direction Register (PORTM)	R/W	—	√	—	<PORTm_base> + 002AH
PORT0.PODR	Port Output Data Register (PORT0)	R/W	√	—	—	<PORTm_base> + 0040H
PORT1.PODR	Port Output Data Register (PORT1)	R/W	√	—	—	<PORTm_base> + 0041H
PORT2.PODR	Port Output Data Register (PORT2)	R/W	√	—	—	<PORTm_base> + 0042H
PORT3.PODR	Port Output Data Register (PORT3)	R/W	√	—	—	<PORTm_base> + 0043H
PORT4.PODR	Port Output Data Register (PORT4)	R/W	√	—	—	<PORTm_base> + 0044H
PORT5.PODR	Port Output Data Register (PORT5)	R/W	√	—	—	<PORTm_base> + 0045H
PORT6.PODR	Port Output Data Register (PORT6)	R/W	√	—	—	<PORTm_base> + 0046H
PORT7.PODR	Port Output Data Register (PORT7)	R/W	√	—	—	<PORTm_base> + 0047H
PORT8.PODR	Port Output Data Register (PORT8)	R/W	√	—	—	<PORTm_base> + 0048H
PORT9.PODR	Port Output Data Register (PORT9)	R/W	√	—	—	<PORTm_base> + 0049H
PORTA.PODR	Port Output Data Register (PORTA)	R/W	√	—	—	<PORTm_base> + 004AH
PORTB.PODR	Port Output Data Register (PORTB)	R/W	√	—	—	<PORTm_base> + 004BH
PORTC.PODR	Port Output Data Register (PORTC)	R/W	√	—	—	<PORTm_base> + 004CH
PORTD.PODR	Port Output Data Register (PORTD)	R/W	√	—	—	<PORTm_base> + 004DH
PORTE.PODR	Port Output Data Register (PORTE)	R/W	√	—	—	<PORTm_base> + 004EH
PORTF.PODR	Port Output Data Register (PORTF)	R/W	√	—	—	<PORTm_base> + 004FH
PORTG.PODR	Port Output Data Register (PORTG)	R/W	√	—	—	<PORTm_base> + 0050H
PORTH.PODR	Port Output Data Register (PORTH)	R/W	√	—	—	<PORTm_base> + 0051H
PORTJ.PODR	Port Output Data Register (PORTJ)	R/W	√	—	—	<PORTm_base> + 0052H

Symbol	Register Name	R/W	8 Bit Access	16 Bit Access	32 Bit Access	Address
PORTK.PODR	Port Output Data Register (PORTK)	R/W	√	—	—	<PORTm_base> + 0053H
PORTL.PODR	Port Output Data Register (PORTL)	R/W	√	—	—	<PORTm_base> + 0054H
PORTM.PODR	Port Output Data Register (PORTM)	R/W	√	—	—	<PORTm_base> + 0055H
PORT0.PIDR	Port Input Data Register (PORT0)	R	√	—	—	<PORTm_base> + 0060H
PORT1.PIDR	Port Input Data Register (PORT1)	R	√	—	—	<PORTm_base> + 0061H
PORT2.PIDR	Port Input Data Register (PORT2)	R	√	—	—	<PORTm_base> + 0062H
PORT3.PIDR	Port Input Data Register (PORT3)	R	√	—	—	<PORTm_base> + 0063H
PORT4.PIDR	Port Input Data Register (PORT4)	R	√	—	—	<PORTm_base> + 0064H
PORT5.PIDR	Port Input Data Register (PORT5)	R	√	—	—	<PORTm_base> + 0065H
PORT6.PIDR	Port Input Data Register (PORT6)	R	√	—	—	<PORTm_base> + 0066H
PORT7.PIDR	Port Input Data Register (PORT7)	R	√	—	—	<PORTm_base> + 0067H
PORT8.PIDR	Port Input Data Register (PORT8)	R	√	—	—	<PORTm_base> + 0068H
PORT9.PIDR	Port Input Data Register (PORT9)	R	√	—	—	<PORTm_base> + 0069H
PORTA.PIDR	Port Input Data Register (PORTA)	R	√	—	—	<PORTm_base> + 006AH
PORTB.PIDR	Port Input Data Register (PORTB)	R	√	—	—	<PORTm_base> + 006BH
PORTC.PIDR	Port Input Data Register (PORTC)	R	√	—	—	<PORTm_base> + 006CH
PORTD.PIDR	Port Input Data Register (PORTD)	R	√	—	—	<PORTm_base> + 006DH
PORTE.PIDR	Port Input Data Register (PORTE)	R	√	—	—	<PORTm_base> + 006EH
PORTF.PIDR	Port Input Data Register (PORTF)	R	√	—	—	<PORTm_base> + 006FH
PORTG.PIDR	Port Input Data Register (PORTG)	R	√	—	—	<PORTm_base> + 0070H
PORTH.PIDR	Port Input Data Register (PORTH)	R	√	—	—	<PORTm_base> + 0071H
PORTJ.PIDR	Port Input Data Register (PORTJ)	R	√	—	—	<PORTm_base> + 0072H
PORTK.PIDR	Port Input Data Register (PORTK)	R	√	—	—	<PORTm_base> + 0073H
PORTL.PIDR	Port Input Data Register (PORTL)	R	√	—	—	<PORTm_base> + 0074H
PORTM.PIDR	Port Input Data Register (PORTM)	R	√	—	—	<PORTm_base> + 0075H
PORT0.PMR	Port Mode Register (PORT0)	R/W	√	—	—	<PORTm_base> + 0080H
PORT1.PMR	Port Mode Register (PORT1)	R/W	√	—	—	<PORTm_base> + 0081H
PORT2.PMR	Port Mode Register (PORT2)	R/W	√	—	—	<PORTm_base> + 0082H
PORT3.PMR	Port Mode Register (PORT3)	R/W	√	—	—	<PORTm_base> + 0083H
PORT4.PMR	Port Mode Register (PORT4)	R/W	√	—	—	<PORTm_base> + 0084H
PORT5.PMR	Port Mode Register (PORT5)	R/W	√	—	—	<PORTm_base> + 0085H
PORT6.PMR	Port Mode Register (PORT6)	R/W	√	—	—	<PORTm_base> + 0086H
PORT7.PMR	Port Mode Register (PORT7)	R/W	√	—	—	<PORTm_base> + 0087H
PORT8.PMR	Port Mode Register (PORT8)	R/W	√	—	—	<PORTm_base> + 0088H
PORT9.PMR	Port Mode Register (PORT9)	R/W	√	—	—	<PORTm_base> + 0089H
PORTA.PMR	Port Mode Register (PORTA)	R/W	√	—	—	<PORTm_base> + 008AH
PORTB.PMR	Port Mode Register (PORTB)	R/W	√	—	—	<PORTm_base> + 008BH
PORTC.PMR	Port Mode Register (PORTC)	R/W	√	—	—	<PORTm_base> + 008CH
PORTD.PMR	Port Mode Register (PORTD)	R/W	√	—	—	<PORTm_base> + 008DH
PORTE.PMR	Port Mode Register (PORTE)	R/W	√	—	—	<PORTm_base> + 008EH
PORTF.PMR	Port Mode Register (PORTF)	R/W	√	—	—	<PORTm_base> + 008FH
PORTG.PMR	Port Mode Register (PORTG)	R/W	√	—	—	<PORTm_base> + 0090H
PORTH.PMR	Port Mode Register (PORTH)	R/W	√	—	—	<PORTm_base> + 0091H
PORTJ.PMR	Port Mode Register (PORTJ)	R/W	√	—	—	<PORTm_base> + 0092H
PORTK.PMR	Port Mode Register (PORTK)	R/W	√	—	—	<PORTm_base> + 0093H
PORTL.PMR	Port Mode Register (PORTL)	R/W	√	—	—	<PORTm_base> + 0094H

Symbol	Register Name	R/W	8 Bit Access	16 Bit Access	32 Bit Access	Address
PORTM.PMR	Port Mode Register (PORTM)	R/W	√	—	—	<PORTm_base> + 0095H
PORT0.DSCR	Driving Ability Control Register (PORT0)	R/W	—	√	—	<PORTm_base> + 0140H
PORT1.DSCR	Driving Ability Control Register (PORT1)	R/W	—	√	—	<PORTm_base> + 0142H
PORT2.DSCR	Driving Ability Control Register (PORT2)	R/W	—	√	—	<PORTm_base> + 0144H
PORT3.DSCR	Driving Ability Control Register (PORT3)	R/W	—	√	—	<PORTm_base> + 0146H
PORT4.DSCR	Driving Ability Control Register (PORT4)	R/W	—	√	—	<PORTm_base> + 0148H
PORT5.DSCR	Driving Ability Control Register (PORT5)	R/W	—	√	—	<PORTm_base> + 014AH
PORT6.DSCR	Driving Ability Control Register (PORT6)	R/W	—	√	—	<PORTm_base> + 014CH
PORT7.DSCR	Driving Ability Control Register (PORT7)	R/W	—	√	—	<PORTm_base> + 014EH
PORT8.DSCR	Driving Ability Control Register (PORT8)	R/W	—	√	—	<PORTm_base> + 0150H
PORT9.DSCR	Driving Ability Control Register (PORT9)	R/W	—	√	—	<PORTm_base> + 0152H
PORTA.DSCR	Driving Ability Control Register (PORTA)	R/W	—	√	—	<PORTm_base> + 0154H
PORTB.DSCR	Driving Ability Control Register (PORTB)	R/W	—	√	—	<PORTm_base> + 0156H
PORTC.DSCR	Driving Ability Control Register (PORTC)	R/W	—	√	—	<PORTm_base> + 0158H
PORTD.DSCR	Driving Ability Control Register (PORTD)	R/W	—	√	—	<PORTm_base> + 015AH
PORTE.DSCR	Driving Ability Control Register (PORTE)	R/W	—	√	—	<PORTm_base> + 015CH
PORTF.DSCR	Driving Ability Control Register (PORTF)	R/W	—	√	—	<PORTm_base> + 015EH
PORTG.DSCR	Driving Ability Control Register (PORTG)	R/W	—	√	—	<PORTm_base> + 0160H
PORTH.DSCR	Driving Ability Control Register (PORTH)	R/W	—	√	—	<PORTm_base> + 0162H
PORTJ.DSCR	Driving Ability Control Register (PORTJ)	R/W	—	√	—	<PORTm_base> + 0164H
PORTK.DSCR	Driving Ability Control Register (PORTK)	R/W	—	√	—	<PORTm_base> + 0166H
PORTL.DSCR	Driving Ability Control Register (PORTL)	R/W	—	√	—	<PORTm_base> + 0168H
PORTM.DSCR	Driving Ability Control Register (PORTM)	R/W	—	√	—	<PORTm_base> + 016AH
P0n.PFS	P0n Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0200H + n
P1n.PFS	P1n Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0208H + n
P2n.PFS	P2n Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0210H + n
P3n.PFS	P3n Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0218H + n
P4n.PFS	P4n Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0220H + n
P5n.PFS	P5n Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0228H + n
P6n.PFS	P6n Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0230H + n
P7n.PFS	P7n Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0238H + n

Symbol	Register Name	R/W	8 Bit Access	16 Bit Access	32 Bit Access	Address
P8n.PFS	P8n Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0240H + n
P9n.PFS	P9n Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0248H + n
PAn.PFS	PAn Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0250H + n
PBn.PFS	PBn Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0258H + n
PCn.PFS	PCn Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0260H + n
PDn.PFS	PDn Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0268H + n
PEn.PFS	PEn Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0270H + n
PFn.PFS	PFn Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0278H + n
PGn.PFS	PGn Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0280H + n
PHn.PFS	PHn Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0288H + n
PJn.PFS	PJn Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0290H + n
PKn.PFS	PKn Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 0298H + n
PLn.PFS	PLn Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 02A0H + n
PMn.PFS	PMn Pin Function Control Register	R/W	√	—	—	<PORTm_base> + 02A8H + n
PWPR	Write Protect Register	R/W	√	—	—	<PORTm_base> + 02FFH
PFENET	Ethernet Control Register	R/W	√	—	—	<PORTm_base> + 0820H
PPOC	Dedicated Pin POC Control Register	R/W	—	—	√	<PORTm_base> + 0900H
PSDMMC0	SD/MMC Host Interface Dedicated Pin Driving Ability Control Register 0	R/W	—	—	√	<PORTm_base> + 0920H
PSDMMC1	SD/MMC Host Interface Dedicated Pin Driving Ability Control Register 1	R/W	—	—	√	<PORTm_base> + 0930H
PSDMMC2	SD/MMC Host Interface Dedicated Pin Driving Ability Control Register 2	R/W	—	—	√	<PORTm_base> + 0940H
PSPIBSC	SPI Multi I/O Bus Controller Dedicated Pin Driving Ability Control Register	R/W	—	—	√	<PORTm_base> + 0960H
PHMOM0	HyperBus Controller / Octa Memory Controller Dedicated Pin Control Register	R/W	—	—	√	<PORTm_base> + 0980H
PMODEPFS	Peripheral Pin Function Control Register	R/W	—	—	√	<PORTm_base> + 09C0H
PCKIO	CKIO Driving Ability Control Register	R/W	√	—	—	<PORTm_base> + 09D0H

51.3.1 Port Direction Register (PDR)

The PDR register is used to select non-use, input, or output (input enable) for individual pins of the corresponding port when the pins are configured as the general I/O pins. When 00 (Hi-Z) is set to this register, this LSI can be protected from input Hi-z state.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PDR7[1:0]		PDR6[1:0]		PDR5[1:0]		PDR4[1:0]		PDR3[1:0]		PDR2[1:0]		PDR1[1:0]		PDR0[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
15 to 14	PDR7[1:0]	00	R/W	Odd bit Even bit 0 0: Hi-Z 0 1: Setting prohibited 1 0: Input (functions as an input pin.) 1 1: Output (functions as an output pin (port read enable).)
13 to 12	PDR6[1:0]	00	R/W	
11 to 10	PDR5[1:0]	00	R/W	
9 to 8	PDR4[1:0]	00	R/W	
7 to 6	PDR3[1:0]	00	R/W	
5 to 4	PDR2[1:0]	00	R/W	
3 to 2	PDR1[1:0]	00	R/W	
1 to 0	PDR0[1:0]	00	R/W	

Note: Each bit of PORTm.PDR corresponds to each pin of port m; pin function can be specified in 2-bit units.
Write 00 (Hi-Z) or 10 (input) to the PORT5.PDRn (n = 0 to 7), PORTL.PDRn (n = 0 to 4), and PORTM.PDR0 bits because the P5_0 to P5_7, PL_0 to PL_4, and JP0_0 pins are only for input. Write 00 (non-use) or 11 (output) to the PORTM.PDR1 bit because the JP0_1 pin is only for input.
The bit corresponding to a pin that does not exist is also reserved. A reserved bit is always read as 0. The write value should always be 0.

51.3.2 Port Output Data Register (PODR)

The PODR register holds the data to be output from the pins used for general I/O.

	7	6	5	4	3	2	1	0
	PODR7	PODR6	PODR5	PODR4	PODR3	PODR2	PODR1	PODR0
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	PODR7	0	R/W	0: Low output 1: High output
6	PODR6	0	R/W	
5	PODR5	0	R/W	
4	PODR4	0	R/W	
3	PODR3	0	R/W	
2	PODR2	0	R/W	
1	PODR1	0	R/W	
0	PODR0	0	R/W	

Note: Write 0 to the PORT5.PODRn (n = 0 to 7), PORTL.PODRn (n = 0 to 4), and PDRTM.PODR0 bits because the P5_0 to P5_7, PL_0 to PL_4, and JP0_0 pins are only for input. The bit corresponding to a pin that does not exist is reserved. A reserved bit is always read as 0. The write value should always be 0.

51.3.3 Port Input Data Register (PIDR)

The PIDR register reflects the states of the individual input port pins.

	7	6	5	4	3	2	1	0
	PIDR7	PIDR6	PIDR5	PIDR4	PIDR3	PIDR2	PIDR1	PIDR0
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	PIDR7	0	R	0: Low input 1: High input
6	PIDR6	0	R	
5	PIDR5	0	R	
4	PIDR4	0	R	
3	PIDR3	0	R	
2	PIDR2	0	R	
1	PIDR1	0	R	
0	PIDR0	0	R	

Note: If PORTm.PDR is set to 10 or 11, the pin states of ports m can be read with PORTm.PIDR, regardless of the values of PORTm.PMR. The bit corresponding to a pin that does not exist is reserved. A reserved bit is always read as 0.

51.3.4 Port Mode Register (PMR)

The PMR register specifies the function of the pins of the port.

	7	6	5	4	3	2	1	0
	PMR7	PMR6	PMR5	PMR4	PMR3	PMR2	PMR1	PMR0
Initial value:	0	0	0	0	0	0	0	0

Note 1. The initial value of the PMR register of PORTM is 2’b11.

Bit	Bit Name	Initial value	R/W	Description
7	PMR7	0	R/W	0: Uses the pin as a general I/O pin. 1: Uses the pin as an I/O port for peripheral functions.
6	PMR6	0	R/W	
5	PMR5	0	R/W	
4	PMR4	0	R/W	
3	PMR3	0	R/W	
2	PMR2	0	R/W	
1	PMR1	0	R/W	
0	PMR0	0	R/W	

Note: Each bit of PORTm.PMR corresponds to each pin of port m; pin function can be specified in 1-bit units. The bit corresponding to a pin that does not exist is reserved. A reserved bit is always read as 0. The write value should always be 0.

51.3.5 GPIO Driving Ability Control Register (DSCR)

The DSCR register controls driving ability of the ports. For pins PG_2, PG_3, PG_4, PG_5, PG_6, PG_7, PJ_0, PJ_1, PJ_2, PJ_3, PJ_4, PJ_5, and PJ_6, setting of 01 (normal driving ability) or 11 (high driving ability) is only possible. For pins other than those above, setting of 01 (normal driving ability) is only possible.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSCR7[1:0]		DSCR6[1:0]		DSCR5[1:0]		DSCR4[1:0]		DSCR3[1:0]		DSCR2[1:0]		DSCR1[1:0]		DSCR0[1:0]	
Initial value:	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Bit	Bit Name	Initial value	R/W	Description
15 to 14	DSCR7[1:0]	01 *2	R/W	Odd bit Even bit Driving ability 0 1: Normal driving ability (4 mA) 1 1: High driving ability (8 mA) *1 0 0: Setting prohibited.*3 1 0: Setting prohibited.
13 to 12	DSCR6[1:0]	01 *2	R/W	
11 to 10	DSCR5[1:0]	01 *2	R/W	
9 to 8	DSCR4[1:0]	01 *2	R/W	
7 to 6	DSCR3[1:0]	01 *2	R/W	
5 to 4	DSCR2[1:0]	01 *2	R/W	
3 to 2	DSCR1[1:0]	01 *2	R/W	
1 to 0	DSCR0[1:0]	01 *2	R/W	

m = 0 to *

Note 1. For pins other than PG_2, PG_3, PG_4, PG_5, PG_6, PG_7, PJ_0, PJ_1, PJ_2, PJ_3, PJ_4, PJ_5, and PJ_6, setting of 01 (normal driving ability) is only possible and setting of 11 (high driving ability) is prohibited. The write value should always be 01.
Note 2. The initial value of pins PG_2, PG_3, PG_4, PG_5, PG_6, PG_7, PJ_0, PJ_1, PJ_2, PJ_3, PJ_4, and PJ_5 is 11.
Note 3. The bit corresponding to a pin that is not present is reserved. A reserved bit is always read as 00. The write value should always be 00.

51.3.6 Write Protect Register (PWPR)

The PWPR register enables or disables write access to the PFS register and the PFSWE bit in this register.

	7	6	5	4	3	2	1	0
	B0WI	PFSWE	—	—	—	—	—	—
Initial value:	1	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	B0WI	1	R/W	0: Writing to the PFSWE bit is enabled 1: Writing to the PFSWE bit is disabled
6	PFSWE	0	R/W	0: Writing to the PFS register is disabled 1: Writing to the PFS register is enabled
5 to 0	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.

PFSWE Bit (PFS Register Write Enable)

Writing to PmnPFS register (m = 0 to 9, A to H, J to M, n = 0 to 7) is enabled only when the PFSWE bit is set to 1. To set the PFSWE bit to 1, write 1 to the PFSWE bit after writing 0 to the B0WI bit.

B0WI Bit (PFSWE Bit Write Disable)

Writing to the PFSWE bit is enabled only when the B0WI bit is set to 0.

51.3.7 P0n Pin Function Control Register (P0nPFS) (n = 0 to 6)

The P0nPFS register selects the pin function.
The P0nPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.5.

PSEL[2:0] bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL bit (pin interrupt input function selection bit)

Set this bit to 1 when using the P0n pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.5 Register Settings for Input/Output Pin Function in 324-Pin Products

PSEL[2:0] Register Setting Value	Pin						
	P0_0	P0_1	P0_2	P0_3	P0_4	P0_5	P0_6
000b (Value after reset)	Hi-Z						
001b	D0	D1	D2	D3	D4	D5	D6
010b	DRP24 *1	DRP25 *1	DRP26 *1	DRP27 *1	DRP28 *1	DRP29 *1	DRP30 *1
011b	DV0_DATA17	DV0_DATA18	DV0_DATA19	DV0_DATA20	DV0_DATA21	DV0_DATA22	DV0_DATA23
100b	MTIOC6B	MTIOC6C	MTIOC6D	MTIOC7A	MTIOC7B	MTIOC7C	MTIOC7D
101b	GTIOC3B	GTIOC4A	GTIOC4B	GTIOC6A	GTIOC6B	GTIOC7A	GTIOC7B
110b	—	—	—	—	—	—	—

Note: —: Setting prohibited.
Note 1. Only in products with a DRP

51.3.8 P1n Pin Function Control Register (P1nPFS) (n = 0 to 4)

The P1nPFS register selects the pin function.
The P1nPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.6.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the P1n pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.6 Register Settings for Input/Output Pin Function in 324-Pin Products

PSEL[2:0] Register Setting Value	Pin				
	P1_0	P1_1	P1_2	P1_3	P1_4
000b (Value after reset)	Hi-Z				
001b	D7	D8	D9	D10	D11
010b	DRP31 *1	MTIOC8A	MTIOC8B	MTIOC8C	MTIOC8D
011b	IRQ0	IRQ1	IRQ2	IRQ3	IRQ4
100b	CAN_CLK	CAN0RX	CAN0RX_DATARATE_EN	CAN0TX	CAN0TX_DATARATE_EN
101b	VBUSEN0	OVRCUR0	VBUSEN1	OTG_ID1	VBUSIN0
110b	—	—	—	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.
Note 1. Only in products with a DRP

51.3.9 P2n Pin Function Control Register (P2nPFS) (n = 0 to 3)

The P2nPFS register selects the pin function.
The P2nPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.7.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the P2n pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.7 Register Settings for Input/Output Pin Function in 324-Pin Products

PSEL[2:0] Register Setting Value	Pin			
	P2_0	P2_1	P2_2	P2_3
000b (Value after reset)	Hi-Z			
001b	D12	D13	D14	D15
010b	GTIOC6A	GTIOC6B	GTIOC7A	GTIOC7B
011b	IRQ5	IRQ6	IRQ7	WDTOVF#/PERROUT#
100b	CAN1RX	CAN1RX_RATARATE_EN	CAN1TX	CAN1TX_RATARATE_EN
101b	OTG_EXICEN0	OTG_ID0	VBUSIN1	OTG_EXICEN1
110b	—	—	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.
Note: The P23PFS register cannot be initialized by an internal power-on reset through the watchdog timer.

51.3.10 P3n Pin Function Control Register (P3nPFS) (n = 0 to 5)

The P3nPFS register selects the pin function.
The P3nPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.8, Table 51.9.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the P3n pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.8 Register Settings for Input/Output Pin Function in 324-/272-/256-Pin Products

PSEL[2:0] Register Setting Value	Pin					
	P3_0	P3_1	P3_2	P3_3	P3_4	P3_5
000b (Value after reset)	Hi-Z					
001b	OTG_EXICEN1	ET1_RXER	ET1_CRS	ET1_MDC	ET1_MDIO	ET1_RXD1
010b	NFDATA4	NFALE	NFRE#	NFWE#	NFRB#	NFCLE
011b	ET1_LINKSTA	VBUSEN0	CC1_Ra1	OTG_EXICEN0	CC2_Ra1	CC2_Ra0
100b	MTIC5W	CAN1RX	CAN1RX_ DATARATE_EN	CAN1TX	CAN1TX_ DATARATE_EN	CAN0TX_ DATARATE_EN
101b	IRQ3	RSPCK2	MOSI2	MISO2	SSL20	SSL00
110b	—	IRQ6	—	IRQ7	—	—
111b	—	RMII1_RXER	RMII1_CRS_DV	—	—	RMII1_RXD1

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.

Table 51.9 Register Settings for Input/Output Pin Function in 176-Pin Products

PSEL[2:0] Register Setting Value	Pin				
	P3_1	P3_2	P3_3	P3_4	P3_5
000b (Value after reset)	Hi-Z				
001b	ET1_RXER	ET1_CRS	ET1_MDC	ET1_MDIO	ET1_RXD1
010b	NFALE	NFRE#	NFWE#	NFRB#	NFCLE
011b	VBUSEN0	CC1_Ra1	OTG_EXICEN0	CC2_Ra1	CC2_Ra0
100b	CAN1RX	CAN1RX_ DATARATE_EN	CAN1TX	CAN1TX_ DATARATE_EN	CAN0TX_ DATARATE_EN
101b	RSPCK2	MOSI2	MISO2	SSL20	SSL00
110b	IRQ6	—	IRQ7	—	—
111b	RMII1_RXER	RMII1_CRS_DV	—	—	RMII1_RXD1

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.

Note: —: Setting prohibited.

51.3.11 P4n Pin Function Control Register (P4nPFS) (n = 0 to 7)

The P4nPFS register selects the pin function.
The P4nPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.10.

PSEL[2:0] Bit (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the P4n pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.10 Register Settings for Input/Output Pin Function

PSEL[2:0] Register Setting Value	Pin							
	P4_0	P4_1	P4_2	P4_3	P4_4	P4_5	P4_6	P4_7
000b (Value after reset)	Hi-Z							
001b	SCK0	RxD0	TxD0	RTS0#	CTS0#	ET0_LINKSTA	ET0_EXOUT/ ET0_SCLKIN	ET0_WOL
010b	TXOUT0P	TXOUT0M	TXOUT1P	TXOUT1M	TXOUT2P	TXOUT2M	TXCLKOUTP	TXDLKOUTM
011b	SCI_SCK1	SCI_RXD1	SCI_TXD1	SCI_CTS1# /RTS1#	SCI_CTS0# /RTS0#	SCI_RXD0	SCI_TXD0	SCI_SCK0
100b	SSIBCK1	SSIRxD1	SSITxD1	SSILRCK1	WDTOVF#/ PERROUT#	RxD4	TxD4	SCK4
101b	MTIOC8A	MTIOC8B	MTIOC8C	MTIOC8D	OTG_ EXICEN0	DREQ0	DACK0	TEND0
110b	IRQ0	IRQ1	IRQ2	IRQ3	—	—	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.
Note: The P44PFS register cannot be initialized by an internal power-on reset through the watchdog timer.

51.3.12 P5n Pin Function Control Register (P5nPFS) (n = 0 to 7)

The P5nPFS register selects the pin function.
The P5nPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.11.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the P5n pin as the pin interrupt function.
PSetting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.11 Register Settings for Input/Output Pin Function

PSEL[2:0] Register Setting Value	Pin							
	P5_0	P5_1	P5_2	P5_3	P5_4	P5_5	P5_6	P5_7
000b (Value after reset)	Hi-Z							
001b	AN000	AN001	AN002	AN003	AN004	AN005	AN006	AN007
010b	IRQ4	IRQ5	IRQ6	IRQ7	IRQ0	IRQ1	IRQ2	IRQ3
011b	SD0_CD	SD0_WP	VBUSIN0	OTG_ID0	SD1_CD	SD1_WP	—	—
100b	SD1_CD	SD1_WP	—	—	—	—	—	—
101b	—	—	—	—	—	—	—	—
110b	—	—	—	—	—	—	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.

51.3.13 P6n Pin Function Control Register (P6nPFS) (n = 0 to 7)

The P6nPFS register selects the pin function.
The P6nPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.
The ISEL bit to which IRQn is not specified is reserved. Be sure to write a value after a reset.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.12, Table 51.13.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the P6n pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.12 Register Settings for Input/Output Pin Function in 324-/272-/256-Pin Products

PSEL[2:0] Register Setting Value	Pin							
	P6_0	P6_1	P6_2	P6_3	P6_4	P6_5	P6_6	P6_7
000b (Value after reset)	Hi-Z							
001b	ADTRG#	ET0_TXEN	ET0_TXD0	ET0_TXD1	CS5#	CS3#	CS2#	WE0#/DQML
010b	—	VIO_CLK	VIO_VD	VIO_HD	DRP00 *1	DRP01 *1	DRP02 *1	DRP03 *1
011b	—	SCK3	RxD3	TxD3	LCD0_ TCON6	LCD0_ TCON5	LCD0_ TCON4	LCD0_ TCON3
100b	—	MTIOC2A	MTIOC2B	POE0	AUDIO_ CLK	AUDIO_ XOUT	DREQ0	DACK0
101b	—	—	OTG_ EXICEN1	—	SD1_CD	CC1_Rd0	CC1_Ra0	CC2_Rd0
110b	—	—	IRQ0	—	—	—	—	—
111b	—	RMII0_ TXD_EN	RMII0_ TXD0	RMII0_ TXD1	—	—	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.
Note 1. Only in products with a DRP

Table 51.13 Register Settings for Input/Output Pin Function in 176-Pin Products

PSEL[2:0] Register Setting Value	Pin			
	P6_0	P6_1	P6_2	P6_3
000b (Value after reset)	Hi-Z			
001b	ADTRG#	ET0_TXEN	ET0_TXD0	ET0_TXD1
010b	—	VIO_CLK	VIO_VD	VIO_HD
011b	—	SCK3	RxD3	TxD3
100b	—	MTIOC2A	MTIOC2B	POE0
101b	—	—	OTG_EXICEN1	—
110b	—	—	IRQ0	—
111b	—	RMII0_TXD_EN	RMII0_TXD0	RMII0_TXD1

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.

Note: —: Setting prohibited.

51.3.14 P7n Pin Function Control Register (P7nPFS) (n = 0 to 7)

The P7nPFS register selects the pin function.
The P7nPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.14, Table 51.15.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the P7n pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.14 Register Settings for Input/Output Pin Function in 324-Pin Products

PSEL[2:0] Register Setting Value	Pin							
	P7_0	P7_1	P7_2	P7_3	P7_4	P7_5	P7_6	P7_7
000b (Value after reset)	Hi-Z							
001b	WE1#/ DQMU	RD/WR#	CS4#	RAS#	CAS#	CKE	AH#	RD#
010b	DRP04 *1	DRP05 *1	DV0_CLK	DRP06 *1	DRP07 *1	DRP08 *1	DV0_ VSYNC	DV0_ HSYNC
011b	DV0_CLK	DV0_ VSYNC	LCD0_ TCON2	DV0_ HSYNC	DV0_ DATA0	DV0_ DATA1	LCD0_ TCON1	LCD0_ TCON0
100b	SCK1	RxD1	TEND0	TxD1	RTS1#	CTS1#	GTIOC3A	GTIOC3B
101b	CC1_Rd1	CC1_Ra1	CC2_Ra0	CC2_Rd1	CC2_Ra1	OVRCUR1	SCK0	RxD0
110b	—	—	—	—	—	—	—	—

Note: —: Setting prohibited.
Note 1. Only in products with a DRP

Table 51.15 Register Settings for Input/Output Pin Function in 272-/256-Pin Products

	Pin		
PSEL[2:0]Register Setting Value	P7_2	P7_6	P7_7
000b (Value after reset)	Hi-Z		
001b	CS4#	AH#	RD#
010b	DV0_CLK	DV0_VSYNC	DV0_HSYNC
011b	LCD0_TCON2	LCD0_TCON1	LCD0_TCON0
100b	TEND0	GTIOC3A	GTIOC3B
101b	CC2_Ra0	SCK0	RxD0
110b	—	—	—

Note: —: Setting prohibited.

51.3.15 P8n Pin Function Control Register (P8nPFS) (n = 0 to 7)

The P8nPFS register selects the pin function.
The P8nPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.16, Table 51.17.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the P8n pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.16 Register Settings for Input/Output Pin Function in 324-Pin Products

PSEL[2:0] Register Setting Value	Pin							
	P8_0	P8_1	P8_2	P8_3	P8_4	P8_5	P8_6	P8_7
000b (Value after reset)	Hi-Z							
001b	A0	A1	A2	A3	A4	A5	A6	A7
010b	DV0_DATA14	DRP23 *1	DRP22 *1	DRP21 *1	DRP20 *1	DRP19 *1	DRP18 *1	DRP17 *1
011b	LCD0_DATA9	DV0_ DATA16	DV0_ DATA15	DV0_ DATA14	DV0_ DATA13	DV0_ DATA12	DV0_ DATA11	DV0_ DATA10
100b	SCI_CTS1#/RTS1#	GTIOC5B	GTIOC5A	MTIOC6A	SSL00	MISO0	MOSI0	RSPCK0
101b	MTIOC8D	IRQ3	IRQ2	GTIOC3A	SSIRxD3	SSITxD3	SSILRCK3	SSIBCK3
110b	—	—	—	—	—	—	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.
Note 1. Only in products with a DRP

Table 51.17 Register Settings for Input/Output Pin Function in 272-/256-Pin Products

PSEL[2:0] Register Setting Value	Pin P8_0
000b (Value after reset)	Hi-Z
001b	A0
010b	DV0_DATA14
011b	LCD0_DATA9
100b	SCI_CTS1#/RTS1#
101b	MTIOC8D
110b	—

Note: —: Setting prohibited.

51.3.16 P9n Pin Function Control Register (P9nPFS) (n = 0 to 7)

The P9nPFS register selects the pin function.
The P9nPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.18.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the P9n pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.18 Register Settings for Input/Output Pin Function in 324-Pin Products

PSEL[2:0] Register Setting Value	Pin							
	P9_0	P9_1	P9_2	P9_3	P9_4	P9_5	P9_6	P9_7
000b (Value after reset)	Hi-Z							
001b	A8	A9	A10	A11	A12	A13	A14	A15
010b	DRP16 *1	DRP15 *1	DRP14 *1	DRP13 *1	DRP12 *1	DRP11 *1	DRP10 *1	DRP09 *1
011b	DV0_ DATA9	DV0_ DATA8	DV0_ DATA7	DV0_ DATA6	DV0_ DATA5	DV0_ DATA4	DV0_ DATA3	DV0_ DATA2
100b	TxD4	RxD4	SCK4	—	—	—	—	—
101b	SSIDATA2	SSILRCK2	SSIBCK2	SSIRxD0	SSITxD0	SSILRCK0	SSIBCK0	SD1_WP
110b	—	—	—	—	—	—	—	—

Note: —: Setting prohibited.
Note 1. Only in products with a DRP

51.3.17 PAn Pin Function Control Register (PAnPFS) (n = 0 to 7)

The PAnPFS register selects the pin function.
The PAnPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.19.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the PAn pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.19 Register Settings for Input/Output Pin Function in 324-/272-/256-Pin Products

PSEL[2:0] Register Setting Value	Pin							
	PA_0	PA_1	PA_2	PA_3	PA_4	PA_5	PA_6	PA_7
000b (Value after reset)	Hi-Z							
001b	A16	A17	A18	A19	A20	A21	A22	A23
010b	DV0_ DATA13	DV0_ DATA12	DV0_ DATA11	DV0_ DATA10	DV0_ DATA9	DV0_ DATA8	DV0_ DATA7	DV0_ DATA6
011b	LCD0_ DATA10	LCD0_ DATA11	LCD0_ DATA12	LCD0_ DATA13	LCD0_ DATA14	LCD0_ DATA15	LCD0_ DATA16	LCD0_ DATA17
100b	SCI_TXD1	SCI_RXD1	SCI_SCK1	SCI_CTS0#/ RTS0#	SCI_TXD0	SCI_RXD0	SCI_SCK0	SSIRxD1
101b	MTIOC8C	MTIOC8B	MTIOC8A	MTIOC0D	MTIOC0C	MTIOC0B	MTIOC0A	POE10
110b	—	IRQ6	—	—	—	IRQ5	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.

51.3.18 PBn Pin Function Control Register (PBnPFS) (n = 0 to 5)

The PBnPFS register selects the pin function.
The PBnPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.20.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the PBn pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.20 Register Settings for Input/Output Pin Function in 324-/272-/256-Pin Products

PSEL[2:0] Register Setting Value	Pin					
	PB_0	PB_1	PB_2	PB_3	PB_4	PB_5
000b (Value after reset)	Hi-Z					
001b	A24	A25	BS0#	CS0#	CS1#	WAIT#
010b	DV0_DATA5	DV0_DATA4	DV0_DATA3	DV0_DATA2	DV0_DATA1	DV0_DATA0
011b	LCD0_DATA18	LCD0_DATA19	LCD0_DATA20	LCD0_DATA21	LCD0_DATA22	LCD0_DATA23
100b	SSITxD1	SSILRCK1	SSIBCK1	SSIDATA2	SSILRCK2	SSIBCK2
101b	POE8	POE4	POE0	CTS0#	RTS0#	TxD0
110b	—	—	—	—	—	—

Note: —: Setting prohibited.

51.3.19 PCn Pin Function Control Register (PCnPFS) (n = 0 to 7)

The PCnPFS register selects the pin function.
The PCnPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.21.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the PCn pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.21 Register Settings for Input/Output Pin Function 324-/272-/256-Pin Products

PSEL[2:0] Register Setting Value	Pin							
	PC_0	PC_1	PC_2	PC_3	PC_4	PC_5	PC_6	PC_7
000b (Value after reset)	Hi-Z							
001b	VBUSIN1	VBUSIN0	OTG_ EXICEN0	OTG_ID0	OTG_ID1	VBUSEN1	VBUSEN0	OVRCUR0
010b	NFDATA5	NFDATA6	NFDATA7	NFCLE	NFALE	NFRE#	NFWE#	NFRB#
011b	ET1_TXCLK	ET1_TXD2	ET1_TXD3	ET1_COL	ET1_TXER	ET1_RXDV	ET1_RXD2	ET1_RXD3
100b	RSPCK2	MOSI2	MISO2	SSL20	SPDIF_IN	SPDIF_OUT	SD1_CD	SD1_WP
101b	IRQ2	LCD0_ TCON6	LCD0_ TCON5	LCD0_ TCON4	LCD0_ TCON3	LCD0_ TCON2	LCD0_ TCON1	LCD0_ TCON0
110b	—	—	—	—	IRQ1	IRQ0	IRQ7	IRQ6

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.

51.3.20 PDn Pin Function Control Register (PDnPFS) (n = 0 to 7)

The PDnPFS register selects the pin function.
The PDnPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.22.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the PDn pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.22 Register Settings for Input/Output Pin Function

PSEL[2:0] Register Setting Value	Pin							
	PD_0	PD_1	PD_2	PD_3	PD_4	PD_5	PD_6	PD_7
000b (Value after reset)	Hi-Z							
001b	RIIC0SCL	RIIC0SDA	RIIC1SCL	RIIC1SDA	RIIC2SCL	RIIC2SDA	RIIC3SCL	RIIC3SDA
010b	IRQ0	IRQ1	IRQ2	IRQ3	IRQ4	IRQ5	IRQ6	IRQ7
011b	MTCLKA	MTCLKB	MTCLKC	MTCLKD	—	—	—	—
100b	GTETRGA	GTETRGB	GTETRGC	GTETRGD	—	—	—	—
101b	—	—	—	—	—	—	—	—
110b	—	—	—	—	—	—	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.

51.3.21 PEn Pin Function Control Register (PEnPFS) (n = 0 to 6)

The PEnPFS register selects the pin function.
The PEnPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.23.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the PEn pin as the pin interrupt function.
PSetting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.23 Register Settings for Input/Output Pin Function

PSEL[2:0] Register Setting Value	Pin						
	PE_0	PE_1	PE_2	PE_3	PE_4	PE_5	PE_6
000b (Value after reset)	Hi-Z						
001b	ET0_RXCLK	ET0_RXD0	ET0_RXD1	ET0_RXER	ET0_CRS	ET0_MDC	ET0_MDIO
010b	VIO_FLD	VIO_D7	VIO_D6	VIO_D5	VIO_D4	VIO_D3	VIO_D2
011b	SCK2	RxD2	TxD2	SSIBCK0	SSILRCK0	SSITxD0	SSIRxD0
100b	POE4	POE8	POE10	MTIOC0A	MTIOC0B	MTIOC0C	MTIOC0D
101b	—	VBUSIN1	—	—	—	CC1_Rd1	CC2_Rd1
110b	—	IRQ1	—	—	—	—	—
111b	REF50CK0	RMII0_RXD0	RMII0_RXD1	RMII0_RXER	RMII0_CRS_DV	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.

51.3.22 PFn Pin Function Control Register (PFnPFS) (n = 0 to 7)

The PFnPFS register selects the pin function.
The PFnPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.24.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the PFn pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.24 Register Settings for Input/Output Pin Function

PSEL[2:0] Register Setting Value	Pin							
	PF_0	PF_1	PF_2	PF_3	PF_4	PF_5	PF_6	PF_7
000b (Value after reset)	Hi-Z							
001b	SCK3	RxD3	TxD3	SCK2	RxD2	TxD2	RTS2#	GTETRGD
010b	DV0_DATA15	DV0_DATA16	DV0_DATA17	DV0_DATA18	DV0_DATA19	DV0_DATA20	DV0_DATA21	DV0_DATA23
011b	LCD0_DATA8	LCD0_DATA7	LCD0_DATA6	LCD0_DATA5	LCD0_DATA4	LCD0_DATA3	LCD0_DATA2	LCD0_DATA0
100b	MTIOC7A	MTIOC7B	MTIOC7C	MTIOC7D	MTIOC6A	MTIOC6B	MTIOC6C	MTCLKD
101b	RSPCK1	MOSI1	MISO1	SSL10	SSIBCK0	SSILRCK0	SSITxD0	IRQ1
110b	—	IRQ4	—	—	IRQ1	—	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.

51.3.23 PGn Pin Function Control Register (PGnPFS) (n = 0 to 7)

The PGnPFS register selects the pin function.
The PGnPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.25.

PSEL[2:0] bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the PGn pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.25 Register Settings for Input/Output Pin Function

PSEL[2:0] Register Setting Value	Pin							
	PG_0	PG_1	PG_2	PG_3	PG_4	PG_5	PG_6	PG_7
000b (Value after reset)	Hi-Z							
001b	ET0_TXCLK	ET0_TXD2	ET0_TXD3	ET0_COL	ET0_TXER	ET0_RXDV	ET0_RXD2	ET0_RXD3
010b	VIO_D8	VIO_D9	VIO_D10	VIO_D11	VIO_D15	VIO_D14	VIO_D13	VIO_D12
011b	RSPCK0	MOSI0	MISO0	SSL00	RSPCK1	MOSI1	MISO1	SSL10
100b	MTIOC3A	MTIOC3C	MTIOC3B	MTIOC3D	MTIOC4A	MTIOC4B	MTIOC4C	MTIOC4D
101b	HM_RSTO#	HM_INT#/ OM_ECS#	GTIOC0A	GTIOC0B	GTIOC1A	GTIOC1B	GTIOC2A	GTIOC2B
110b	—	—	IRQ4	—	—	—	IRQ5	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.

51.3.24 PHn Pin Function Control Register (PHnPFS) (n = 0 to 6)

The PHnPFS register selects the pin function.
The PHnPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.26, Table 51.27.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the PHn pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.26 Register Settings for Input/Output Pin Function in 324-/272-/256-Pin Products

PSEL[2:0] Register Setting Value	Pin						
	PH_0	PH_1	PH_2	PH_3	PH_4	PH_5	PH_6
000b (Value after reset)	Hi-Z						
001b	AUDIO_CLK	AUDIO_XOUT	CTS2	HM_RSTO#	HM_INT#/ OM_ECS#	HM_RSTO#	HM_INT#/ OM_ECS#
010b	VIO_D1	VIO_D0	DV0_DATA22	RTS2#	CTS2#	NFDATA2	NFDATA3
011b	GTIOC4A	GTIOC4B	LCD0_DATA1	GTIOC6A	GTIOC6B	ET1_EXOUT/ ET1_SCLKIN	ET1_WOL
100b	MTIOC1A	MTIOC1B	MTIOC6D	MTIOC2A	MTIOC2B	MTIC5U	MTIC5V
101b	CC1_Rd0	CC2_Rd0	SSIRxD0	SD0_CD	SD0_WP	IRQ5	IRQ4
110b	IRQ3	IRQ2	—	IRQ3	IRQ2	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.

Table 51.27 Register Settings for Input/Output Pin Function in 176-Pin Products

PSEL[2:0] Register Setting Value	Pin			
	PH_0	PH_1	PH_3	PH_4
000b (Value after reset)	Hi-Z			
001b	AUDIO_CLK	AUDIO_XOUT	HM_RSTO#	HM_INT#/ OM_ECS#
010b	VIO_D1	VIO_D0	RTS2#	CTS2#
011b	GTIOC4A	GTIOC4B	GTIOC6A	GTIOC6B
100b	MTIOC1A	MTIOC1B	MTIOC2A	MTIOC2B
101b	CC1_Rd0	CC2_Rd0	SD0_CD	SD0_WP
110b	IRQ3	IRQ2	IRQ3	IRQ2

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.

Note: —: Setting prohibited.

51.3.25 PJn Pin Function Control Register (PJnPFS) (n = 0 to 7)

The PJnPFS register selects the pin function.
The PJnPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.28, Table 51.29.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the PJn pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.28 Register Settings for Input/Output Pin Function in 324-/272-/256-Pin Products

PSEL[2:0] Register Setting Value	Pin							
	PJ_0	PJ_1	PJ_2	PJ_3	PJ_4	PJ_5	PJ_6	PJ_7
000b (Value after reset)	Hi-Z							
001b	TRACE CLK	TRACE CTL	TRACE DATA0	TRACE DATA1	TRACE DATA2	TRACE DATA3	GTETRGC	GTETRGB
010b	SPDIF_OUT	SPDIF_IN	NFCE#	NFDATA0	NFDATA1	NFDATA2	NFCE#	NFDATA0
011b	—	—	—	—	—	OVRCUR0	LCD0_ CLK	LCD0_ EXTCLK
100b	SCK1	RxD1	TxD1	RTS1#	CTS1#	MTIOC1A	MTCLKC	MTCLKB
101b	SSIRxD3	VBUSIN0	SSITxD3	SSILRCK3	SSIBCK3	SSILRCK2	IRQ0	—
110b	—	IRQ0	—	—	—	IRQ4	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.

Table 51.29 Register Settings for Input/Output Pin Function in 176-Pin Products

PSEL[2:0] Register Setting Value	Pin					
	PJ_0	PJ_1	PJ_2	PJ_3	PJ_4	PJ_5
000b (Value after reset)	Hi-Z					
001b	TRACECLK	TRACECTL	TRACEDATA0	TRACEDATA1	TRACEDATA2	TRACEDATA3
010b	SPDIF_OUT	SPDIF_IN	NFCE#	NFDATA0	NFDATA1	NFDATA2
011b	—	—	—	—	—	OVRCUR0
100b	SCK1	RxD1	TxD1	RTS1#	CTS1#	MTIOC1A
101b	SSIRxD3	VBUSIN0	SSITxD3	SSILRCK3	SSIBCK3	SSILRCK2
110b	—	IRQ0	—	—	—	IRQ4

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.

Note: —: Setting prohibited.

51.3.26 PKn Pin Function Control Register (PKnPFS) (n = 0 to 5)

The PKnPFS register selects the pin function.
The PKnPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.30, Table 51.31.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the PKn pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.30 Register Settings for Input/Output Pin Function in 324-/272-/256-Pin Products

PSEL[2:0] Register Setting Value	Pin					
	PK_0	PK_1	PK_2	PK_3	PK_4	PK_5
000b (Value after reset)	Hi-Z					
001b	ET1_TXEN	ET1_TXD0	ET1_TXD1	ET1_RXCLK	ET1_RXD0	GTETRGA
010b	NFDATA3	NFDATA4	NFDATA5	NFDATA6	NFDATA7	NFDATA1
011b	CC1_Rd0	CC1_Ra0	VBUSEN1	CC2_Rd0	OVRCUR1	WDTOVF#/ PERROUT#
100b	MTIOC1B	CAN_CLK	CAN0RX	CAN0RX_ DATARATE_EN	CAN0TX	MTCLKA
101b	SSIBCK2	SSIDATA2	RSPCK0	MOSI0	MISO0	—
110b	—	—	IRQ5	—	IRQ6	—
111b	RMII1_TXD_EN	RMII1_TXD0	RMII1_TXD1	REF50CK1	RMII1_RXD0	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.
Note: The PK5PFS register cannot be initialized by an internal power-on reset through the watchdog timer.

Table 51.31 Register Settings for Input/Output Pin Function in 176-Pin Products

PSEL[2:0] Register Setting Value	Pin				
	PK_0	PK_1	PK_2	PK_3	PK_4
000b (Value after reset	Hi-Z				
001b	ET1_TXEN	ET1_TXD0	ET1_TXD1	ET1_RXCLK	ET1_RXD0
010b	NFDATA3	NFDATA4	NFDATA5	NFDATA6	NFDATA7
011b	CC1_Rd0	CC1_Ra0	VBUSEN1	CC2_Rd0	OVRCUR1
100b	MTIOC1B	CAN_CLK	CAN0RX	CAN0RX_ DATARATE_EN	CAN0TX
101b	SSIBCK2	SSIDATA2	RSPCK0	MOSI0	MISO0
110b	—	—	IRQ5	—	IRQ6
111b	RMII1_TXD_EN	RMII1_TXD0	RMII1_TXD1	REF50CK1	RMII1_RXD0

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.

Note: —: Setting prohibited.

51.3.27 PLn Pin Function Control Register (PLnPFS) (n = 0 to 4)

The PLnPFS register selects the pin function.
The PLnPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.32.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bit (pin interrupt input function selection bit)

Set this bit to 1 when using the PLn pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.

Table 51.32 Register Settings for Input/Output Pin Function

PSEL[2:0] Register Setting Value	Pin				
	PL_0	PL_1	PL_2	PL_3	PL_4
000b (Value after reset)	Hi-Z				
001b	—	—	—	—	—
010b	—	—	—	—	—
011b	—	—	—	—	—
100b	—	—	—	—	—
101b	IRQ4	IRQ5	IRQ6	IRQ7	IRQ0
110b	—	—	—	—	—

Note: Set the IRQMSK bit of the interrupt request register (IRQRR) to 1 when using an IRQ interrupt for release from software standby. For details, refer to section 7, Interrupt Controller.
Note: —: Setting prohibited.

51.3.28 PMn Pin Function Control Register (PMnPFS) (n = 0 to 1)

The PMnPFS register selects the pin function.
The PMnPFS register is protected by the Write Protect Register (PWPR). Modify the register after releasing the protection.

	7	6	5	4	3	2	1	0
	—	ISEL	—	—	—	PSEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
6	ISEL	0	R/W	0: Not used as an interrupt input pin. 1: Used as an interrupt input pin.
5 to 3	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	PSEL[2:0]	000	R/W	These bits select the peripheral function. For individual pin functions, see Table 51.33.

PSEL[2:0] Bits (pin function selection bits)

These bits set the peripheral functions which are allocated to pins.

ISEL Bits (pin interrupt input function selection bit)

Set this bit to 1 when using the JP0_0 pin as the pin interrupt function.
Setting 0 here leads to the interrupt controller stopping control over pin interrupts.
As the JP0_1 pin is exclusively used as an output pin, the setting of this bit has no effect.

Table 51.33 Register Settings for Input/Output Pin Function

PSEL[2:0] Register Setting Value	Pin	
	JP0_0	JP0_1
000b	Hi-Z	
001b (Value after reset)	TDI	TDO/SWO
010b	—	—
011b	—	—
100b	—	—
101b	—	—
110b	—	—

Note: —: Setting prohibited.

51.3.29 Ethernet Control Register (PFENET)

The Ethernet control register (PFENET) selects an Ethernet interface mode to be used.

PHYMODE0

This bit specifies the Ethernet interface mode of the ETHERC channel 0. Select the same mode as the one specified by the pin function select bits (PmnPFS.PSEL[2:0]). When the signals for the RMII mode have been specified by the PmnPFS.PSEL[2:0] bits, set the PHYMODE bit to 0 (RMII mode), whereas when the signals for the MII mode have been specified by the PmnPFS.PSEL[2:0] bits, set the PHYMODE bit to 1 (MII mode).

PHYMODE1

This bit specifies the Ethernet interface mode of the ETHERC channel 1. Select the same mode as the one specified by the pin function select bits (PmnPFS.PSEL[2:0]). When the signals for the RMII mode have been specified by the PmnPFS.PSEL[2:0] bits, set the PHYMODE bit to 0 (RMII mode), whereas when the signals for the MII mode have been specified by the PmnPFS.PSEL[2:0] bits, set the PHYMODE bit to 1 (MII mode).

	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PHY MODE1	PHY MODE0
Initial value:	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
7 to 2	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
1	PHYMODE1	0	R/W	Sets the Ethernet interface mode of channel 1. 0: RMII mode (ETHERC channel 1) 1: MII mode (ETHERC channel 1)
0	PHYMODE0	0	R/W	Sets the Ethernet interface mode of channel 0 0: RMII mode (ETHERC channel 0) 1: MII mode (ETHERC channel 0)

51.3.30 Dedicated Pin POC Control Register (PPOC)

The PPOC register selects the operating voltage of the pins only used for the SPI multi I/O bus controller and SD/MMC host interfaces. This register should be set to match the voltage applied to the power supply for dedicated pins before the dedicated pins are used.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	POC SEL0	—	—	—	—	POC3	POC2	—	POC0
Initial value:	0	0	0	0	0	0	0	1*	0	0	0	0	1*	1*	X	1*

Note: * In boot mode 3 or 4, different values are set according to the boot program.
For details, see section 3.5.4, Boot Mode 3, and section 3.5.5, Boot Mode 4.

Bit	Bit Name	Initial value	R/W	Description
31 to 9	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
8	POCSEL0	1*	R/W	See the description of the POC0 bit.
7 to 4	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
3	POC3	1*	R/W	0: The dedicated pins of the SD/MMC host interface 1 operate from a 1.8-V power supply. 1: The dedicated pins of the SD/MMC host interface 1 operate from a 3.3-V power supply.
2	POC2	1*	R/W	0: The dedicated pins of the SD/MMC host interface 0 operate from a 1.8-V power supply. 1: The dedicated pins of the SD/MMC host interface 0 operate from a 3.3-V power supply.
1	—	X	R/W	Reserved When read, an undefined value is returned.
0	POC0	1*	R/W	[POCSEL0, POC0] 00, 01, or 10: The dedicated pins of the SPI multi I/O bus controller operate from a 1.8-V power supply. 11: The dedicated pins of the SPI multi I/O bus controller operate from a 3.3-V power supply.

51.3.31 SD/MMC Host Interface Dedicated Pin Driving Ability Control Register 0 (PSDMMC0)

The PSDMMC0 register specifies the driving ability of the pins of the SD/MMC host interface 0.
This register should be set in accord with the POC2 bit of the PPOC register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SD0_CLK_TDSEL		SD0_DAT3_DRV		SD0_DAT2_DRV		SD0_DAT1_DRV		SD0_DAT0_DRV		SD0_CMD_DRV		SD0_CLK_DRV	
Initial value:	0	0	1	1	1	0	1	0	1	0	1	0	1	0	1	0

Bit	Bit Name	Initial value	R/W	Description
31	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
30	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
29, 28	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
27	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
26, 25	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
24	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
23, 22	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
21	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
20, 19	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
18	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
17 to 14	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	SD0_CLK_TDSEL	11	R/W	SD0_CLK Clock Control Register When the SD/MMC host interface 0 operates from a 1.8-V power supply: Odd bit Even bit 0 1: Operates from a 1.8-V power supply Others: Setting prohibited When the SD/MMC host interface 0 operates from a 3.3-V power supply: Odd bit Even bit 1 1: Operates from a 3.3-V power supply Others: Setting prohibited

Bit	Bit Name	Initial value	R/W	Description
11, 10	SD0_DAT3_DRV	10	R/W	SD0_DAT3 Port Driving Ability Control Register When the SD/MMC host interface 0 operates from a 1.8-V power supply: Odd bit Even bit 1 1: 6 mA Others: Setting prohibited When the SD/MMC host interface 0 operates from a 3.3-V power supply: Odd bit Even bit 1 0: 6mA Others: Setting prohibited
9, 8	SD0_DAT2_DRV	10	R/W	SD0_DAT2 Port Driving Ability Control Register
7, 6	SD0_DAT1_DRV	10	R/W	SD0_DAT1 Port Driving Ability Control Register
5, 4	SD0_DAT0_DRV	10	R/W	SD0_DAT0 Port Driving Ability Control Register
3, 2	SD0_CMD_DRV	10	R/W	SD0_CMD Port Driving Ability Control Register
1, 0	SD0_CLK_DRV	10	R/W	SD0_CLK Port Driving Ability Control Register When the SD/MMC host interface 0 operates from a 1.8-V power supply: Odd bit Even bit 1 1: 12 mA Others: Setting prohibited When the SD/MMC host interface 0 operates from a 3.3-V power supply: Odd bit Even bit 1 0: 12 mA Others: Setting prohibited

51.3.32 SD/MMC Host Interface Dedicated Pin Driving Ability Control Register 1 (PSDMMC1)

The PSDMMC1 register specifies the driving ability of the pins of the SD/MMC host interface 0.
This register should be set in accord with the POC2 bit of the PPOC register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SD0_RSTN_DRV		SD0_DAT7_DRV		SD0_DAT6_DRV		SD0_DAT5_DRV		SD0_DAT4_DRV	
Initial value:	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0

Bit	Bit Name	Initial value	R/W	Description
31	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
30	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
29, 28	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
27	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
26, 25	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
24	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
23, 22	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
21	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
20, 19	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
18	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
17 to 10	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	SD0_RSTN_DRV	10	R/W	SD0_RST# Port Driving Ability Control Register
7, 6	SD0_DAT7_DRV	10	R/W	SD0_DAT7 Port Driving Ability Control Register
5, 4	SD0_DAT6_DRV	10	R/W	SD0_DAT6 Port Driving Ability Control Register
3, 2	SD0_DAT5_DRV	10	R/W	SD0_DAT5 Port Driving Ability Control Register
1, 0	SD0_DAT4_DRV	10	R/W	SD0_DAT4 Port Driving Ability Control Register

When the SD/MMC host interface 0 operates from a 1.8-V power supply:

Odd bit	Even bit
1	1: 6 mA

Others: Setting prohibited

When the SD/MMC host interface 0 operates from a 3.3-V power supply:

Odd bit	Even bit
1	0: 6 mA

Others: Setting prohibited

51.3.33 SD/MMC Host Interface Dedicated Pin Driving Ability Control Register 2 (PSDMMC2)

The PSDMMC2 register specifies the driving ability of the pins of the SD/MMC host interface 1.
This register should be set in accord with the POC3 bit of the PPOC register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SD1_CLK_TDSEL		SD1_DAT3_DRV		SD1_DAT2_DRV		SD1_DAT1_DRV		SD1_DAT0_DRV		SD1_CMD_DRV		SD1_CLK_DRV	
Initial value:	0	0	1	1	1	0	1	0	1	0	1	0	1	0	1	0

Bit	Bit Name	Initial value	R/W	Description
31	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
30	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
29, 28	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
27	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
26, 25	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
24	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
23, 22	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
21	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
20, 19	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
18	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
17 to 14	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	SD1_CLK_TDSEL	11	R/W	SD1_CLK Clock Control Register When the SD/MMC host interface 1 operates from a 1.8-V power supply: Odd bit Even bit 0 1: Operates from a 1.8-V power supply Others: Setting prohibited When the SD/MMC host interface 1 operates from a 3.3-V power supply: Odd bit Even bit 1 1: Operates from a 3.3-V power supply Others: Setting prohibited

Bit	Bit Name	Initial value	R/W	Description
11, 10	SD1_DAT3_DRV	10	R/W	SD1_DAT3 Port Driving Ability Control Register When the SD/MMC host interface 1 operates from a 1.8-V power supply: Odd bit Even bit 1 1: 6 mA Others: Setting prohibited When the SD/MMC host interface 1 operates from a 3.3-V power supply: Odd bit Even bit 1 0: 6 mA Others: Setting prohibited
9, 8	SD1_DAT2_DRV	10	R/W	SD1_DAT2 Port Driving Ability Control Register
7, 6	SD1_DAT1_DRV	10	R/W	SD1_DAT1 Port Driving Ability Control Register
5, 4	SD1_DAT0_DRV	10	R/W	SD1_DAT0 Port Driving Ability Control Register
3, 2	SD1_CMD_DRV	10	R/W	SD1_CMD Port Driving Ability Control Register
1, 0	SD1_CLK_DRV	10	R/W	SD1_CLK Port Driving Ability Control Register When the SD/MMC host interface 1 operates from a 1.8-V power supply: Odd bit Even bit 1 1: 12 mA Others: Setting prohibited When the SD/MMC host interface 1 operates from a 3.3-V power supply: Odd bit Even bit 1 0: 12 mA Others: Setting prohibited

51.3.34 SPI Multi I/O Bus Controller Dedicated Pin Driving Ability Control Register (PSPIBSC)

The PSPIBSC register specifies the driving ability of the pins of the SPI multi I/O bus controller.
This register should be set in accord with the POCSEL0 and POC0 bits of the PPOC register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	QSPI1_SSL_DRV		QSPI1_IO3_DRV		QSPI1_IO2_DRV		QSPI1_IO1_DRV		QSPI1_IO0_DRV		QSPI1_SPCLK_DRV	
Initial value:	0	0	0	0	1*	1*	1*	1*	1*	1*	1*	1*	1*	1*	1*	1*

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RPC_WPN_DRV		RPC_RESETN_DRV		QSPI0_SSL_DRV		QSPI0_IO3_DRV		QSPI0_IO2_DRV		QSPI0_IO1_DRV		QSPI0_IO0_DRV		QSPI0_SPCLK_DRV	
Initial value:	1*	1*	1*	1*	1*	1*	1*	1*	1*	1*	1*	1*	1*	1*	1*	1*

Note: * In boot mode 3, different values are set according to the boot program.
For details, see section 3.5.4, Boot Mode 3.

Bit	Bit Name	Initial value	R/W	Description
31 to 28	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
27, 26	QSPI1_SSL_DRV	11*	R/W	QSPI1_SSL Port Driving Ability Control Register
25, 24	QSPI1_IO3_DRV	11*	R/W	QSPI1_IO3 Port Driving Ability Control Register
23, 22	QSPI1_IO2_DRV	11*	R/W	QSPI1_IO2 Port Driving Ability Control Register
21, 20	QSPI1_IO1_DRV	11*	R/W	QSPI1_IO1 Port Driving Ability Control Register
19, 18	QSPI1_IO0_DRV	11*	R/W	QSPI1_IO0 Port Driving Ability Control Register
17, 16	QSPI1_SPCLK_DRV	11*	R/W	QSPI1_SPCLK Port Driving Ability Control Register
15, 14	RPC_WPN_DRV	11*	R/W	RPC_WP# Port Driving Ability Control Register
13, 12	RPC_RESETN_DRV	11*	R/W	RPC_RESET# Port Driving Ability Control Register
11, 10	QSPI0_SSL_DRV	11*	R/W	QSPI0_SSL Port Driving Ability Control Register
9, 8	QSPI0_IO3_DRV	11*	R/W	QSPI0_IO3 Port Driving Ability Control Register
7, 6	QSPI0_IO2_DRV	11*	R/W	QSPI0_IO2 Port Driving Ability Control Register
5, 4	QSPI0_IO1_DRV	11*	R/W	QSPI0_IO1 Port Driving Ability Control Register
3, 2	QSPI0_IO0_DRV	11*	R/W	QSPI0_IO0 Port Driving Ability Control Register
1, 0	QSPI0_SPCLK_DRV	11*	R/W	QSPI0_SPCLK Port Driving Ability Control Register

When the SPI multi I/O bus controller operates from a 1.8-V power supply:
Odd bit Even bit
1 1: 12 mA
Others: Setting prohibited

When the SPI multi I/O bus controller operates from a 3.3-V power supply:
Odd bit Even bit
0 1: 8 mA
Others: Setting prohibited

Note: The initial value of these bits (odd and even bits) is as follows:
Boot mode 3: 01
Others: 11

51.3.35 HyperBus Controller and Octa Memory Controller Dedicated Pin Control Register (PHMOM0)

PHMOM0 selects HyperBus Controller or Octa Memory Controller.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	HOSEL
Initial value:	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

Bit	Bit Name	Initial value	R/W	Description
31 to 30	—	00	R/W	Reserved These bits are always read as 0. The write value should always be 0.
29 to 16	—	All 1	R/W	Reserved These bits are always read as 1. The write value should always be 1.
15	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
14 to 1	—	All 1	R/W	Reserved These bits are always read as 1. The write value should always be 1.
0	HOSEL	0	R/W	0: Select HyperBus controller 1: Select Octa memory controller

51.3.36 Peripheral Pin Function Control Register (PMODEPFS)

The PMODEPFS register switches the ETHER pin function between EXOUT (output) and SCLKIN (input), and selects the output function of the VBUSEN pin of the USB.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	VBUS1_SEL	VBUS0_SEL	ET1_EXOUT_SEL	ET0_EXOUT_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Bit Name	Initial value	R/W	Description
31 to 4	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
3	VBUS1_SEL	0	R/W	Selects the output function of the VBUSEN1 pin of the channel 1 USB 2.0 host module. 1: The VBUSEN1 pin is controlled by the VBOUT bit of the VBCTRL register. 0: The VBUSEN1 pin is controlled by the PP bit of the PORTSC1 register.
2	VBUS0_SEL	0	R/W	Selects the output function of the VBUSEN0 pin of the channel 0 USB 2.0 host module. 1: The VBUSEN0 pin is controlled by the VBOUT bit of the VBCTRL register. 0: The VBUSEN0 pin is controlled by the PP bit of the PORTSC1 register.
1	ET1_EXOUT_SEL	0	R/W	Selects EXOUT or SCLKIN for ETHER1. 1: SCLKIN1 input 0: EX1 EXOUT output
0	ET0_EXOUT_SEL	0	R/W	Selects EXOUT or SCLKIN for ETHER0 1: SCLKIN0 input 0: EX0 EXOUT output

51.3.37 CKIO pin Driving Ability Control Register (PCKIO)

The PCKIO register controls driving ability of the CKIO pin.

	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CKIO_DRV	
Initial value:	0	0	0	0	0	0	0	1

Bit	Bit Name	Initial value	R/W	Description
7 to 2	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	CKIO_DRV	01	R/W	These bits control the driving ability of the CKIO pin. 00: Setting prohibited. 01: 8 mA (recommended for 66-MHz operation) 10: 12 mA (recommended for 132-MHz operation) 11: Setting prohibited.

51.4 Usage Notes

51.4.1 Procedure for Specifying Input/Output Pin Function

Use the following procedure to specify the input/output pin functions.

1. Set the bits corresponding to the given pin in the port direction register (PDR) and the port mode register (PMR) to 00 and 0, respectively to set the pin function as the general-purpose I/O port pin.
2. Specify the assignments of input/output signals for peripheral modules to the desired pins.
3. Setting the PWPR.PFSWE bit to 1 after setting the PWPR.B0WI bit to 0 leads to the Pmn pin function control register (PmnPFS) (m = 0 to 9, A to H, J to M, n = 0 to 7) being writable.
4. Specify the input/output function for the pin through the PSEL[2:0] bit settings in the PmnPFS register.
5. Set the PFSWE bit in the PWPR register to 0 to disable writing to the PmnPFS register.
6. Set the PMR to 1 as necessary to switch to the selected input/output function for the pin.
7. The level of port pins can be read by setting the corresponding bits in the PDR register to 10 as required.

51.4.2 Notes on PFS Register Setting

1. Settings of the Pmn Pin Function Control Register (PmnPFS) (m = 0 to 9, A to H, J to M, n = 0 to 7) should be made only while the PMR register for the target pin is set to 0. If a PmnPFS register is set while the PMR register is 1, unexpected edges may be input through the input pin or unexpected pulses are output through the output pin.
2. When the PmnPFS.ISEL bit is set to 1 while pin interrupts are used, the pin state is input because input to the target pin is enabled regardless of the PDR register setting. Therefore, the following conditions must be satisfied.
   • Follow the procedure for setting up the interrupt controller. For details, see Figure 7.3, Flow of Switching Pin Functions.
   • Do not set the PmnPFS.ISEL bit to 1 for a pin for which the output function was selected.
   If PmnPFS.ISEL bit is set with a wrong procedure, a malfunction might occur by input of an unintended edge.
3. Only the allowed values (functions) should be specified in the Pmn pin function control registers. If a value that is not allowed for the register is specified, correct operation is not guaranteed.
4. Do not assign a single function to multiple pins through the PFS register settings.
5. Points to note regarding the Port Mode Register (PMR), Port Direction Register (PDR), Pmn Pin Function Control Register (PmnPFS) settings for pins that have multiplexed pin functions are listed in Table 51.34.
   Reading the level of pins of port 5 is only available when the analog input alternative function is not selected.
   Modify the PSEL[2:0] bits when the bit corresponding to the given pin in the PMR register is set to 0.

Table 51.34 Register Settings

	PORTM.PMR		PmnPFS		
Item	PMRn	PDRn[1:0]	ISEL	PSEL[2:0]	Point to Note
After a reset	0 *1	00	0	000*1	These registers are not in use (Hi-Z input protection) after release from the reset state.
When these registers are not in use	0	00	0	X	
General input/ output ports	0	10/11 *2	0/1 *3	X	If these are multiplexed with pin interrupt inputs, set the PmnPFS.ISEL bit to 1 after setting PDRn[1:0] to 10 (input).
Peripheral functions	1	00/10 *4	0/1 *3	Peripheral function (see Table 51.5 to Table 51.33)	If these are multiplexed with pin interrupt inputs, set the PmnPFS.ISEL bit to 1 for pins for which the input function or input/output function was selected. The PmnPFS.ISEL bit must be set to 0 for pins for which the output function was selected because they cannot be multiplexed with pin interrupt inputs. When using this function with the port read function (reading the level of port pins from PIDRn bit), set PDRn[1:0] bits to 10.
Interrupt inputs	0	10	1	X	See section 51.4.4, Note on Pin Interrupts.
Analog inputs	1	00	X *5	001	To switch the output buffer and input buffer off, set the port pins for not in use.

x: Setting not required.

Note 1. The values after a reset in PORTM.PMR.PMR0, PORTM.PMR.PMR1, PM0PFS.PSEL[2:0], and PM1PFS.PSEL[2:0] are different with those specified as the initial values.
For details on the PSEL[2:0] bits, refer to “PMn Pin Function Control Register (PMnPFS) (n = 0 to 1)”.
For details on the PORTM.PMR register, refer to section 51.3.4, Port Mode Register (PMR).
Note 2. Setting the PDRn[1:0] bits to 10 sets the pin function for the general-purpose input port pin.
Setting the PDRn[1:0] bits to 11 sets the pin function for the general-purpose output port pin.
Note 3. Setting the PmnPFS.ISEL bit to 0 makes the pin incapable of functioning as an IRQ pin.
Setting the PmnPFS.ISEL bit to 1 makes the pin capable of functioning as an IRQ pin.
Note 4. When the PDRn[1:0] bits are set to 00, the port read function (reading the level of port pins from PIDRn bit) cannot be used.
When the PDRn[1:0] bits are set to 10, the port read function can be used.
Note 5. Even if the PmnPFS.ISEL bit is set to 1, the pin with this function cannot be used for pin interrupts.

51.4.3 Usage Note on Port Read Function

When a peripheral module for a pin which is bidirectional or an output is in use and the value of the PDR bits for the given pin is changed from 00 (initial value) to 10 (input enabled), the state of the pin can be read (port read function) from the PIDR register while the peripheral function is in use.
When a peripheral module for a pin which is an input or any of the input/output pin functions listed in Table 51.35 is in use, input is always enabled and the pin function can be used in parallel with port reading without changing the setting of the PDR register.
A shoot-through current flows when an external pin enters the Hi-Z state. An external pin must be pulled down or up when in the Hi-Z state.
When the pin functions in Table 51.36 are in use with an external bus controller, setting the PDR register to 10 (input enabled) is prohibited because the external bus controller controls enabling and disabling of input to the pins.

Table 51.35 List of Pins which Always Function as Input Pins

Module / Function	Channel	Port Function
Multi Function Timer Pulse Unit 3	MTU0	MTIOC0A (InOut)
		MTIOC0B (InOut)
		MTIOC0C (InOut)
		MTIOC0D (InOut)
	MTU1	MTIOC1A (InOut)
		MTIOC1B (InOut)
	MTU2	MTIOC2A (InOut)
		MTIOC2B (InOut)
	MTU3	MTIOC3A (InOut)
		MTIOC3B (InOut)
		MTIOC3C (InOut)
		MTIOC3D (InOut)
	MTU4	MTIOC4A (InOut)
		MTIOC4B (InOut)
		MTIOC4C (InOut)
		MTIOC4D (InOut)
	MTU6	MTIOC6A (InOut)
		MTIOC6B (InOut)
		MTIOC6C (InOut)
		MTIOC6D (InOut)
	MTU7	MTIOC7A (InOut)
		MTIOC7B (InOut)
		MTIOC7C (InOut)
		MTIOC7D (InOut)
	MTU8	MTIOC8A (InOut)
		MTIOC8B (InOut)
		MTIOC8C (InOut)
		MTIOC8D (InOut)

Table 51.35 List of Pins which Always Function as Input Pins

Module / Function	Channel	Port Function
General PWM Timer	GPT0	GTIOC0A (InOut)
		GTIOC0B (InOut)
	GPT1	GTIOC1A (InOut)
		GTIOC1B (InOut)
	GPT2	GTIOC2A (InOut)
		GTIOC2B (InOut)
	GPT3	GTIOC3A (InOut)
		GTIOC3B (InOut)
	GPT4	GTIOC4A (InOut)
		GTIOC4B (InOut)
	GPT5	GTIOC5A (InOut)
		GTIOC5B (InOut)
	GPT6	GTIOC6A (InOut)
		GTIOC6B (InOut)
	GPT7	GTIOC7A (InOut)
		GTIOC7B (InOut)
Serial Communication Interface with FIFO (SCIFA)	SCIF0	SCK0 (InOut)
		CTS0# (InOut)
		RTS0# (InOut)
	SCIF1	SCK1 (InOut)
		CTS1# (InOut)
		RTS1# (InOut)
	SCIF2	SCK2 (InOut)
		CTS2# (InOut)
		RTS2# (InOut)
	SCIF3	SCK3 (InOut)
	SCIF4	SCK4 (InOut)
Serial Communication Interface	SCI0	SCI_TXD0 (InOut)
		SCI_RXD0 (InOut)
		SCI_CTS0#/RTS0# (InOut)
		SCI_SCK0 (InOut)
	SCI1	SCI_TXD1 (InOut)
		SCI_RXD0 (1InOut)
		SCI_CTS1#/RTS1# (InOut)
		SCI_SCK1 (InOut)

Table 51.35 List of Pins which Always Function as Input Pins

Module / Function	Channel	Port Function
Renesas Serial Peripheral Interface	RSPI0	RSPCK0 (InOut)
		MOSI0 (InOut)
		MISO0 (InOut)
		SSL00 (InOut)
	RSPI1	RSPCK1 (InOut)
		MOSI1 (InOut)
		MISO1 (InOut)
		SSL10 (InOut)
	RSPI2	RSPCK2 (InOut)
		MOSI2 (InOut)
		MISO2 (InOut)
		SSL20 (InOut)
Ethernet Controller 0	Ether0	ET0_MDIO (InOut)
Ethernet Controller 1	Ether1	ET1_MDIO (InOut)
NAND Flash Controller	—	NFDATA0 (InOut)
		NFDATA1 (InOut)
		NFDATA2 (InOut)
		NFDATA3 (InOut)
		NFDATA4 (InOut)
		NFDATA5 (InOut)
		NFDATA6 (InOut)
		NFDATA7 (InOut)

Table 51.36 PDR setting prohibited function list

Module / Function	Port Function
Bus State Controller	D0 (InOut)
	D1 (InOut)
	D2 (InOut)
	D3 (InOut)
	D4 (InOut)
	D5 (InOut)
	D6 (InOut)
	D7 (InOut)
	D8 (InOut)
	D9 (InOut)
	D10 (InOut)
	D11 (InOut)
	D12 (InOut)
	D13 (InOut)
	D14 (InOut)
	D15 (InOut)
	WAIT#

51.4.4 Note on Pin Interrupts

51.4.4.1 Control of Pin Interrupts

Signals input from the general-purpose I/O port pins can be used for pin interrupts. For details on pin functions, see section 7.4.4, Pin Interrupts. Interrupt IDs 480 to 511 are allocated to the pin interrupts. Therefore, the general-purpose I/O port pins are classified into 32 groups and an interrupt ID number from 480 to 511 is allocated to each group. Table 51.37 shows the general-purpose I/O port pins in each group and the correspondence between the groups and interrupt IDs.

If multiple port pins in the same group are to be used for pin interrupts, the following restriction on input applies; when a port pin for use as a pin interrupt is at the high level, input of an interrupt request from another pin is prohibited. Details on the conditions for input are given in Table 51.38.

For details on how to set priority levels for the interrupt IDs allocated to the individual groups and the interrupt configuration register (GICD_ICFGRn), see section 7, Interrupt Controller.

Table 51.37 Details of Pin Interrupt Groups

Interrupt ID	Group Name into which Port Pins are Classified		GPIO Port Pin						
480	Group 0	(TINT0)	PL_0	PL_1	PL_2	PL_3	PL_4	PM_0	—
481	Group 1	(TINT1)	PK_0	PK_1	PK_2	PK_3	PK_4	PK_5	—
482	Group 2	(TINT2)	PJ_4	PJ_5	PJ_6	PJ_7	—	—	—
483	Group 3	(TINT3)	PJ_0	PJ_1	PJ_2	PJ_3	—	—	—
484	Group 4	(TINT4)	PH_0	PH_1	PH_2	PH_3	PH_4	PH_5	PH_6
485	Group 5	(TINT5)	PG_4	PG_5	PG_6	PG_7	—	—	—
486	Group 6	(TINT6)	PG_0	PG_1	PG_2	PG_3	—	—	—
487	Group 7	(TINT7)	PF_4	PF_5	PF_6	PF_7	—	—	—
488	Group 8	(TINT8)	PF_0	PF_1	PF_2	PF_3	—	—	—
489	Group 9	(TINT9)	PE_0	PE_1	PE_2	PE_3	PE_4	PE_5	PE_6
490	Group 10	(TINT10)	PD_4	PD_5	PD_6	PD_7	—	—	—
491	Group 11	(TINT11)	PD_0	PD_1	PD_2	PD_3	—	—	—
492	Group 12	(TINT12)	PC_4	PC_5	PC_6	PC_7	—	—	—
493	Group 13	(TINT13)	PC_0	PC_1	PC_2	PC_3	—	—	—
494	Group 14	(TINT14)	PB_0	PB_1	PB_2	PB_3	PB_4	PB_5	—
495	Group 15	(TINT15)	PA_4	PA_5	PA_6	PA_7	—	—	—
496	Group 16	(TINT16)	PA_0	PA_1	PA_2	PA_3	—	—	—
497	Group 17	(TINT17)	P9_4	P9_5	P9_6	P9_7	—	—	—
498	Group 18	(TINT18)	P9_0	P9_1	P9_2	P9_3	—	—	—
499	Group 19	(TINT19)	P8_4	P8_5	P8_6	P8_7	—	—	—
500	Group 20	(TINT20)	P8_0	P8_1	P8_2	P8_3	—	—	—
501	Group 21	(TINT21)	P7_4	P7_5	P7_6	P7_7	—	—	—
502	Group 22	(TINT22)	P7_0	P7_1	P7_2	P7_3	—	—	—
503	Group 23	(TINT23)	P6_4	P6_5	P6_6	P6_7	—	—	—
504	Group 24	(TINT24)	P6_0	P6_1	P6_2	P6_3	—	—	—
505	Group 25	(TINT25)	P5_4	P5_5	P5_6	P5_7	—	—	—
506	Group 26	(TINT26)	P5_0	P5_1	P5_2	P5_3	—	—	—
507	Group 27	(TINT27)	P4_1	P4_2	P4_3	P4_4	P4_5	P4_6	P4_7
508	Group 28	(TINT28)	P3_1	P3_2	P3_3	P3_4	P3_5	P4_0	—
509	Group 29	(TINT29)	P2_0	P2_1	P2_2	P2_3	P3_0	—	—
510	Group 30	(TINT30)	P1_0	P1_1	P1_2	P1_3	P1_4	—	—

Interrupt ID	Group Name into which Port Pins are Classified		GPIO Port Pin						
511	Group 31	(TINT31)	P0_0	P0_1	P0_2	P0_3	P0_4	P0_5	P0_6

Table 51.38 shows the behavior of two port pins, PL_0 and PL_1 in group 0. The other port pins in group 0 behave in the same way. In addition, this also applies to the behavior of multiple port pins in each of groups 1 to 31.

Table 51.38 Conditions for Input of Interrupt Requests between Port Pins in the Same Group

GICD_ICFGRn Setting	Conditions for Input of Pin Interrupts					Whether or Not to Meet the Condition for Input of Pin Interrupt Requests
	PL_0	PL_1	Target Interrupt	State of Interrupt ID Signal 480	State of Interrupt Request	
Rising Edge Detection		Low	PL_0	Rising Edge	Interrupt requests are issued from PL_0.	○
	Low		PL_1	Rising Edge	Interrupt requests are issued from PL_1.	○
			PL_0	Rising Edge	Interrupt requests are issued from PL_0.*1	○
			PL_1	Rising Edge	Interrupt requests are issued from PL_1.*1	○
		High	PL_0	High Level	Interrupt requests are not issued from PL_0.	×
	High		PL_1	High Level	Interrupt requests are not issued from PL_1.	×
			PL_0, PL_1	Rising Edge	Interrupt requests are issued from both PL_0 and PL_1.*2	×
High Level Detection	High Level	Low	PL_0	High Level	Interrupt requests are issued from PL_0.	○
	Low	High Level	PL_1	High Level	Interrupt requests are issued from PL_1.	○
	High Level		PL_0	High Level	Interrupt requests are issued from PL_0.*1	○
		High Level	PL_1	High Level	Interrupt requests are issued from PL_1.*1	○
	High Level	High Level	PL_0, PL_1	High Level	Interrupt requests are issued from both PL_0 and PL_1.*2	×
	High Level		PL_0	High Level	Interrupt requests are issued from both PL_0 and PL_1.*2	×
		High Level	PL_1	High Level	Interrupt requests are issued from both PL_0 and PL_1.*2	×

Note 1. The restriction on the input from pins in the same group is met. For details, refer to section 51.4.4.2, Restriction on the Input from Pins in the Same Group.

Note 2. Interrupt requests from multiple pins in the same group cannot be input at the same time. You cannot identify the port pin from which an interrupt request has been input.

51.4.4.2 Restriction on the Input from Pins in the Same Group

When setting pin interrupts with the use of pins in the same group, set a further interrupt so that it's issued the wait time limit (tTINTGW) after an interrupt has been issued. For details on tTINTGW, refer to "Wait time limit tTINTGW for TINT inputs from pins in the same group" in (2) Timing for Input of Interrupt Signals in the description of the timing pins for control signals in the electrical characteristics section.
Figure 51.16 shows an outline of the timing under the condition for the cases where pins in the same group are used for pin interrupts.

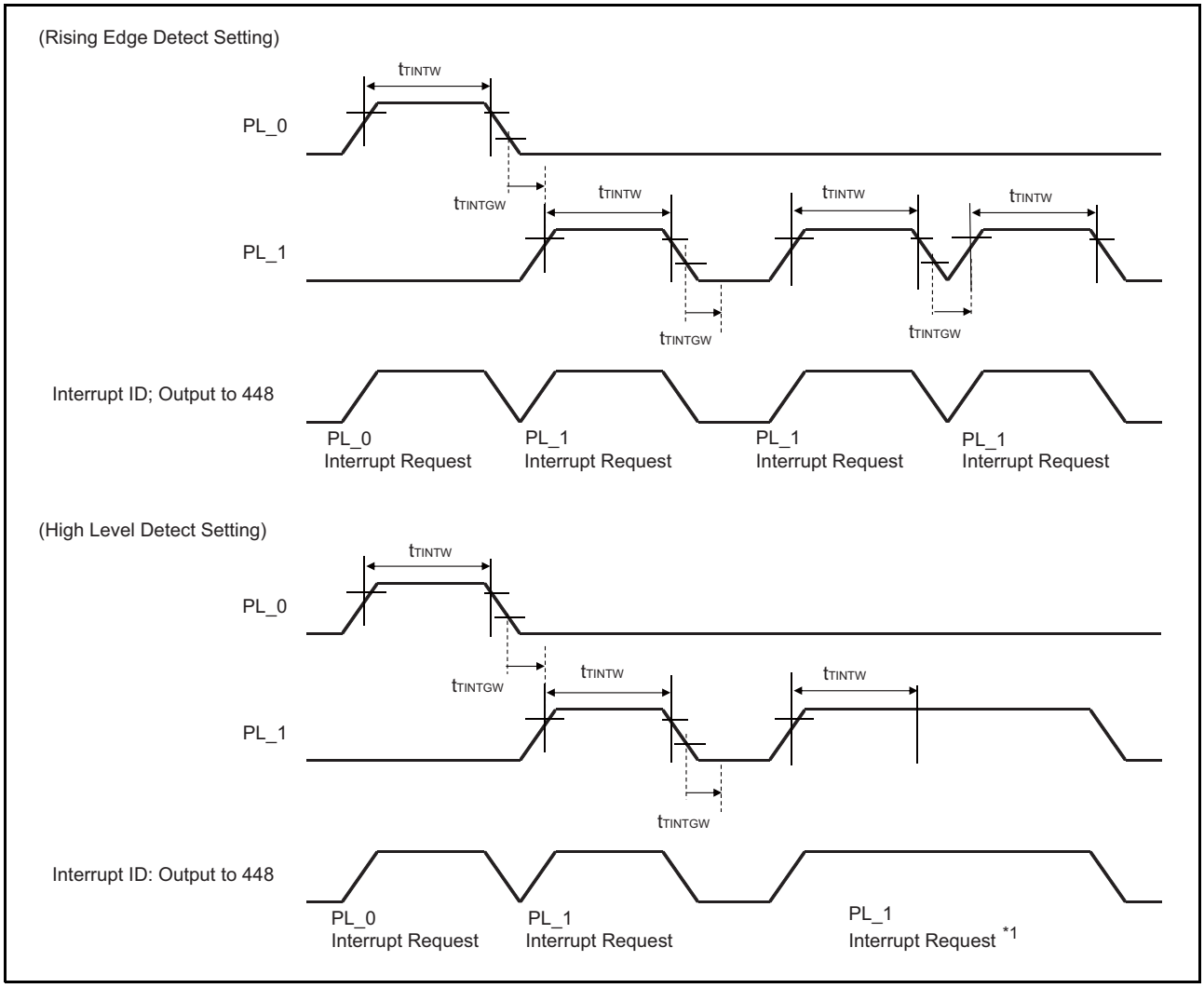


Figure 51.16 Restriction on the Timing of Input on Pins in the Same Group for Use as Pin Interrupts

tTINTGW represents the wait time after a pin interrupt signal falls.
The minimum interval in cases where pins in the same group are used for consecutive pin interrupts is "the time for the operating width of a port pin, TINTW" plus "the wait time for TINT inputs on pins in the same group, tTINTGW".

Note 1. The restriction of tTINTGW can only be ignored when high-level detection is selected for the interrupt controller and interrupts are set as to only issue from a single pin.

52. Power-Down Modes

This LSI supports sleep mode, software standby mode, deep standby mode, and module standby mode. In power-down modes, functions of CPU, clocks, on-chip memory, or part of on-chip peripheral modules are halted or the power-supply is turned off, through which low power consumption is achieved. These modes are canceled by a reset or interrupt.

52.1 Features

52.1.1 States of Processing and Power-Down Modes

(1) States of processing

This LSI has three general states of processing: the reset state, the program execution state, and the power-down modes. Figure 52.1 shows transitions between these states.

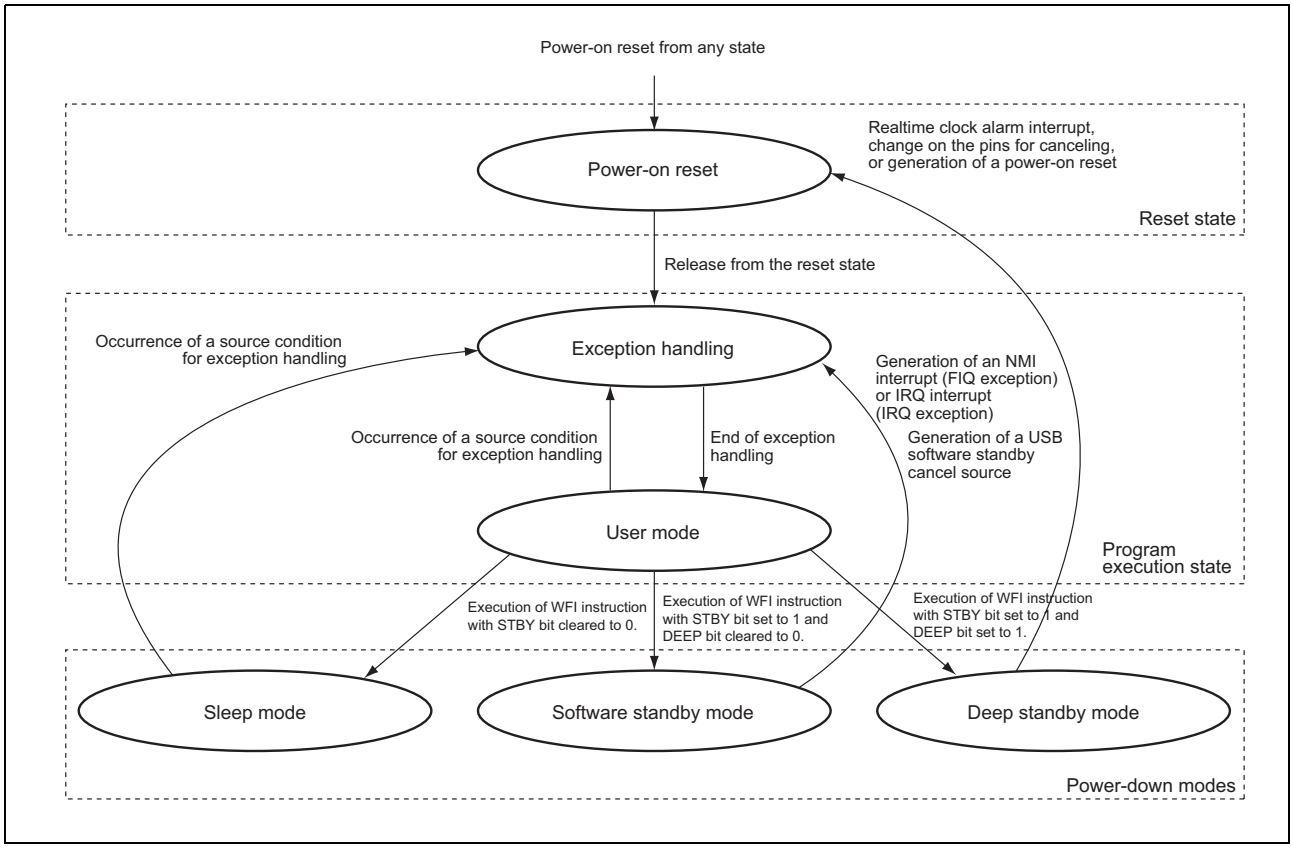


Figure 52.1 Transitions between Processing States

(2) Power-down modes

This LSI has the following power-down modes and function:

1. Sleep mode
2. Software standby mode
3. Deep standby mode
4. Module standby function

Table 52.1 shows the transition conditions for entering the modes from the program execution state, as well as the CPU and peripheral module states in each mode and the procedures for canceling each mode.

Table 52.1 States of Power-Down Modes

		State							
Power Down Mode	Transition Conditions	CPU, CPU Register, Primary Cache, TLB	Secondary Cache	Large-Capacity On-Chip RAM (including on-chip data-retention RAM)	On-Chip Peripheral Modules	Realtime Clock	Internal Power Supply	External Memory	Canceling Procedure
Sleep mode	Execute WFI instruction with STBY bit in STBCR1 cleared to 0	Halted Contents are held.*4	Running	Running	Running	Running *1	Applied	Auto refresh	• Interrupt • Power-on reset
Software standby mode	Execute WFI instruction with STBY and DEEP bits in STBCR1 set to 1 and 0, respectively	Halted Contents are held.*4	Halted Contents are held.*4	Halted Contents are held.*5	Halted	Running *1	Applied	Self refresh	• NMI or IRQ interrupt • Power-on reset • Change on the USB Software standby cancel source signal
Deep standby mode	Execute WFI instruction with STBY and DEEP bits in STBCR1 set to 1	Halted Contents are not held.	Halted Contents are not held.	Halted Contents in on-chip data-retention RAM are held*2, and the other contents in large-capacity on-chip RAM are not held.	Halted	Running *1	Shut off*6	Self refresh	• Power-on reset*3 • Realtime clock alarm interrupts 0 and 1*3 • Change on the pins for canceling*3 • USB 2.0 host/ function module channel interrupts 0 and 1
Module standby mode	Set the MSTP bits in STBCR2 to STBCR10 to 1	Running	Running	Running	Specified module halted	Halted	Applied	Auto refresh	• Clear MSTP bit to 0

Note 1. The realtime clock operates when the START bit in the RCR2 register is set to 1. For details, see section 16, Realtime Clock (RTC). When deep standby mode is canceled by a power-on reset, the running state cannot be retained. Make the initial setting for the realtime clock again.

Note 2. Setting the bits RRAMKP3 to RRAMKP0 in the RRAMKP register to 1 enables to retain the data in the corresponding area on the on-chip data-retention RAM during the transition to deep standby mode. When the deep standby mode is canceled by a power-on reset, the retained contents are initialized.

Note 3. Deep standby mode can be canceled by a power-on reset, a realtime clock alarm 0 or 1 interrupt, a USB cancel source 0 or 1, or change on the pins for canceling (PK_4, PK_2, PJ_5, PJ_1, PH_0, PG_6, PG_2, PH_1, PE_1, P6_2, P3_3, P3_1, and NMI). Even when deep standby mode is canceled by a source other than a power-on reset, the reset exception handling is executed instead of the interrupt exception handling.

Note 4. When sleep mode or software standby mode is canceled by a power-on reset, the retained contents are initialized.

Note 5. By setting the VRAME bit in the SYSCR1 register or VRAMWE bit in the SYSCR2 register to enable accesses, the retained contents are initialized when software standby mode is canceled by a power-on reset. By setting the VRAME bit in the SYSCR1

register or VRAMWE bit in the SYSCR2 register to disable accesses, the contents can be retained when software standby mode is canceled by a power-on reset. Note that this area can not be retained because the area at addresses from H'8002_0000 to H'8002_3FFF is used as working memory for the boot program in boot modes 0 to 7. In boot modes 1 and 2, this area can not be retained because a program is transferred from the external flash memory to the area at addresses from H'8002_4000 to H'8002_4000 + the size of the program - 1. For details, see section 3, Boot Mode.

Note 6. In deep standby mode, the fake debug mode, in which shutting off the supply of power to all modules is disabled, is usable for debugging.
For details, see section 53.4.3, Clock Power Control Register (ICEREGCLKPWRCTRL).

52.2 Register Descriptions

Table 52.2 shows the register configuration.

Table 52.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Standby control register 1	STBCR1	R/W	H'00	H'FCFE0020	8
Standby control register 2	STBCR2	R/W	H'6A	H'FCFE0024	8
Standby control register 3	STBCR3	R/W	H'FD	H'FCFE0420	8
Standby control register 4	STBCR4	R/W	H'FF	H'FCFE0424	8
Standby control register 5	STBCR5	R/W	H'F3	H'FCFE0428	8
Standby control register 6	STBCR6	R/W	H'FF	H'FCFE042C	8
Standby control register 7	STBCR7	R/W	H'FF	H'FCFE0430	8
Standby control register 8	STBCR8	R/W	H'F7	H'FCFE0434	8
Standby control register 9	STBCR9	R/W	H'FF	H'FCFE0438	8
Standby control register 10	STBCR10	R/W	H'9F	H'FCFE043C	8
Software reset control register 1	SWRSTCR1	R/W	H'00	H'FCFE0460	8
Software reset control register 2	SWRSTCR2	R/W	H'00	H'FCFE0464	8
System control register 1	SYSCR1	R/W	H'FF	H'FCFE0400	8
System control register 2	SYSCR2	R/W	H'FF	H'FCFE0404	8
System control register 3	SYSCR3	R/W	H'00	H'FCFE0408	8
CPU status register	CPUSTS	R	H'00	H'FCFE0018	8
Standby request register 1	STBREQ1	R/W	H'00	H'FCFE0030	8
Standby request register 2	STBREQ2	R/W	H'00	H'FCFE0034	8
Standby request register 3	STBREQ3	R/W	H'00	H'FCFE0038	8
Standby acknowledge register 1	STBACK1	R	H'00	H'FCFE0040	8
Standby acknowledge register 2	STBACK2	R	H'00	H'FCFE0044	8
Standby acknowledge register 3	STBACK3	R	H'00	H'FCFE0048	8
USB software standby cancel source control register 0	SSTBCCR0	R/W	H'0000	H’FCFEF810	16
USB software standby cancel source control register 1	SSTBCCR1	R/W	H'0000	H’FCFEF812	16
USB Software standby cancel source request register 0	SSTBCRR0	R/W	H'0000	H’FCFEF814	16
USB Software standby cancel source request register 1	SSTBCRR1	R/W	H'0000	H’FCFEF816	16
On-chip data-retention RAM area setting register	RRAMKP	R/W	H'00	H'FCFFC000	8
Deep standby control register	DSCTR	R/W	H'00	H'FCFFC002	8
Deep standby cancel source select register	DSSSR	R/W	H'0000	H'FCFFC004	16
USB Deep standby cancel source select register	USBDSSSR	R/W	H'00	H'FCFFC020	8
Deep standby cancel edge select register	DSESR	R/W	H'0000	H'FCFFC006	16
Deep standby cancel source flag register	DSFR	R/W	H'0000	H'FCFFC008	16
USB Deep standby cancel source flag register	USBDSFR	R/W	H'00	H'FCFFC024	8
Deep standby cancel Oscillation stability count register	DSCNT	R/W	H'0000	H'FCFFC00E	16
XTAL crystal oscillator gain control register	XTALCTR	R/W	H'00	H'FCFF1810	8
RTCXTAL select register	RTCXTALSEL	R/W	H'000x	H'FCFFC040	16

52.2.1 Standby Control Register 1 (STBCR1)

STBCR1 is an 8-bit readable/writable register that specifies the state of the power-down mode.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	STBY	DEEP	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	STBY	0	R/W	Software Standby, Deep Standby
6	DEEP	0	R/W	Specifies transition to software standby mode or deep standby mode. 0x: Executing WFI instruction puts chip into sleep mode. 10: Executing WFI instruction puts chip into software standby mode. 11: Executing WFI instruction puts chip into deep standby mode.
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

[Legend]

x: Don't care

52.2.2 Standby Control Register 2 (STBCR2)

STBCR2 is an 8-bit readable/writable register that controls the operation of each module.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	HIZ	—	—	—	—	—	—	MSTP 20
Initial value:	0	1	1	0	1	0	1	0
R/W:	R/W	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	HIZ	0	R/W	Port High Impedance Selects whether the state of specific output pin is retained or high impedance in software standby mode or deep standby mode. As to which pins are controlled, see section 57.1, Pin States. 0: The pin state is retained in software standby mode or deep standby mode. 1: The pin is set to high-impedance in software standby mode or deep standby mode.
6, 5	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
0	MSTP20	0	R/W	Module Stop 20*1 When the MSTP20 bit is set to 1, the clock supply to CoreSight is halted. 0: CoreSight runs. 1: Clock supply to CoreSight is halted.

Note 1. The transition to and release from module stop mode proceed in the steps described in section 52.3.5, Module Standby Function.

52.2.3 Standby Control Register 3 (STBCR3)

STBCR3 is an 8-bit readable/writable register that controls the operation of each module.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	—	MSTP 36	MSTP 35	MSTP 34	MSTP 33	MSTP 32	—	MSTP 30
Initial value:	1	1	1	1	1	1	0	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
6	MSTP36	1	R/W	Module Stop 36 When the MSTP36 bit is set to 1, the clock supply to the OS timer channel 0 is halted. 0: OS timer channel 0 runs. 1: Clock supply to OS timer channel 0 is halted.
5	MSTP35	1	R/W	Module Stop 35 When the MSTP35 bit is set to 1, the clock supply to the OS timer channel 1 is halted. 0: OS timer channel 1 runs. 1: Clock supply to OS timer channel 1 is halted.
4	MSTP34	1	R/W	Module Stop 34 When the MSTP34 bit is set to 1, the clock supply to the OS timer channel 2 is halted. 0: The OS timer channel 2 runs. 1: Clock supply to the OS timer channel 2 is halted.
3	MSTP33	1	R/W	Module Stop 33 When the MSTP33 bit is set to 1, the clock supply to the multi-function timer pulse unit 3 (MTU3a) is halted. 0: The multi-function timer pulse unit 3 runs. 1: Clock supply to the multi-function timer pulse unit 3 is halted.
2	MSTP32	1	R/W	Module Stop 32 When the MSTP32 bit is set to 1, the clock supply to the CANFD interface is halted. 0: The CANFD interface runs. 1: Clock supply to the CANFD interface is halted.
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	MSTP30	1	R/W	Module Stop 30 When the MSTP30 bit is set to 1, the clock supply to the General PWM Timer (GPT) is halted. 0: The General PWM Timer runs. 1: Clock supply to the General PWM Timer is halted.

52.2.4 Standby Control Register 4 (STBCR4)

STBCR4 is an 8-bit readable/writable register that controls the operation of each module.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP47	MSTP46	MSTP45	MSTP44	MSTP43	MSTP42	MSTP41	MSTP40
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP47	1	R/W	Module Stop 47 When the MSTP47 bit is set to 1, the clock supply to channel 0 of the serial communication interface with FIFO is halted. 0: Channel 0 of the serial communication interface with FIFO runs. 1: Clock supply to channel 0 of the serial communication interface with FIFO is halted.
6	MSTP46	1	R/W	Module Stop 46 When the MSTP46 bit is set to 1, the clock supply to channel 1 of the serial communication interface with FIFO is halted. 0: Channel 1 of the serial communication interface with FIFO runs. 1: Clock supply to channel 1 of the serial communication interface with FIFO is halted.
5	MSTP45	1	R/W	Module Stop 45 When the MSTP45 bit is set to 1, the clock supply to channel 2 of the serial communication interface with FIFO is halted. 0: Channel 2 of the serial communication interface with FIFO runs. 1: Clock supply to channel 2 of the serial communication interface with FIFO is halted.
4	MSTP44	1	R/W	Module Stop 44 When the MSTP44 bit is set to 1, the clock supply to channel 3 of the serial communication interface with FIFO is halted. 0: Channel 3 of the serial communication interface with FIFO runs. 1: Clock supply to channel 3 of the serial communication interface with FIFO is halted.
3	MSTP43	1	R/W	Module Stop 43 When the MSTP43 bit is set to 1, the clock supply to channel 4 of the serial communication interface with FIFO is halted. 0: Channel 4 of the serial communication interface with FIFO runs. 1: Clock supply to channel 4 of the serial communication interface with FIFO is halted.
2	MSTP42	1	R/W	Module Stop 42 When the MSTP42 bit is set to 1, the clock supply to channel 0 of the serial communication interface (SCI) is halted. 0: Channel 0 of the serial communication interface (SCI) runs. 1: Clock supply to channel 0 of the serial communication interface (SCI) is halted.
1	MSTP41	1	R/W	Module Stop 41 When the MSTP41 bit is set to 1, the clock supply to channel 1 of the serial communication interface (SCI) is halted. 0: Channel 1 of the serial communication interface (SCI) runs. 1: Clock supply to channel 1 of the serial communication interface (SCI) is halted.
0	MSTP40	1	R/W	Module Stop 40 When the MSTP40 bit is set to 1, the clock supply to the serial communication interface (IrDA) is halted. 0: The serial communication interface (IrDA) runs. 1: Clock supply to the serial communication interface (IrDA) is halted.

52.2.5 Standby Control Register 5 (STBCR5)

STBCR5 is an 8-bit readable/writable register that controls the operation of each module.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP57	MSTP56	—	—	MSTP53	MSTP52	MSTP51	—
Initial value:	1	1	1	1	0	0	1	1
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP57	1	R/W	Module Stop 57 When the MSTP57 bit is set to 1, the clock supply to the A/D converter is halted. 0: The A/D converter runs. 1: Clock supply to the A/D converter is halted.
6	MSTP56	1	R/W	Module Stop 56 When the MSTP56 bit is set to 1, the clock supply to the capture engine unit (CEU) is halted. 0: The capture engine unit (CEU) runs. 1: Clock supply to the capture engine unit (CEU) is halted.
5	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
4	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
3	MSTP53	0	R/W	Module Stop 53 When the MSTP53 bit is set to 1, the clock supply to the realtime clock channel 0 is halted. 0: The realtime clock channel 0 runs. 1: Clock supply to the realtime clock channel 0 is halted. Note: When the realtime clock channel 0 is halted, set the bits in registers shown below. • Set the RCKSEL bit in the control register 4 (RCR4) of the realtime clock channel 0 to 0. • Set the RTCEN bit in the control register 3 (RCR3) of the realtime clock channel 0 to 0. After the settings above, set the MSTP53 bit to 1.
2	MSTP52	0	R/W	Module Stop 52 When the MSTP52 bit is set to 1, the clock supply to the realtime clock channel 1 is halted. 0: The realtime clock channel 1 runs. 1: Clock supply to the realtime clock channel 1 is halted. Note: When the realtime clock channel 1 is halted, set the bits in registers shown below. • Set the RCKSEL bit in the control register 4 (RCR4) of the realtime clock channel 1 to 0. • Set the RTCEN bit in the control register 3 (RCR3) of the realtime clock channel 1 to 0. After the settings above, set the MSTP52 bit to 1.
1	MSTP51	1	R/W	Module Stop 51* When the MSTP51 bit is set to 1, the clock supply to the JPEG codec unit is halted. 0: The JPEG codec unit runs. 1: Clock supply to the JPEG codec unit is halted.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

Note: * The transition to and release from module stop mode proceed in the steps described in section 52.3.5, Module Standby Function.

52.2.6 Standby Control Register 6 (STBCR6)

STBCR6 is an 8-bit readable/writable register that controls the operation of each module.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	—	MSTP66	MSTP65	MSTP64	MSTP63	MSTP62	MSTP61	MSTP60
Initial value:	1	1	1	1	1	1	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
6	MSTP66	1	R/W	Module Stop 66* When the MSTP66 bit is set to 1, the clock supply to the video input module is halted. 0: The video input module runs. 1: Clock supply to the video input module is halted.
5	MSTP65	1	R/W	Module Stop 65* When the MSTP65 bit is set to 1, the clock supply to the channel 0 Ethernet controller and the channel 0 DMA controller for the Ethernet controller is halted. 0: The channel 0 Ethernet controller and the channel 0 DMA controller for the Ethernet controller run. 1: Clock supply to the channel 0 Ethernet controller and the channel 0 DMA controller for the Ethernet controller is halted.
4	MSTP64	1	R/W	Module Stop 64* When the MSTP64 bit is set to 1, the clock supply to the channel 1 Ethernet controller and the channel 1 DMA controller for the Ethernet controller is halted. 0: The channel 1 Ethernet controller and the channel 1 DMA controller for the Ethernet controller run. 1: Clock supply to the channel 1 Ethernet controller and the channel 1 DMA controller for the Ethernet controller is halted.
3	MSTP63	1	R/W	Module Stop 63* When the MSTP63 bit is set to 1, the clock supply to the PTP controller for the Ethernet controllers and DMA controller for the PTP controller is halted. 0: The PTP controller for the Ethernet controllers and DMA controller for the PTP controller run. 1: Clock supply to the PTP controller for the Ethernet controllers and DMA controller for the PTP controller is halted.
2	MSTP62	1	R/W	Module Stop 62* When the MSTP62 bit is set to 1, the clock supply to the circuits shared by the Ethernet controllers, PTP controller for the Ethernet controllers, and DMA controllers for the Ethernet controllers is halted. 0: The circuits shared by the Ethernet controllers, PTP controller for the Ethernet controllers, and DMA controllers for the Ethernet controllers run. 1: Clock supply to the circuits shared by the Ethernet controllers, PTP controller for the Ethernet controllers, and DMA controllers for the Ethernet controllers is halted.
1	MSTP61	1	R/W	Module Stop 61* When the MSTP61 bit is set to 1, the clock supply to the channel 0 USB 2.0 host/function module is halted. 0: The channel 0 USB 2.0 host/function module runs. 1: Clock supply to the channel 0 USB 2.0 host/function module is halted.
0	MSTP60	1	R/W	Module Stop 60* When the MSTP60 bit is set to 1, the clock supply to the channel 1 USB 2.0 host/function module is halted. 0: The channel 1 USB 2.0 host/function module runs. 1: Clock supply to the channel 1 USB 2.0 host/function module is halted.

Note: The transition to and release from module stop mode proceed in the steps described in section 52.3.5, Module Standby Function.

52.2.7 Standby Control Register 7 (STBCR7)

STBCR7 is an 8-bit readable/writable register that controls the operation of each module.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP77	MSTP76	MSTP75	—	MSTP73	MSTP72	MSTP71	MSTP70
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP77	1	R/W	Module Stop 77* When the MSTP77 bit is set to 1, the clock supply to the image renderer is halted. 0: The image renderer runs. 1: Clock supply to the image renderer is halted.
6	MSTP76	1	R/W	Module Stop 76* When the MSTP76 bit is set to 1, the clock supply to the 2D Drawing Engine is halted. 0: The 2D Drawing Engine runs. 1: Clock supply to the 2D Drawing Engine is halted.
5	MSTP75	1	R/W	Module Stop 75 When the MSTP75 bit is set to 1, the clock supply to the MIPI CSI-2 interface is halted. 0: The MIPI CSI-2 interface runs. 1: Clock supply to the MIPI CSI-2 interface is halted.
4	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
3	MSTP73	1	R/W	Module Stop 73 When the MSTP73 bit is set to 1, the clock supply to channel 0 of the serial sound interface is halted. 0: Channel 0 of the serial sound interface runs. 1: Clock supply to channel 0 of the serial sound interface is halted.
2	MSTP72	1	R	Module Stop 72 When the MSTP72 bit is set to 1, the clock supply to channel 1 of the serial sound interface is halted. 0: Channel 1 of the serial sound interface runs. 1: Clock supply to channel 1 of the serial sound interface is halted.
1	MSTP71	1	R/W	Module Stop 71 When the MSTP71 bit is set to 1, the clock supply to channel 2 of the serial sound interface is halted. 0: Channel 2 of the serial sound interface runs. 1: Clock supply to channel 2 of the serial sound interface is halted.
0	MSTP70	1	R/W	Module Stop 70 When the MSTP70 bit is set to 1, the clock supply to channel 3 of the serial sound interface is halted. 0: Channel 3 of the serial sound interface runs. 1: Clock supply to channel 3 of the serial sound interface is halted.

Note: The transition to and release from module stop mode proceed in the steps described in section 52.3.5, Module Standby Function.

52.2.8 Standby Control Register 8 (STBCR8)

STBCR8 is an 8-bit readable/writable register that controls the operation of each module.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP87	MSTP86	MSTP85	MSTP84	MSTP83	—	MSTP81	—
Initial value:	1	1	1	1	0	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP87	1	R/W	Module Stop 87 When the MSTP87 bit is set to 1, the clock supply to channel 0 of the I2C bus interface (RIIC) is halted. 0: Channel 0 of the I2C bus interface runs. 1: Clock supply to channel 0 of the I2C bus interface is halted.
6	MSTP86	1	R/W	Module Stop 86 When the MSTP86 bit is set to 1, the clock supply to channel 1 of the I2C bus interface (RIIC) is halted. 0: Channel 1 of the I2C bus interface runs. 1: Clock supply to channel 1 of the I2C bus interface is halted.
5	MSTP85	1	R/W	Module Stop 85 When the MSTP85 bit is set to 1, the clock supply to channel 2 of the I2C bus interface (RIIC) is halted. 0: Channel 2 of the I2C bus interface runs. 1: Clock supply to channel 2 of the I2C bus interface is halted.
4	MSTP84	1	R/W	Module Stop 84 When the MSTP84 bit is set to 1, the clock supply to channel 3 of the I2C bus interface (RIIC) is halted. 0: Channel 3 of the I2C bus interface runs. 1: Clock supply to channel 3 of the I2C bus interface is halted.
3	MSTP83	0	R/W	Module Stop 83 When the MSTP83 bit is set to 1, the clock supply to the SPI multi I/O bus controller is halted. 0: The SPI multi I/O bus controller runs. 1: Clock supply to the SPI multi I/O bus controller is halted.
2	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
1	MSTP81	1	R/W	Module Stop 81* When the MSTP81 bit is set to 1, the clock supply to the video display controller 6 is halted. 0: The video display controller 6 runs. 1: Clock supply to the video display controller 6 is halted.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

Note: The transition to and release from module stop mode proceed in the steps described in section 52.3.5, Module Standby Function.

52.2.9 Standby Control Register 9 (STBCR9)

STBCR9 is an 8-bit readable/writable register that controls the operation of each module.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP97	MSTP96	MSTP95	—	MSTP93	MSTP92	MSTP91	MSTP90
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP97	1	R/W	Module Stop 97 When the MSTP97 bit is set to 1, the clock supply to channel 0 of the Renesas serial peripheral interface is halted. 0: Channel 0 of the Renesas serial peripheral interface runs. 1: Clock supply to channel 0 of the Renesas serial peripheral interface is halted.
6	MSTP96	1	R/W	Module Stop 96 When the MSTP96 bit is set to 1, the clock supply to channel 1 of the Renesas serial peripheral interface is halted. 0: Channel 1 of the Renesas serial peripheral interface runs. 1: Clock supply to channel 1 of the Renesas serial peripheral interface is halted.
5	MSTP95	1	R/W	Module Stop 95 When the MSTP95 bit is set to 1, the clock supply to channel 2 of the Renesas serial peripheral interface is halted. 0: Channel 2 of the Renesas serial peripheral interface runs. 1: Clock supply to channel 2 of the Renesas serial peripheral interface is halted.
4	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
3	MSTP93	1	R/W	Module Stop 93 When the MSTP93 bit is set to 1, the clock supply to the HyperBus controller is halted. 0: The HyperBus controller runs. 1: Clock supply to the HyperBus controller is halted.
2	MSTP92	1	R/W	Module Stop 92 When the MSTP92 bit is set to 1, the clock supply to the octa memory controller is halted. 0: The octa memory controller runs. 1: Clock supply to the octa memory controller is halted.
1	MSTP91	1	R/W	Module Stop 91* When the MSTP91 bit is set to 1, the clock supply to the Renesas SPDIF interface is halted. 0: The Renesas SPDIF interface run. 1: Clock supply to the Renesas SPDIF interface is halted.
0	MSTP90	1	R/W	Module Stop 90* When the MSTP90 bit is set to 1, the clock supply to the dynamic reconfigurable processer (DRP) is halted. 0: The dynamic reconfigurable processer (DRP) runs. 1: Clock supply to the dynamic reconfigurable processer (DRP) is halted. If the product does not have a DRP, the value written to this bit should always be 1.

Note: The transition to and release from module stop mode proceed in the steps described in section 52.3.5, Module Standby Function.

52.2.10 Standby Control Register 10 (STBCR10)

STBCR10 is an 8-bit readable/writable register that controls the operation of each module.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP107	—	—	MSTP104	MSTP103	MSTP102	MSTP101	MSTP100
Initial value:	1	0	0	1	1	1	1	1
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP107	1	R/W	Module Stop 107*1 When the MSTP107 bit is set to 1, the clock supply to the Trusted Secure IP module is halted. 0: The Trusted Secure IP module runs. 1: Clock supply to the Trusted Secure IP module is halted. Note: If the product does not have a Trusted Secure IP module, the value written to this bit should always be 1.
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	MSTP104	1	R/W	Module Stop 104*1 When the MSTP104 bit is set to 1, the clock supply to the NAND flash controller is halted. 0: The NAND flash controller runs. 1: Clock supply to the NAND flash controller is halted.
3	MSTP103	1	R/W	Module Stop 103 and 102*1 When the MSTP103 and MSTP102 bits are set to 11, the clock supply to the channel 0 SD/MMC host interface is halted. 00: The channel 0 SD/MMC host interface runs. 11: Clock supply to the channel 0 SD/MMC host interface is halted. Others: Setting prohibited
2	MSTP102	1	R/W	
1	MSTP101	1	R/W	Module Stop 101 and 100*1 When the MSTP101 and MSTP100 bits are set to 11, the clock supply to the channel 1 SD/MMC host interface is halted. 00: The channel 1 SD/MMC host interface runs. 11: Clock supply to the channel 1 SD/MMC host interface is halted. Others: Setting prohibited
0	MSTP100	1	R/W	

Note 1. The transition to and release from module stop mode proceed in the steps described in section 52.3.5, Module Standby Function.

52.2.11 Software Reset Control Register 1 (SWRSTCR1)

SWRSTCR1 is an 8-bit readable/writable register that controls software resetting of individual modules and the operation of the crystal oscillator for audio.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	AXTALE	—	—	—	SRST13	SRST12	SRST11	SRST10
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	AXTALE	0	R/W	AUDIO_X1 Clock Control Controls the function of AUDIO_X1 pin. 0: Runs the on-chip crystal oscillator/enables the external clock input. 1: Halts the on-chip crystal oscillator/disables the external clock input.
6 to 4	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	SRST13	0	R/W	Video Display Controller 6 Software Reset Controls resetting of the video display controller 6 by software. 0: The video display controller 6 reset is canceled. 1: The video display controller 6 is reset.
2	SRST12	0	R/W	Trusted Secure IP Software Reset Controls resetting of the Trusted Secure IP module by software. 0: The Trusted Secure IP reset is canceled. 1: The Trusted Secure IP module is reset. Note: If the product does not have a Trusted Secure IP module, the value written to this bit should always be 0.
1	SRST11	0	R/W	JPEG Codec Unit Software Reset Controls resetting of the JPEG codec unit by software. 0: The JPEG codec unit reset is canceled. 1: The JPEG codec unit is reset.
0	SRST10	0	R/W	SPI Multi I/O Bus Controller Software Reset Controls resetting of the SPI multi I/O bus controller by software. 0: The SPI multi I/O bus controller reset is canceled. 1: The SPI multi I/O bus controller is reset.

52.2.12 Software Reset Control Register 2 (SWRSTCR2)

SWRSTCR2 is an 8-bit readable/writable register that controls a software reset for each module.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	—	SRST 26	SRST 25	SRST 24	SRST 23	SRST 22	SRST 21	—
Initial value:	1	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
6	SRST26	0	R/W	NAND Flash Controller Software Reset Controls resetting of the NAND flash controller by software. 0: The NAND flash controller reset is canceled. 1: The NAND flash controller is reset.
5	SRST25	0	R/W	Video Input Module Software Reset Controls resetting of the video input module by software. 0: The video input module reset is canceled. 1: The video input module is reset.
4	SRST24	0	R/W	Image Renderer Software Reset Controls resetting of the image renderer by software. 0: The image renderer reset is canceled. 1: The image renderer is reset.
3	SRST23	0	R/W	MIPI CSI-2 Interface Software Reset Controls resetting of the MIPI CSI-2 interface by software. 0: The MIPI CSI-2 interface reset is canceled. 1: The MIPI CSI-2 interface is reset.
2	SRST22	0	R/W	2D Drawing Engine Software Reset Controls resetting of the 2D drawing engine by software. 0: The 2D drawing engine reset is canceled. 1: The 2D drawing engine is reset.
1	SRST21	0	R/W	Capture Engine Unit Software Reset Controls resetting of the capture engine unit by software. 0: The capture engine unit reset is canceled. 1: The capture engine unit is reset.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

52.2.13 System Control Register 1 (SYSCR1)

SYSCR1 is an 8-bit readable/writable register that enables or disables access (read and write) to a specified page in the large-capacity on-chip RAM.
When a VRAMEn (n = 0 to 4) bit in SYSCR1 is set to 1, access to page n is enabled. When a VRAMEn bit is cleared to 0, page n cannot be accessed. In this case, an undefined value is returned when reading data or fetching an instruction from page n, and writing to page n is ignored. The initial value of a VRAMEn bit is 1.
SYSCR1 should be set with a program located in an area other than the large-capacity on-chip RAM. Furthermore, an instruction to read SYSCR1 should be located immediately after the instruction to write to SYSCR1. If not, normal access is not guaranteed.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	VRAME4	VRAME3	VRAME2	VRAME1	VRAME0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
4	VRAME4	1	R/W	RAM Enable 4 (corresponding area: page 4*1 in large-capacity on-chip RAM) 0: Access to page 4 is disabled. 1: Access to page 4 is enabled.
3	VRAME3	1	R/W	RAM Enable 3 (corresponding area: page 3*1 in large-capacity on-chip RAM) 0: Access to page 3 is disabled. 1: Access to page 3 is enabled.
2	VRAME2	1	R/W	RAM Enable 2 (corresponding area: page 2*1 in large-capacity on-chip RAM) 0: Access to page 2 is disabled. 1: Access to page 2 is enabled.
1	VRAME1	1	R/W	RAM Enable 1 (corresponding area: page 1*1 in large-capacity on-chip RAM) 0: Access to page 1 is disabled. 1: Access to page 1 is enabled.
0	VRAME0	1	R/W	RAM Enable 0 (corresponding area: page 0*1 in large-capacity on-chip RAM) 0: Access to page 0 is disabled. 1: Access to page 0 is enabled.

Note 1. For addresses in each page, see section 50, On-Chip RAM.

52.2.14 System Control Register 2 (SYSCR2)

SYSCR2 is an 8-bit readable/writable register that enables or disables writing to a specified page in the large-capacity on-chip RAM.
When a VRAMWEn (n = 0 to 4) bit in SYSCR2 is set to 1, writing to page n is enabled. When a VRAMWEn bit is cleared to 0, writing to page n is ignored. The initial value of a VRAMWEn bit is 1.
SYSCR2 should be set with a program located in an area other than the large-capacity on-chip RAM. Furthermore, an instruction to read SYSCR2 should be located immediately after the instruction to write to SYSCR2. If not, normal access is not guaranteed.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	VRAME4	VRAME3	VRAME2	VRAME1	VRAME0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
4	VRAMWE4	1	R/W	RAM Write Enable 4 (corresponding area: page 4*1 in large-capacity on-chip RAM) 0: Writing to page 4 is disabled. 1: Writing to page 4 is enabled.
3	VRAMWE3	1	R/W	RAM Write Enable 3 (corresponding area: page 3*1 in large-capacity on-chip RAM) 0: Writing to page 3 is disabled. 1: Writing to page 3 is enabled.
2	VRAMWE2	1	R/W	RAM Write Enable 2 (corresponding area: page 2*1 in large-capacity on-chip RAM) 0: Writing to page 2 is disabled. 1: Writing to page 2 is enabled.
1	VRAMWE1	1	R/W	RAM Write Enable 1 (corresponding area: page 1*1 in large-capacity on-chip RAM) 0: Writing to page 1 is disabled. 1: Writing to page 1 is enabled.
0	VRAMWE0	1	R/W	RAM Write Enable 0 (corresponding area: page 0*1 in large-capacity on-chip RAM) 0: Writing to page 0 is disabled. 1: Writing to page 0 is enabled.

Note 1. For addresses in each page, see section 50, On-Chip RAM.

52.2.15 System Control Register 3 (SYSCR3)

SYSCR3 is an 8-bit readable/writable register that enables or disables writing to a specified page in the on-chip data-retention RAM.

When a RRAMWEn (n = 0 to 3) bit in SYSCR3 is set to 1, writing to page n is enabled. When a RRAMWEn bit is cleared to 0, writing to page n is ignored. The initial value of a RRAMWEn bit is 0.

SYSCR3 should be set with a program located in an area other than the on-chip data-retention RAM.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	RRAMWE3	RRAMWE2	RRAMWE1	RRAMWE0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	RRAMWE3	0	R/W	RAM Write Enable 3 (corresponding area: page 3*2 in on-chip data-retention RAM) 0: Writing to page 3 is disabled. 1: Writing to page 3 is enabled.
2	RRAMWE2	0	R/W	RAM Write Enable 2 (corresponding area: page 2*2 in on-chip data-retention RAM) 0: Writing to page 2 is disabled. 1: Writing to page 2 is enabled.
1	RRAMWE1	0	R/W	RAM Write Enable 1 (corresponding area: page 1*2 in on-chip data-retention RAM) 0: Writing to page 1 is disabled. 1: Writing to page 1 is enabled.
0	RRAMWE0	0	R/W	RAM Write Enable 0 (corresponding area: page 0*2 in on-chip data-retention RAM) 0: Writing to page 0 is disabled. 1: Writing to page 0 is enabled.

Note 1. For addresses in each page, see section 50, On-Chip RAM.

Note 2. When the VRAME0 bit in SYSCR1 is cleared to 0 (access to page 0 in large-capacity on-chip RAM is invalid), the on-chip data-retention RAM cannot be accessed (read and written), regardless of the setting of this bit.
When the VRAMWE0 bit in SYSCR2 is cleared to 0 (writing to page 0 in large-capacity on-chip RAM is invalid), the on-chip data-retention RAM cannot be written, regardless of the setting of this bit.

52.2.16 CPU Status Register (CPUSTS)

CPUSTS is an 8-bit readable register that indicates the status of the CPU.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	IS BUSY	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0.
4	ISBUSY	0	R	State during Changing of the Frequency of CPU and Return from Software Standby Indicates that the frequency of CPU is being changed or that return from software standby is in progress. Do not execute a WFI instruction while this bit is 1. 0: The frequency of CPU is not being changed and returning from software standby is not in progress 1: The frequency of CPU is being changed or return from software standby is in progress
3 to 0	—	All 0	R	Reserved These bits are always read as 0.

52.2.17 Standby Request Register 1 (STBREQ1)

This register is used to request notification of whether a CPU or peripheral module is ready for standby. The CPU or peripheral module returns a standby acknowledgement on receipt of a request for notification if it is ready for standby.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	—	—	STBRQ 15	—	STBRQ 13	STBRQ 12	STBRQ 11	STBRQ 10
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	STBRQ15	0	R/W	Standby Request to CoreSight 0: The standby request to the CoreSight is invalid. 1: The standby request to the CoreSight is valid.*1
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3	STBRQ13	0	R/W	Standby Request to JPEG Codec Unit 0: The standby request to the JPEG codec unit is invalid. 1: The standby request to the JPEG codec unit is valid.*1
2	STBRQ12	0	R/W	Standby Request to the Channel 0 SD/MMC Host Interface 0: The standby request to the channel 0 SD/MMC host interface is invalid. 1: The standby request to the channel 0 SD/MMC host interface is valid.*1
1	STBRQ11	0	R/W	Standby Request to the Channel 1 SD/MMC Host Interface 0: The standby request to the channel 1 SD/MMC host interface is invalid. 1: The standby request to the channel 1 SD/MMC host interface is valid.*1
0	STBRQ10	0	R/W	Standby Request to Capture Engine Unit 0: The standby request to the capture engine unit is invalid. 1: The standby request to the capture engine unit is valid.*1

Note 1. When the MSTP bit for the corresponding module is 1, writing 1 to the STBRQ bit has no effect.

52.2.18 Standby Request Register 2 (STBREQ2)

This register is used to request notification of whether a peripheral module is ready for standby. The peripheral module returns a standby acknowledgement on receipt of a request for notification if it is ready for standby.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	STBRQ27	STBRQ26	STBRQ25	STBRQ24	STBRQ23	STBRQ22	STBRQ21	STBRQ20
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	STBRQ27	0	R/W	Standby Request to Video Input Module 0: The standby request to the video input module is invalid. 1: The standby request to the video input module is valid.*1
6	STBRQ26	0	R/W	Standby Request to Ethernet Controller 0: The standby request to the Ethernet controller is invalid. 1: The standby request to the Ethernet controller is valid.*1
5	STBRQ25	0	R/W	Standby Request to Channel 0 of Video Display Controller 6 0: The standby request to channel 0 of the video display controller 6 is invalid. 1: The standby request to channel 0 of the video display controller 6 is valid.*1
4	STBRQ24	0	R/W	Standby Request to Dynamic Reconfigurable Processor (DRP) 0: The standby request to the dynamic reconfigurable processor (DRP) is invalid. 1: The standby request to the dynamic reconfigurable processor (DRP) is valid.*1 If the product does not have a DRP, the value written to this bit should always be 0.
3	STBRQ23	0	R/W	Standby Request to Image Renderer 0: The standby request to the image renderer is invalid. 1: The standby request to the image renderer is valid.*1
2	STBRQ22	0	R/W	Standby Request to NAND Flash Controller 0: The standby request to the NAND flash controller is invalid. 1: The standby request to the NAND flash controller is valid.*1
1	STBRQ21	0	R/W	Standby Request to 2D Drawing Engine (Texture) 0: The standby request to the 2D drawing engine (texture) is invalid. 1: The standby request to the 2D drawing engine (texture) is valid.*1
0	STBRQ20	0	R/W	Standby Request to 2D Drawing Engine (Data) 0: The standby request to the 2D drawing engine (data) is invalid. 1: The standby request to the 2D drawing engine (data) is valid.*1

Note 1. When the MSTP bit for the corresponding module is 1, writing 1 to the STBRQ bit has no effect.

52.2.19 Standby Request Register 3 (STBREQ3)

This register is used to request notification of whether a peripheral module is ready for standby. The peripheral module returns a standby acknowledgement on receipt of a request for notification if it is ready for standby.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	STBRQ33	STBRQ32	STBRQ31	STBRQ30
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	STBRQ33	0	R/W	Standby Request to the Channel 1 USB 2.0 Function Module 0: The standby request to the channel 1 USB 2.0 function module is invalid. 1: The standby request to the channel 1 USB 2.0 function module is valid.*1
2	STBRQ32	0	R/W	Standby Request to the Channel 1 USB 2.0 Host Module 0: The standby request to the channel 1 USB 2.0 host module is invalid. 1: The standby request to the channel 1 USB 2.0 host module is valid.*1
1	STBRQ31	0	R/W	Standby Request to the Channel 0 USB 2.0 Function Module 0: The standby request to the channel 0 USB 2.0 function module is invalid. 1: The standby request to the channel 0 USB 2.0 function module is valid.*1
0	STBRQ30	0	R/W	Standby Request to the Channel 0 USB 2.0 Host Module 0: The standby request to the channel 0 USB 2.0 host module is invalid. 1: The standby request to the channel 0 USB 2.0 host module is valid.*1

Note 1. When the MSTP bit for the corresponding module is 1, writing 1 to the STBRQ bit has no effect.

52.2.20 Standby Acknowledge Register 1 (STBACK1)

This register is used to provide notification that a CPU or peripheral module is in the standby ready state. The CPU or peripheral module returns a standby acknowledgement on reception of a standby notification request if it is ready for standby. This register is a read-only register.

Bit:	7	6	5	4	3	2	1	0
	—	—	STBAK15	—	STBAK13	STBAK12	STBAK11	STBAK10
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0.
5	STBAK15	0	R	Standby Acknowledgement from CoreSight 0: The standby acknowledgement from CoreSight is invalid. 1: The standby acknowledgement from CoreSight is valid.*1
4	—	0	R	Reserved This bit is always read as 0.
3	STBAK13	0	R	Standby Acknowledgement from JPEG Codec Unit 0: The standby acknowledgement from the JPEG codec unit is invalid. 1: The standby acknowledgement from the JPEG codec unit is valid.*1
2	STBAK12	0	R	Standby Acknowledgement from the Channel 0 SD/MMC Host Interface 0: The standby acknowledgement from the channel 0 SD/MMC host interface is invalid. 1: The standby acknowledgement from the channel 0 SD/MMC host interface is valid.*1
1	STBAK11	0	R	Standby Acknowledgement from the Channel 1 SD/MMC Host Interface 0: The standby acknowledgement from the channel 1 SD/MMC host interface is invalid. 1: The standby acknowledgement from the channel 1 SD/MMC host interface is valid.*1
0	STBAK10	0	R	Standby Acknowledgement from Capture Engine Unit 0: The standby acknowledgement from the capture engine unit is invalid. 1: The standby acknowledgement from the capture engine unit is valid.*1

Note 1. The bits in STBACK1 are set to 1 when a standby acknowledgement is transmitted from the module while the MSTP bit for the corresponding module is set to 0.

52.2.21 Standby Acknowledge Register 2 (STBACK2)

This register is used to provide notification that a peripheral module is in the standby ready state. The peripheral module returns a standby acknowledgement on reception of a standby notification request if it is ready for standby. This register is a read-only register.

Bit:	7	6	5	4	3	2	1	0
	STBAK 27	STBAK 26	STBAK 25	STBAK 24	STBAK 23	STBAK 22	STBAK 21	STBAK 20
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	STBAK27	0	R	Standby Acknowledgement from Video Input Module 0: The standby acknowledgement from the video input module is invalid. 1: The standby acknowledgement from the video input module is valid.*1
6	STBAK26	0	R	Standby Acknowledgement from Ethernet Controller 0: The standby acknowledgement from the Ethernet controller is invalid. 1: The standby acknowledgement from the Ethernet controller is valid.*1
5	STBAK25	0	R	Standby Acknowledgement from Video Display Controller 6 0: The standby acknowledgement from the video display controller 6 is invalid. 1: The standby acknowledgement from the video display controller 6 is valid.*1
4	STBAK24	0	R	Standby Acknowledgement from Dynamic Reconfigurable Processor (DRP) 0: The standby acknowledgement from the dynamic reconfigurable processor (DRP) is invalid. 1: The standby acknowledgement from the dynamic reconfigurable processor (DRP) is valid.*1 If the product does not have a DRP, the value written to this bit should always be 0.
3	STBAK23	0	R	Standby Acknowledgement from Image Renderer 0: The standby acknowledgement from the image renderer is invalid. 1: The standby acknowledgement from the image renderer is valid.*1
2	STBAK22	0	R	Standby Acknowledgement from NAND Flash Controller 0: The standby acknowledgement from the NAND flash controller is invalid. 1: The standby acknowledgement from the NAND flash controller is valid.*1
1	STBAK21	0	R	Standby Acknowledgement from 2D Drawing Engine (Texture) 0: The standby acknowledgement from the 2D drawing engine (texture) is invalid. 1: The standby acknowledgement from the 2D drawing engine (texture) is valid.*1
0	STBAK20	0	R	Standby Acknowledgement from 2D Drawing Engine (Data) 0: The standby acknowledgement from the 2D drawing engine (data) is invalid. 1: The standby acknowledgement from the 2D drawing engine (data) is valid.*1

Note 1. The bits in STBACK2 are set to 1 when a standby acknowledgement is transmitted from the module while the MSTP bit for the corresponding module is set to 0.

52.2.22 Standby Acknowledge Register 3 (STBACK3)

This register is used to provide notification that a peripheral module is in the standby ready state. The peripheral module returns a standby acknowledgement on reception of a standby notification request if it is ready for standby. This register is a read-only register.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	STBAK 33	STBAK 32	STBAK 31	STBAK 30
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0.
3	STBAK33	0	R	Standby Acknowledgement from the Channel 1 USB 2.0 Function Module 0: The standby acknowledgement from the channel 1 USB 2.0 function module is invalid. 1: The standby acknowledgement from the channel 1 USB 2.0 function module is valid.*1
2	STBAK32	0	R	Standby Acknowledgement from the Channel 1 USB 2.0 Host Module 0: The standby acknowledgement from the channel 1 USB 2.0 host module is invalid. 1: The standby acknowledgement from the channel 1 USB 2.0 host module is valid.*1
1	STBAK31	0	R	Standby Acknowledgement from the Channel 0 USB 2.0 Function Module 0: The standby acknowledgement from the channel 0 USB 2.0 function module is invalid. 1: The standby acknowledgement from the channel 0 USB 2.0 function module is valid.*1
0	STBAK30	0	R	Standby Acknowledgement from the Channel 0 USB 2.0 Host Module 0: The standby acknowledgement from the channel 0 USB 2.0 host module is invalid. 1: The standby acknowledgement from the channel 0 USB 2.0 host module is valid.*1

Note 1. The bits in STBACK2 are set to 1 when a standby acknowledgement is transmitted from the module while the MSTP bit for the corresponding module is set to 0.

52.2.23 On-Chip Data-Retention RAM Area Setting Register (RRAMKP)

RRAMKP is an 8-bit readable/writable register that selects whether the contents of the corresponding area of the on-chip data-retention RAM are retained or not in deep standby mode.
When the RRAMKP3 to RRAMKP0 bits are set to 1, the contents of the corresponding area of the on-chip data-retention RAM are retained in deep standby mode. When these bits are cleared to 0, the contents of the corresponding area of the on-chip data-retention RAM are not retained in deep standby mode.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	RRAMKP3	RRAMKP2	RRAMKP1	RRAMKP0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	RRAMKP3	0	R/W	On-Chip Data-Retention RAM Storage Area 3 (corresponding area: page 3*1 in on-chip data-retention RAM) 0: The contents of the on-chip data-retention RAM are not retained in deep standby mode. 1: The contents of the on-chip data-retention RAM are retained in deep standby mode.
2	RRAMKP2	0	R/W	On-Chip Data-Retention RAM Storage Area 2 (corresponding area: page 2*1 in on-chip data-retention RAM) 0: The contents of the on-chip data-retention RAM are not retained in deep standby mode. 1: The contents of the on-chip data-retention RAM are retained in deep standby mode.
1	RRAMKP1	0	R/W	On-Chip Data-Retention RAM Storage Area 1 (corresponding area: page 1*1 in on-chip data-retention RAM) 0: The contents of the on-chip data-retention RAM are not retained in deep standby mode. 1: The contents of the on-chip data-retention RAM are retained in deep standby mode.
0	RRAMKP0	0	R/W	On-Chip Data-Retention RAM Storage Area 0 (corresponding area: page 0*1 in on-chip data-retention RAM) 0: The contents of the on-chip data-retention RAM are not retained in deep standby mode. 1: The contents of the on-chip data-retention RAM are retained in deep standby mode.

Note 1. For addresses in each page, see section 50, On-Chip RAM.

52.2.24 USB Software Standby Cancel Source Control Register 0 (SSTBCCR0)

SSTBCCR0 is a 16-bit readable/writable*1 register that corresponds to the USB channel 0. This register selects whether to mask the corresponding cancellation source in software standby mode. When bits 11 to 8, bit 1, and bit 0 of SSTBCCR0 are set to 0, corresponding cancellation sources are masked. When these bits are set to 1, the corresponding cancellation sources are not masked.
The VBUSIN, OVRCLR, CC1_RD, and CC2_RD bits of this register correspond to the VBUSIN0, OVRCLR0, CC1_Rd0, and CC2_Rd0 pins of the USB input/output port, respectively.

Note 1. When writing to this register, see section 52.4, Usage Notes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CC2_RD	CC1_RD	OVRCLR	VBUSIN	—	—	—	—	—	—	DM	DP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	CC2_RD	0	R/W	Mask selection of the cancellation source signal in software standby state 1: Software standby mode, does not mask the cancellation source. 0: Software standby mode, to mask the cancellation source.
10	CC1_RD	0	R/W	Mask selection of the cancellation source signal in software standby state 1: Software standby mode, does not mask the cancellation source. 0: Software standby mode, to mask the cancellation source.
9	OVRCLR	0	R/W	Mask selection of the cancellation source signal in software standby state 1: Software standby mode, does not mask the cancellation source. 0: Software standby mode, to mask the cancellation source.
8	VBUSIN	0	R/W	Mask selection of the cancellation source signal in software standby state 1: Software standby mode, does not mask the cancellation source. 0: Software standby mode, to mask the cancellation source.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	DM	0	R/W	Mask selection of the cancellation source signal in software standby state 1: Software standby mode, does not mask the cancellation source. 0: Software standby mode, to mask the cancellation source.
0	DP	0	R/W	Mask selection of the cancellation source signal in software standby state 1: Software standby mode, does not mask the cancellation source. 0: Software standby mode, to mask the cancellation source.

52.2.25 USB Software Standby Cancel Source Control Register 1 (SSTBCCR1)

SSTBCCR1 is a 16-bit readable/writable*1 register that corresponds to the USB channel 1. This register selects whether to mask the corresponding cancellation source in software standby mode. When bits 11 to 8, bit 1, and bit 0 of SSTBCCR1 are set to 0, corresponding cancellation sources are masked. When these bits are set to 1, the corresponding cancellation sources are not masked.
The VBUSIN, OVRCLR, CC1_RD, and CC2_RD bits of this register correspond to the VBUSIN1, OVRCLR1, CC1_Rd1, and CC2_Rd1 pins of the USB input/output port, respectively.

Note 1. When writing to this register, see section 52.4, Usage Notes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CC2_RD	CC1_RD	OVRCLR	VBUSIN	—	—	—	—	—	—	DM	DP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	CC2_RD	0	R/W	Mask selection of the cancellation source signal in software standby state 1: Software standby mode, does not mask the cancellation source. 0: Software standby mode, to mask the cancellation source.
10	CC1_RD	0	R/W	Mask selection of the cancellation source signal in software standby state 1: Software standby mode, does not mask the cancellation source. 0: Software standby mode, to mask the cancellation source.
9	OVRCLR	0	R/W	Mask selection of the cancellation source signal in software standby state 1: Software standby mode, does not mask the cancellation source. 0: Software standby mode, to mask the cancellation source.
8	VBUSIN	0	R/W	Mask selection of the cancellation source signal in software standby state 1: Software standby mode, does not mask the cancellation source. 0: Software standby mode, to mask the cancellation source.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	DM	0	R/W	Mask selection of the cancellation source signal in software standby state 1: Software standby mode, does not mask the cancellation source. 0: Software standby mode, to mask the cancellation source.
0	DP	0	R/W	Mask selection of the cancellation source signal in software standby state 1: Software standby mode, does not mask the cancellation source. 0: Software standby mode, to mask the cancellation source.

52.2.26 USB Software Standby Cancel Source Request Register 0 (SSTBCRR0)

SSTBCRR0 is a 16-bit readable/writable*1 register that corresponds to the USB channel 0. This register sets a flag to indicate which cancellation source canceled the software standby mode, and reads and clears the flag after cancellation of the software standby mode.
The VBUSIN, OVRCLR, CC1_RD, and CC2_RD bits of this register correspond to the VBUSIN0, OVRCLR0, CC1_Rd0, and CC2_Rd0 pins of the USB input/output port, respectively.

Note 1. When writing to this register, see section 52.4, Usage Notes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CC2_RD	CC1_RD	OVRCLR	VBUSIN	—	—	—	—	—	—	DM	DP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R	R	R	R	R/(W)*	R/(W)*

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	CC2_RD	0	R/(W)*	Cancellation source flag This bit retains the cancellation source which canceled the software standby mode. 0: No cancellation flag is set. [Clearing condition] • Writing 0 after reading 1 1: Cancellation flag is set. [Setting condition] When software standby is canceled by this source first after transition to the software standby mode
10	CC1_RD	0	R/(W)*	Cancellation source flag This bit retains the cancellation source which canceled the software standby mode. 0: No cancellation flag is set. [Clearing condition] • Writing 0 after reading 1 1: Cancellation flag is set. [Setting condition] When software standby is canceled by this source first after transition to the software standby mode
9	OVRCLR	0	R/(W)*	Cancellation source flag This bit retains the cancellation source which canceled the software standby mode. 0: No cancellation flag is set. [Clearing condition] • Writing 0 after reading 1 1: Cancellation flag is set. [Setting condition] When software standby is canceled by this source first after transition to the software standby mode
8	VBUSIN	0	R/(W)*	Cancellation source flag This bit retains the cancellation source which canceled the software standby mode. 0: No cancellation flag is set. [Clearing condition] • Writing 0 after reading 1 1: Cancellation flag is set. [Setting condition] When software standby is canceled by this source first after transition to the software standby mode
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	DM	0	R/(W)*	Cancellation source flag This bit retains the cancellation source which canceled the software standby mode. 0: No cancellation flag is set. [Clearing condition] • Writing 0 after reading 1 1: Cancellation flag is set. [Setting condition] When software standby is canceled by this source first after transition to the software standby mode
0	DP	0	R/(W)*	Cancellation source flag This bit retains the cancellation source which canceled the software standby mode. 0: No cancellation flag is set. [Clearing condition] • Writing 0 after reading 1 1: Cancellation flag is set. [Setting condition] When software standby is canceled by this source first after transition to the software standby mode

Note: * Writing is permitted only in the case where [Clearing condition] is met.

52.2.27 USB Software Standby Cancel Source Request Register 1 (SSTBCRR1)

SSTBCRR1 is a 16-bit readable/writable*1 register that corresponds to the USB channel 1. This register sets a flag to indicate which cancellation source canceled the software standby mode, and reads and clears the flag after cancellation of the software standby mode.
The VBUSIN, OVRCLR, CC1_RD, and CC2_RD bits of this register correspond to the VBUSIN1, OVRCLR1, CC1_Rd1, and CC2_Rd1 pins of the USB input/output port, respectively.

Note 1. When writing to this register, see section 52.4, Usage Notes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CC2_RD	CC1_RD	OVRCLR	VBUSIN	—	—	—	—	—	—	DM	DP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R	R	R	R	R/(W)*	R/(W)*

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	CC2_RD	0	R/(W)*	Cancellation source flag This bit retains the cancellation source which canceled the software standby mode. 0: No cancellation flag is set. [Clearing condition] • Writing 0 after reading 1 1: Cancellation flag is set. [Setting condition] When software standby is canceled by this source first after transition to the software standby mode
10	CC1_RD	0	R/(W)*	Cancellation source flag This bit retains the cancellation source which canceled the software standby mode. 0: No cancellation flag is set. [Clearing condition] • Writing 0 after reading 1 1: Cancellation flag is set. [Setting condition] When software standby is canceled by this source first after transition to the software standby mode
9	OVRCLR	0	R/(W)*	Cancellation source flag This bit retains the cancellation source which canceled the software standby mode. 0: No cancellation flag is set. [Clearing condition] • Writing 0 after reading 1 1: Cancellation flag is set. [Setting condition] When software standby is canceled by this source first after transition to the software standby mode
8	VBUSIN	0	R/(W)*	Cancellation source flag This bit retains the cancellation source which canceled the software standby mode. 0: No cancellation flag is set. [Clearing condition] • Writing 0 after reading 1 1: Cancellation flag is set. [Setting condition] When software standby is canceled by this source first after transition to the software standby mode
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	DM	0	R/(W)*	Cancellation source flag This bit retains the cancellation source which canceled the software standby mode. 0: No cancellation flag is set. [Clearing condition] • Writing 0 after reading 1 1: Cancellation flag is set. [Setting condition] When software standby is canceled by this source first after transition to the software standby mode
0	DP	0	R/(W)*	Cancellation source flag This bit retains the cancellation source which canceled the software standby mode. 0: No cancellation flag is set. [Clearing condition] • Writing 0 after reading 1 1: Cancellation flag is set. [Setting condition] When software standby is canceled by this source first after transition to the software standby mode

* Writing is permitted only in the case where [Clearing condition] is met.

52.2.28 Deep Standby Control Register (DSCTR)

DSCTR is an 8-bit readable/writable register that selects whether the states of the external memory control pins are retained or not when returning from deep standby mode and specifies the method to start the LSI.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	EBUS KEEPE	RAM BOOT	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	EBUSKEEPE	0	R/W	Retention of External Memory Control Pin State 0: The state of the external memory control pins is not retained when returning from deep standby mode. 1: The state of the external memory control pins is retained when returning from deep standby mode. Note: The following setting is prohibited: (EBUSKEEPE, RAMBOOT) = (1, 0).
6	RAMBOOT	0	R/W	Selection of Method after Returning from Deep Standby Mode Selects an activation method after returning from deep standby mode. 0: Activated according to the boot mode specified for a reset. 1: The program is read from the on-chip data-retention RAM. Instruction fetch from H'8000_0000 Note: The following setting is prohibited: (EBUSKEEPE, RAMBOOT) = (1, 0).
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

52.2.29 Deep Standby Cancel Source Select Register (DSSSR)

DSSSR is a 16-bit readable/writable register that consists of the bits for selecting a source to cancel deep standby mode. The realtime clock alarm interrupt or change on the pins for canceling (PK_4, PK_2, PJ_5, PJ_1, PH_0, PG_6, PG_2, PH_1, PE_1, P6_2, P3_3, P3_1, and NMI) can be selected as a cancel source. The USB return factor can be selected with the USB Deep standby cancel source select register (USBDSSSR). The pins for canceling can be used for canceling deep standby, regardless of pin function settings in the general I/O port.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PK_4	PK_2	PJ_5	PJ_1	PH_0	PG_6	NMI	RTCAR1	RTCAR0	PG_2	PH_1	PE_1	P6_2	P3_3	P3_1
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PK_4	0	R/W	Cancel by Change on PK_4 0: Deep standby mode is not canceled by change on the PK_4 pin. 1: Deep standby mode is canceled by change on the PK_4 pin.
13	PK_2	0	R/W	Cancel by Change on PK_2 0: Deep standby mode is not canceled by change on the PK_2 pin. 1: Deep standby mode is canceled by change on the PK_2 pin.
12	PJ_5	0	R/W	Cancel by Change on PJ_5 0: Deep standby mode is not canceled by change on the PJ_5 pin. 1: Deep standby mode is canceled by change on the PJ_5 pin.
11	PJ_1	0	R/W	Cancel by Change on PJ_1 0: Deep standby mode is not canceled by change on the PJ_1 pin. 1: Deep standby mode is canceled by change on the PJ_1 pin.
10	PH_0	0	R/W	Cancel by Change on PH_0 0: Deep standby mode is not canceled by change on the PH_0 pin. 1: Deep standby mode is canceled by change on the PH_0 pin.
9	PG_6	0	R/W	Cancel by Change on PG_6 0: Deep standby mode is not canceled by change on the PG_6 pin. 1: Deep standby mode is canceled by change on the PG_6 pin.
8	NMI	0	R/W	Cancel by Change on NMI 0: Deep standby mode is not canceled by change on the NMI pin. 1: Deep standby mode is canceled by change on the NMI pin.
7	RTCAR1	0	R/W	Cancel by Realtime Clock Alarm 1 Interrupt 0: Deep standby mode is not canceled by a realtime clock alarm 1 interrupt. 1: Deep standby mode is canceled by a realtime clock alarm 1 interrupt.
6	RTCAR0	0	R/W	Cancel by Realtime Clock Alarm 0 Interrupt 0: Deep standby mode is not canceled by a realtime clock alarm 0 interrupt. 1: Deep standby mode is canceled by a realtime clock alarm 0 interrupt.
5	PG_2	0	R/W	Cancel by Change on PG_2 0: Deep standby mode is not canceled by change on the PG_2 pin. 1: Deep standby mode is canceled by change on the PG_2 pin.
4	PH_1	0	R/W	Cancel by Change on PH_1 0: Deep standby mode is not canceled by change on the PH_1 pin. 1: Deep standby mode is canceled by change on the PH_1 pin.
3	PE_1	0	R/W	Cancel by Change on PE_1 0: Deep standby mode is not canceled by change on the PE_1 pin. 1: Deep standby mode is canceled by change on the PE_1 pin.

Bit	Bit Name	Initial Value	R/W	Description
2	P6_2	0	R/W	Cancel by Change on P6_2 0: Deep standby mode is not canceled by change on the P6_2 pin. 1: Deep standby mode is canceled by change on the P6_2 pin.
1	P3_3	0	R/W	Cancel by Change on P6_2 0: Deep standby mode is not canceled by change on the P3_3 pin. 1: Deep standby mode is canceled by change on the P3_3 pin.
0	P3_1	0	R/W	Cancel by Change on P6_2 0: Deep standby mode is not canceled by change on the P3_1 pin. 1: Deep standby mode is canceled by change on the P3_1 pin.

52.2.30 USB Deep Standby Cancel Source Select Register (USBDSSSR)

USBDSSSR is an 8-bit readable/writable register that selects whether or not to cancel the deep standby mode by the return factor of the channel 0 or 1 USB 2.0 host/function module.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	USBDSCE3	USBDSCE2	USBDSCE1	USBDSCE0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	USBDSCE3	0	R/W	Cancel by the Return Factor (CLRVM) of the Channel 1 USB 2.0 Host/Function Module 0: Deep standby mode is not canceled by the return factor of the channel 1 USB 2.0 host/function module. 1: Deep standby mode is canceled by the return factor of the channel 1 USB 2.0 host/function module.
2	USBDSCE2	0	R/W	Cancel by the Return Factor (CLRVP) of the Channel 1 USB 2.0 Host/Function Module 0: Deep standby mode is not canceled by the return factor of the channel 1 USB 2.0 host/function module. 1: Deep standby mode is canceled by the return factor of the channel 1 USB 2.0 host/function module.
1	USBDSCE1	0	R/W	Cancel by the Return Factor (CLRVM) of the Channel 0 USB 2.0 Host/Function Module 0: Deep standby mode is not canceled by the return factor of the channel 0 USB 2.0 host/function module. 1: Deep standby mode is canceled by the return factor of the channel 0 USB 2.0 host/function module.
0	USBDSCE0	0	R/W	Cancel by the Return Factor (CLRVP) of the Channel 0 USB 2.0 Host/Function Module 0: Deep standby mode is not canceled by the return factor of the channel 0 USB 2.0 host/function module. 1: Deep standby mode is canceled by the return factor of the channel 0 USB 2.0 host/function module.

52.2.31 Deep Standby Cancel Edge Select Register (DSESR)

DSESR is a 16-bit readable/writable register that consists of the bits for selecting an edge to be detected for the pin specified as a deep standby cancel source with DSSSR. This register setting is always valid for canceling deep standby, regardless of the interrupt controller setting.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PK_4E	PK_2E	PJ_5E	PJ_1E	PH_0E	PG_6E	NMIE	—	—	PG_2E	PH_1E	PE_1E	P6_2E	P3_3E	P3_1E
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PK_4E	0	R/W	PK_4 Edge Detection 0: Falling edge of PK_4 is detected. 1: Rising edge of PK_4 is detected.
13	PK_2E	0	R/W	PK_2 Edge Detection 0: Falling edge of PK_2 is detected. 1: Rising edge of PK_2 is detected.
12	PJ_5E	0	R/W	PJ_5 Edge Detection 0: Falling edge of PJ_5 is detected. 1: Rising edge of PJ_5 is detected.
11	PJ_1E	0	R/W	PJ_1 Edge Detection 0: Falling edge of PJ_1 is detected. 1: Rising edge of PJ_1 is detected.
10	PH_0E	0	R/W	PH_0 Edge Detection 0: Falling edge of PH_0 is detected. 1: Rising edge of PH_0 is detected.
9	PG_6E	0	R/W	PG_6 Edge Detection 0: Falling edge of PG_6 is detected. 1: Rising edge of PG_6 is detected.
8	NMIE	0	R/W	NMI Edge Detection 0: Falling edge of NMI is detected. 1: Rising edge of NMI is detected.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PG_2E	0	R/W	PG_2 Edge Detection 0: Falling edge of PG_2 is detected. 1: Rising edge of PG_2 is detected.
4	PH_1E	0	R/W	PH_1 Edge Detection 0: Falling edge of PH_1 is detected. 1: Rising edge of PH_1 is detected.
3	PE_1E	0	R/W	PE_1 Edge Detection 0: Falling edge of PE_1 is detected. 1: Rising edge of PE_1 is detected.
2	P6_2E	0	R/W	P6_2 Edge Detection 0: Falling edge of P6_2 is detected. 1: Rising edge of P6_2 is detected.
1	P3_3E	0	R/W	P3_3 Edge Detection 0: Falling edge of P3_3 is detected. 1: Rising edge of P3_3 is detected.
0	P3_1E	0	R/W	P3_1 Edge Detection 0: Falling edge of P3_1 is detected. 1: Rising edge of P3_1 is detected.

52.2.32 Deep Standby Cancel Source Flag Register (DSFR)

DSFR is a 16-bit readable/writable register composed of two types of bits. One is the flag that confirms which source canceled deep standby mode. The other is the bit that releases the state of pins after canceling deep standby mode. When deep standby mode is canceled by an interrupt (NMI or realtime clock alarm interrupt) and changes on the pins for canceling, this register retains the previous data although power-on reset exception handling is executed. When deep standby mode is canceled by a power-on reset, this register is initialized to H'0000.
All flags must be cleared immediately before transition to deep standby mode.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IO KEEP	PK_4F	PK_2F	PJ_5F	PJ_1F	PH_0F	PG_6F	NMIF	RTC ARF1	RTC ARF0	PG_2F	PH_1F	PE_1F	P6_2F	P3_3F	P3_1F
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Bit	Bit Name	Initial Value	R/W	Description
15	IOKEEP	0	R/(W)*1	Release of Pin State Retention Releases the retention of the pin state after canceling deep standby mode 0: Pin state not retained [Clearing condition] • Writing 0 after reading 1 1: Pin state retained [Setting condition] • When deep standby mode is entered
14	PK_4F	0	R/(W)*1	PK_4 Flag 0: No change on the PK_4 pin 1: Change on the PK_4 pin
13	PK_2F	0	R/(W)*1	PK_2 Flag 0: No change on the PK_2 pin 1: Change on the PK_2 pin
12	PJ_5F	0	R/(W)*1	PJ_5 Flag 0: No change on the PJ_5 pin 1: Change on the PJ_5 pin
11	PJ_1F	0	R/(W)*1	PJ_1 Flag 0: No change on the PJ_1 pin 1: Change on the PJ_1 pin
10	PH_0F	0	R/(W)*1	PH_0 Flag 0: No change on the PH_0 pin 1: Change on the PH_0 pin
9	PG_6F	0	R/(W)*1	PG_6 Flag 0: No change on the PG_6 pin 1: Change on the PG_6 pin
8	NMIF	0	R/(W)*1	NMI Flag 0: No interrupt on NMI pin 1: Interrupt on NMI pin
7	RTCARF1	0	R/(W)*1	RTCAR1 Flag 0: No realtime clock alarm 1 interrupt generated 1: Realtime clock alarm 1 Interrupt generated
6	RTCARF0	0	R/(W)*1	RTCAR0 Flag 0: No realtime clock alarm 0 interrupt generated 1: Realtime clock alarm 0 Interrupt generated
5	PG_2F	0	R/(W)*1	PG_2 Flag 0: No change on the PG_2 pin 1: Change on the PG_2 pin

Bit	Bit Name	Initial Value	R/W	Description
4	PH_1F	0	R/(W)*1	PH_1 Flag 0: No change on the PH_1 pin 1: Change on the PH_1 pin
3	PE_1F	0	R/(W)*1	PE_1 Flag 0: No change on the PE_1 pin 1: Change on the PE_1 pin
2	P6_2F	0	R/(W)*1	P6_2 Flag 0: No change on the P6_2 pin 1: Change on the P6_2 pin
1	P3_3F	0	R/(W)*1	P3_3 Flag 0: No change on the P3_3 pin 1: Change on the P3_3 pin
0	P3_1F	0	R/(W)*1	P3_1 Flag 0: No change on the P3_1 pin 1: Change on the P3_1 pin

Note 1. Only 0 can be written after reading 1 to clear the flag.

52.2.33 USB Deep standby cancel source flag register (USBDSFR)

USBDSFR is an 8-bit readable/writable register that serves as a flag to check whether deep standby mode has been canceled by the channel 0 or 1 USB 2.0 host/function module.
When deep standby mode is canceled on the channel 0 or 1 USB 2.0 host/function module, power-on reset exception handling is executed, but this register holds the previous value.
When deep standby mode is canceled by a power-on reset, this register is initialized to H'00.
All flags of this register must be cleared immediately before the LSI chip enters deep standby mode.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	USBDSF3	USBDSF2	USBDSF1	USBDSF0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	USBDSF3	0	R/(W)*	Channel 1 USB 2.0 Host/Function Module Flag (CLRVM) 0: The return source that cancels the deep standby mode is not in the channel 1 USB 2.0 host/function module. 1: The return source that cancels the deep standby mode is in the channel 1 USB 2.0 host/function module.
2	USBDSF2	0	R/(W)*	Channel 1 USB 2.0 Host/Function Module Flag (CLRVP) 0: The return source that cancels the deep standby mode is not in the channel 1 USB 2.0 host/function module. 1: The return source that cancels the deep standby mode is in the channel 1 USB 2.0 host/function module.
1	USBDSF1	0	R/(W)*	Channel 0 USB 2.0 Host/Function Module Flag (CLRVM) 0: The return source that cancels the deep standby mode is not in the channel 0 USB 2.0 host/function module. 1: The return source that cancels the deep standby mode is in the channel 0 USB 2.0 host/function module.
0	USBDSF0	0	R/(W)*	Channel 0 USB 2.0 Host/Function Module Flag (CLRVP) 0: The return source that cancels the deep standby mode is not in the channel 0 USB 2.0 host/function module. 1: The return source that cancels the deep standby mode is in the channel 0 USB 2.0 host/function module.

Note: Only 0 can be written to clear the flag after reading 1.

52.2.34 Deep standby cancel Oscillation stability count register (DSCNT)

DSCNT is a 16-bit readable/writable register. When deep standby mode is canceled by other than a power-on reset, this register serves as the oscillation stabilization counter for recovery.
This register is initialized to H'0000 at power on reset or after canceling deep standby mode.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CNTD[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	CNTD[7:0]	H'00	R/(W)*	Initial value of oscillation stability count for deep standby canceling Set the start value of counting except reset. When the count value overflows, deep standby mode is canceled and power-on reset exception handling is executed. Table 52.3 shows the setting contents of CNTD[7:0] bits.

Table 52.3 Oscillation Settling Time for deep standby canceling (other than power on reset)

	Oscillation Settling Time	
CNTD[7:0] Setting Value	P0φ = 33 MHz	P0φ = 27.5 MHz
H'00 (Initial Value)	15.8 ms	19.0 ms
H'A0	5.9 ms	7.1 ms
H'B0	4.9 ms	5.9 ms
H'C0	3.9 ms	4.7 ms
H'D0	2.9 ms	3.5 ms
H'E0	1.9 ms	2.3 ms

Note: Oscillation Settling Time(s) = (H'FF – CNTD[7:0] Setting Value)/(P0φ/2048)
Setting for 1ms or less is prohibited.

When the setting of the RTC1XT and RTC0XT bits in the RTCXTALSEL register is 0, the EXTAL clock stops oscillating when the chip is placed on deep standby. Set the CNTD[7:0] bits to obtain an interval that is no less than the sum of the on-chip oscillation circuit oscillation settling time (t_{ROSC}) of the EXTAL and the internal PLL circuit oscillation settling time (t_{POSC}).
When the setting of one or both of the RTC1XT or RTC0XT bits is 1, the EXTAL clock continues oscillating even on deep standby. In this case, set the CNTD[7:0] bits to obtain an interval that is no less than the internal PLL circuit oscillation settling time (t_{POSC}).

52.2.35 XTAL Crystal Oscillator Gain Control Register (XTALCTR)

XTALCTR is an 8-bit readable/writable register that controls the gain of the crystal oscillator for XTAL and the realtime clock.

If the realtime clock uses the EXTAL input, the GAIN0 bit retains the previous value when software standby mode or deep standby mode is canceled by a source other than a power-on reset. If the realtime clock does not use the EXTAL input, this bit is initialized to 0 when software standby or deep standby mode is entered.

This bit is initialized to H'00 when software standby or deep standby mode is canceled by a power-on reset.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GAIN0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GAIN0	0	R/W	XTAL Crystal Oscillator (EXTAL or XTAL Pin) Gain Select 0: Large gain 1: Small gain

52.2.36 RTCXTAL select register (RTCXTALSEL)

RTCXTALSEL is a 16-bit readable/writable register that controls EXTAL oscillation in deep standby mode.

When RTC1XT and RTC0XT bits are both 0, EXTAL stops oscillation simultaneously with deep standby mode.

Note: When writing to this register, see section 52.4, Usage Notes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RTC1XT	RTC0XT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x	x
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	RTC1XT	x	R/W	RTC1 EXTAL used bit 0: E XTAL unused 1: EXTAL used
0	RTC0XT	x	R/W	RTC0 EXTAL used bit 0: EXTAL unused 1: EXTAL used

52.3 Operation

52.3.1 Sleep Mode

(1) Transition to Sleep Mode

When the WFI instruction is executed while the STBY bit of STBCR1 is 0, the program execution state is shifted to the sleep mode. Although the CPU halts after executing the WFI instruction, the contents of its internal registers remain unchanged. The on-chip peripheral modules continue to run. The clock output from the CKIO pin is continued.
Note: When writing to STBCR1 register, see section 52.4, Usage Notes.

(2) Canceling Sleep Mode

Sleep mode is canceled by an interrupt (NMI, IRQ, and on-chip peripheral module) or a power-on reset.

- Canceling by an interrupt
  When an NMI, IRQ, or on-chip peripheral module interrupt occurs, sleep mode is canceled and interrupt exception handling is executed. When the priority level of the generated interrupt is lower than the setting of the interrupt controller execution priority register, or the interrupt by the on-chip peripheral module is disabled on the module side, the interrupt request is not accepted and sleep mode is not canceled. An interrupt request occurs with an interrupt masked by bits 6 (the FIQ mask bit) and 7 (the IRQ mask bit) in the CPSR register in the CPU, sleep mode is canceled but interrupt exception handling is not executed.
- Canceling by a reset
  Sleep mode is canceled by a power-on reset.

52.3.2 Software Standby Mode

(1) Transition to Software Standby Mode

Executing the WFI instruction with the STBY bit and DEEP bit in STBCR1 set to 1 and 0, respectively, leads to the LSI chip entering software standby mode from the program execution state. In software standby mode, not only the CPU but also the clock and on-chip peripheral modules other than the USB module are halted. To halt the USB module as well, set the USBCTR.DIRPD bit to 1 before executing the procedure for transition to standby mode. For details, see section 52.4.3, Usage Notes when USB 2.0 Host/Function Modules are Not to be Used.
Only make this setting when the USB module is not to be used. Output of the clock signal from the CKIO pin also stops. The contents of the CPU and cache registers remain unchanged. Some registers of on-chip peripheral modules are, however, initialized. For the states of on-chip peripheral module registers in software standby mode, see section 55, Register States.

The CPU takes one cycle to finish writing to STBCR1, and then executes processing for the next instruction. However, it takes one or more cycles to actually write. Therefore, execute a WFI instruction after reading STBCR1 to have the values written to STBCR1 by the CPU to be definitely reflected in the WFI instruction.
After a WFI instruction is issued, the hardware automatically stops the bus master following the wait for completion of the issuing-finished request from the bus master and the transition to software standby proceeds.
Since the transition to software standby is not possible if completion of the issuing-finished request is not possible at this time, do not proceed with access to the registers of modules in the module-standby state and so on.
To suppress an unintended request from a bus master, use the software to stop bus masters before transition to the software standby mode.

The procedure for switching to software standby mode is as follows:

1. Set the standby_mode_en bit of the power control register in the PL310 to 1. For the details of this register, see CoreLink Level 2 Cache Controller L2C-310 Technical Reference Manual issued by Arm Ltd.
2. Clear the TME bit in the watchdog timer control/status register (WTCSR) of the watchdog timer to 0 to stop the watchdog timer.
3. Set the CKS[3:0] bits in WTCSR and the watchdog timer counter (WTCNT) so that the watchdog timer overflow period is no less than the oscillation settling time after release from the standby state.
4. When an IRQ interrupt signal is used as a software standby cancellation source, change the input/output port to a desired IRQ[*1] and clear all interrupt request flags for the IRQ7 to IRQ0 pins. Then, set the IRQMSK bit to 1 in the IRQ interrupt request register (IRQRR) of the interrupt controller to cancel the cancellation source signal mask of IRQ.
   For details of the interrupt request register, see section 7.3.3, IRQ Interrupt Request Register (IRQRR).
5. When using the USB software standby cancellation source signal, change the input/output port to a desired USB signal[*2] and set the corresponding cancellation source mask bit to 1 in the USB Software standby cancel source control register (SSTBCCR0/1) to cancel the mask. At this time, clear the cancellation source bit of the USB used as a cancellation source by the USB Software Standby Cancel Source Request Register (SSTBCRR0/1).
   For details about the USB Software standby cancel source control registers, see section 52.2.24, USB Software Standby Cancel Source Control Register 0 (SSTBCCR0) and section 52.2.25, USB Software Standby Cancel Source Control Register 1 (SSTBCCR1). For details about the USB Software standby cancel source request registers, see section 52.2.26, USB Software Standby Cancel Source Request Register 0 (SSTBCRR0) and section 52.2.27, USB Software Standby Cancel Source Request Register 1 (SSTBCRR1).
6. After setting the STBY and DEEP bits in STBCR1 to 1 and 0 respectively, read STBCR1. Then, execute a WFI instruction.

Note 1. If the desired IRQ has already been used, input/output port switching is not required.
Note 2. If the desired USB signal has already been used, input/output port switching is not required.

(2) Canceling Software Standby Mode

Software standby mode is canceled by interrupts (NMI or IRQ), USB software standby cancel source, or a reset (power-on reset). Clock signal starts to be output from the CKIO pin.

- Canceling by an interrupt or USB software standby cancel source
  When the falling edge or rising edge of the NMI pin (selected by the NMI edge select bit (NMIE) in the interrupt control register 0 (ICR0) of the interrupt controller) or the falling edge or rising edge of an IRQ pin (IRQ7 to IRQ0) (selected by the IRQn sense select bits (IRQn1S and IRQn0S) in interrupt control register 1 (ICR1) of the interrupt controller) or the falling edge or rising edge of the USB software standby cancel source (DP, DM, VBUSIN, OVRCLR, CC1_RD, or CC2_RD) is detected, clock oscillation is started.
  This clock pulse is supplied only to the oscillation settling counter (watchdog timer) used to count the oscillation settling time.
  When the time specified by the clock select bits (CKS[3:0]) in the watchdog timer control/status register (WTCSR) of the watchdog timer has elapsed before the transition to software standby mode, the watchdog timer overflow occurs.
  Since this overflow indicates that the clock has been stabilized, the clock pulse will be supplied to the entire LSI. Software standby mode is thus cleared and NMI interrupts exception handling (IRQ interrupt exception handling in case of IRQ, and USB interrupt exception handling in case of USB) is started. If the priority level of the generated interrupt is lower than the setting of the interrupt controller execution priority register, the interrupt request is not accepted and sleep mode is entered. So, cancel the software standby mode by an interrupt whose priority level is higher than the setting of the interrupt controller execution priority register. When canceling software standby mode by the NMI interrupt, IRQ interrupts, or USB software standby cancellation source, set the CKS[3:0] bits so that the watchdog timer overflow period will be equal to or longer than the oscillation settling time. The clock output phase of the CKIO pin may be unstable immediately after detecting an interrupt and until software standby mode is canceled.
- Canceling by a reset
  When the RES# pin is driven low, software standby mode is canceled and the LSI enters the power-on reset state. After that, if the RES# pin is driven high, the power-on reset exception handling is started.
  Keep the RES# pin low until the clock oscillation settles. The internal clock will continue to be output to the CKIO pin.

(3) Note on Transition to Software Standby Mode

Release from software standby mode is triggered by interrupts (NMI or IRQ) or a power-on reset. If, however, a WFI instruction and an interrupt other than NMI and IRQ are generated at the same time, software standby mode may be canceled due to acceptance of the interrupt.
When initiating a transition to software standby mode, make settings so that interrupts other than NMI and IRQ are not generated before execution of the WFI instruction.

(4) Note on Canceling Software Standby Mode

After software standby mode is canceled, unstable clock pulses are output from the CKIO pin during oscillation settling time. To prevent malfunction due to the unstable pulses, the CKOEN[1:0] bits in FRQCR should be modified.

52.3.3 Software Standby Mode Application Example

This example describes a transition to software standby mode on the falling edge of the NMI signal, and cancellation on the rising edge of the NMI signal. The timing is shown in Figure 52.2.
When the NMI pin is changed from high to low level while the NMI edge select bit (NMIE) in the interrupt control register 0 (ICR0) is set to 0 (falling edge detection), the NMI interrupt is accepted. When the NMIE bit is set to 1 (rising edge detection) by the NMI exception service routine, the STBY and DEEP bits in STBCR1 are set to 1 and 0 respectively, and a WFI instruction is executed, software standby mode is entered. Thereafter, software standby mode is canceled when the NMI pin is changed from low to high level.

Note: When writing to STBCR1 register, see section 52.4, Usage Notes.

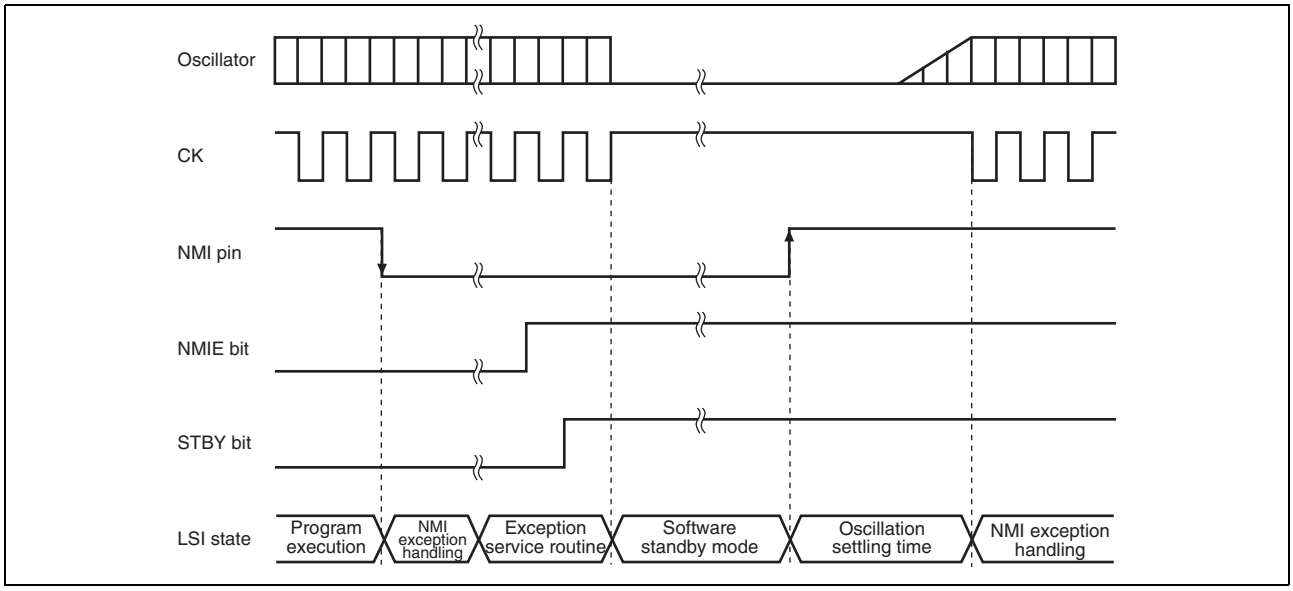


Figure 52.2 NMI Timing in Software Standby Mode (Application Example)

52.3.4 Deep Standby Mode

(1) Transition to Deep Standby Mode

The LSI switches from a program execution state to deep standby mode by executing the WFI instruction when the STBY bit and DEEP bit in STBCR1 are set to 1. In deep standby mode, not only the CPU, clocks, and on-chip peripheral modules but also power supply is turned off excluding the on-chip data-retention RAM area specified by the RRAMKP3 to RRAMKP0 bits in RRAMKP and realtime clock. This can significantly reduce power consumption. Therefore, data in the registers of the CPU, cache, and on-chip peripheral modules are not retained. Pin state values immediately before the transition to deep standby mode are retained.
The CPU takes one cycle to finish writing to STBCR1, and then executes processing for the next instruction. However, it actually takes one or more cycles to write. Therefore, execute a WFI instruction after reading STBCR1 to reflect the values written to STBCR1 by the CPU in the WFI instruction without fail.
After a WFI instruction is issued, the hardware automatically stops the bus master following the wait for completion of the issuing-finished request from the bus master and the transition to deep standby proceeds.
Since the transition to deep standby is not possible if completion of the issuing-finished request is not possible at this time, do not proceed with access to the registers of modules in the module-standby state and so on.
In addition, to suppress unintended requests from bus masters, the software should stop the bus masters before executing the procedure for the transition to deep standby mode. When not using the return factor of the USB 2.0 host/function module, execute the procedure in section 52.4.3, Usage Notes when USB 2.0 Host/Function Modules are Not to be Used before executing the procedure for the transition to deep standby mode.

The procedure for switching to deep standby mode is as follows. Figure 52.3 also shows its flowchart.

1. Set the standby_mode_en bit of the power control register in the PL310 to 1. For the details of this register, see CoreLink Level 2 Cache Controller L2C-310 Technical Reference Manual issued by Arm Ltd.
2. Set the RRAMKP3 to RRAMKP0 bits in RRAMKP for the corresponding on-chip data-retention RAM area that must be retained. Transfer the programs to be retained to the specified areas of the on-chip data-retention RAM.
3. Set the RAMBOOT and EBUSKEEPE bits in DSCTR to specify the activation method for returning from deep standby mode and to select whether the external memory control pin status is retained or not.
4. When canceling deep standby, set the corresponding bit in DSSSR or USBDSSSR, to select the pin or source to cancel deep standby mode. In this case, specify the input signal detection mode for the selected pin with the corresponding bit in DSESR. Also, set the oscillation stabilization time setting with the corresponding bit in the Deep standby cancel Oscillation stability count register (DSCNT).
5. Read any address on each page of the on-chip data-retention RAM area that must be retained. When this is not executed, data last written may not be written to the on-chip data-retention RAM. If there is a write to the on-chip data-retention RAM after this time, execute this processing after the last write to the on-chip data-retention RAM.
6. Set the STBY and DEEP bits in the STBCR1 register to 1, and then read this register.
7. Clear the flag in the DSFR, USBDSFR register.
8. Set the CPU interface control register (GICC_CTLR) of the interrupt controller to 0 so that the CPU is not notified of interrupts other than NMIs. Then, read the GICC_CTLR register.
9. Execute the WFI instruction.
10. The conflict between the instruction and the NMI interrupt may prevent transition to deep standby mode. After executing the WFI instruction, locate the branch instruction to return to step 9.

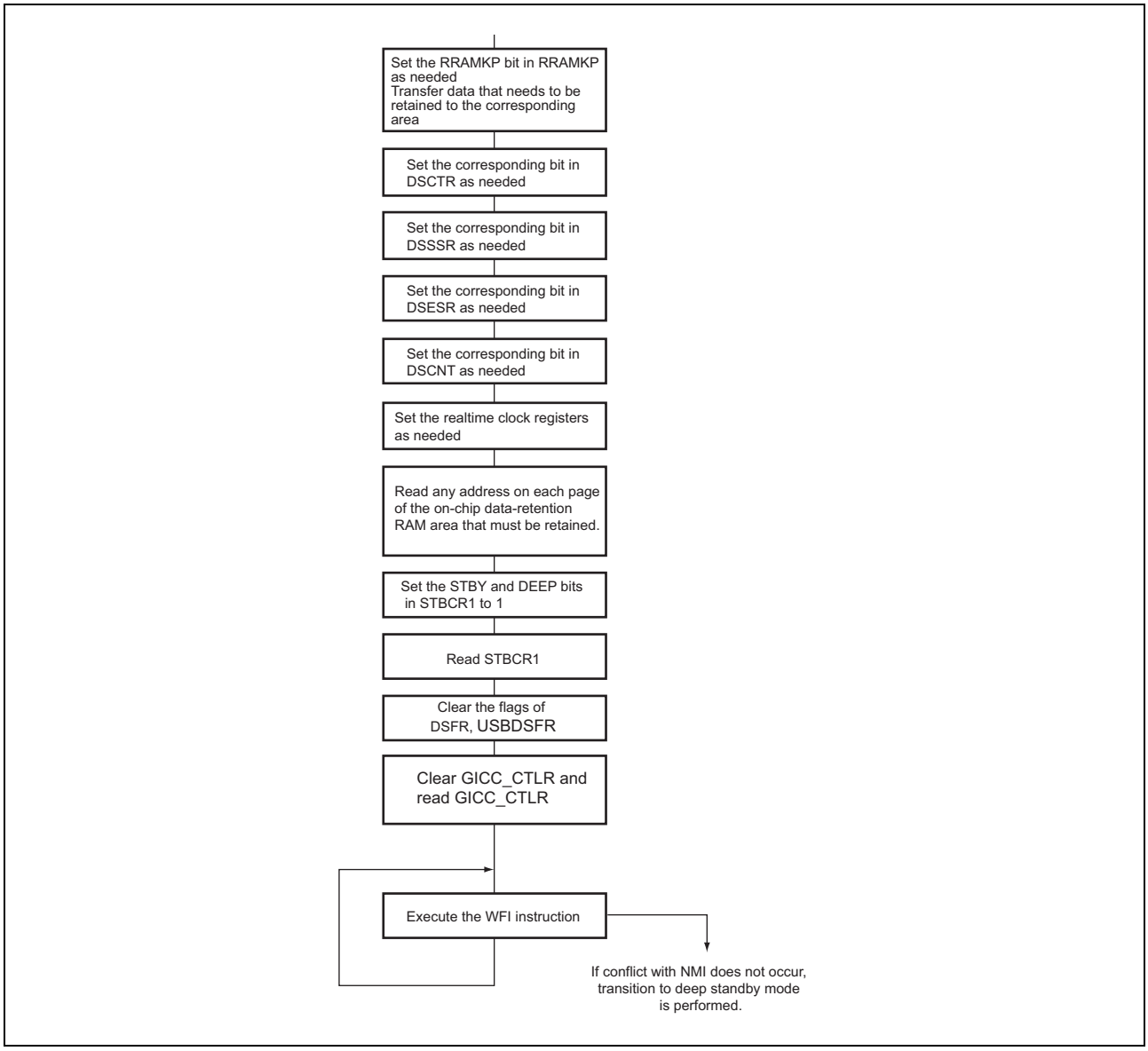


Figure 52.3 Flowchart of Transition to Deep Standby Mode

(2) Canceling Deep Standby Mode

Deep standby mode is canceled by interrupts (NMI or alarm interrupt of realtime clocks 0 and 1 or cancel source of the channels 0 and 1 USB 2.0 host/function modules), change on the pins for canceling, or a reset (power-on reset). The interrupt from the channels 0 and 1 USB 2.0 host/function modules can always cancel deep standby mode regardless of the setting value of the execution priority register of the interrupt controller. The realtime clock 0 or 1 alarm interrupt can always cancel deep standby mode regardless of the setting of the execution priority register for the interrupt controller and of the alarm interrupt enable flag (RCR1.AIE). This interrupt can also always cancel this mode regardless of the states of bit 6 (the FIQ mask bit) and bit 7 (the IRQ mask bit) in the CPSR register of the CPU. When canceling the mode by a source other than a reset, a power-on reset exception handling is executed instead of an interrupt exception handling.

Figure 52.4 shows the flowchart of canceling deep standby mode.

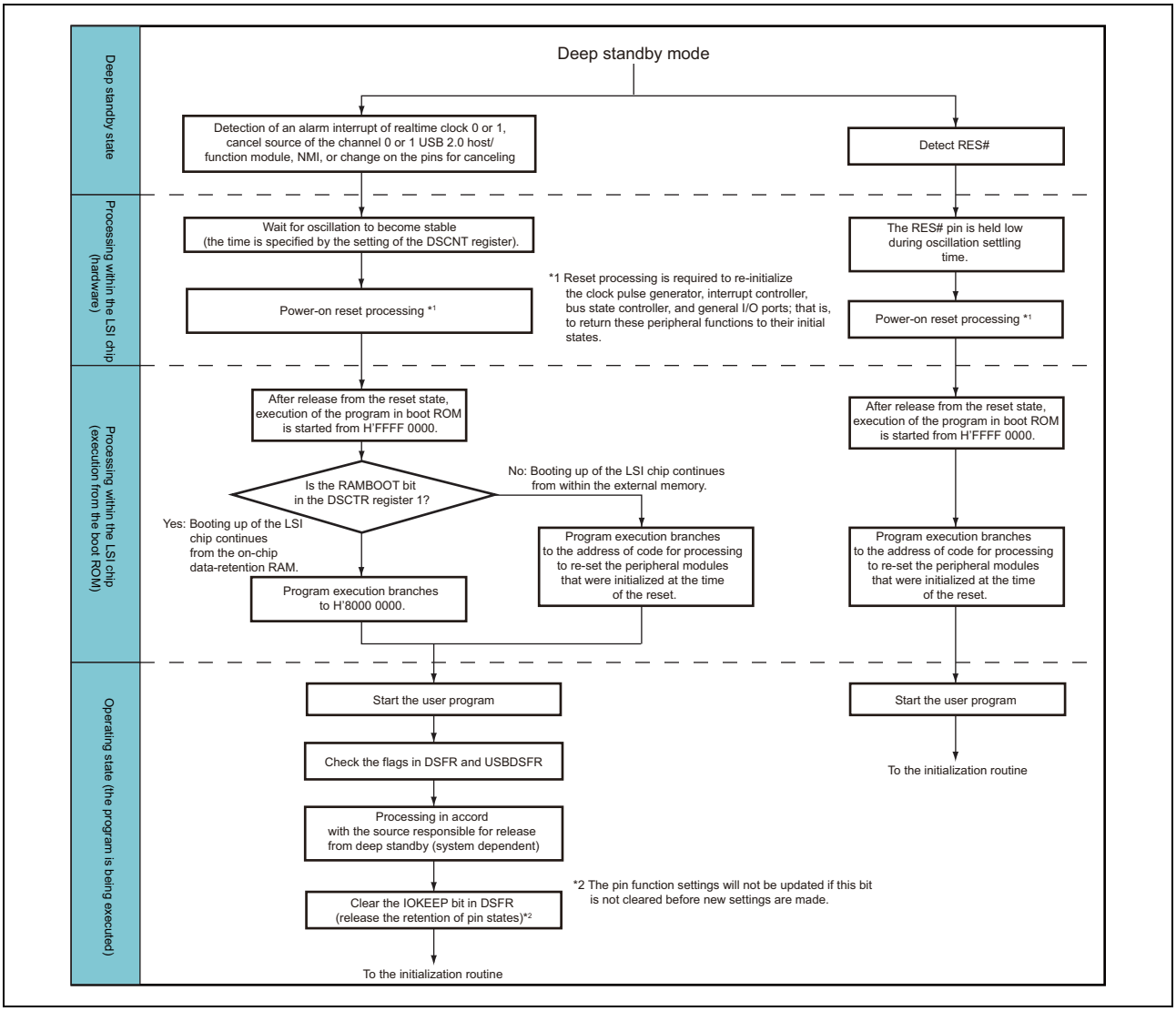


Figure 52.4 Flowchart of Canceling Deep Standby Mode

- Canceling by a source other than a reset
  When the falling or rising edge of the NMI pin (selected by a corresponding bit in DSESR) or falling or rising edge of the pins for canceling (selected by a corresponding bit in DSESR) is detected or a realtime clock 0 or 1 alarm interrupt (see section 16.3.6, Alarm Function) or channel 0 or 1 USB 2.0 host/function module interrupt is generated, clock oscillation is started after the wait time for the power supply stabilization settling time. After the oscillation settling time has elapsed (setting the corresponding bits in DSCNT), deep standby mode is cancelled and the power-on reset exception handling is executed.
  The clock output phase of the CKIO pin may be unstable immediately after detecting a cancel source and until deep standby mode is canceled.
  The detecting of the NMI pin, the pins for canceling, and the realtime clock 0 or 1 alarm interrupt becomes enabled when the corresponding bits in DSSSR are set. Likewise, the detecting of the interrupt from the channels 0 and 1 USB 2.0 host/function modules becomes enabled when the corresponding bits in USBDSSSR are set. The detected cancel sources are kept, but they are reflected to DSFR and USBDSFR after canceling the deep standby mode.
  When the CPU accepts any interrupts and reads the interrupt response register (ICCIAR), all of the cancel sources that are kept are cleared. When the CPU enters the deep standby mode as the detected cancel sources are kept, the deep standby mode is canceled immediately after the CPU enters the deep standby mode.
- Canceling with a reset
  Driving the RES# pin low cancels deep standby mode and causes a transition to the power-on reset state. After this, driving the RES# pin high initiates power-on reset exception handling. Output of the internal clock from the CKIO pin also starts by driving the RES# pin low.
  Keep the RES# pin low until the clock oscillation has settled.

(3) Operation after Canceling Deep Standby Mode

After canceling deep standby mode, the LSI can be activated through the external memory or from the on-chip data-retention RAM, which can be selected by setting the RAMBOOT bit in DSCTR. By setting the EBUSKEEPE bit, the states of the external memory control pins can be retained even after cancellation of deep standby mode. Table 52.4 shows the pin states after cancellation of deep standby mode according to the setting of each bit. Table 52.4 lists the external memory control pins.

Table 52.4 Pin States after Cancellation of Deep Standby Mode and System Activation Method by the DSCTR Settings

EBUSKEEPE Bit	RAMBOOT Bit	Activation Method	Pin States After Cancellation of Deep Standby Mode
0	0	External memory	The states of the external memory control pins are not retained. After cancellation of deep standby mode, the retention of the external memory control pin states is cancelled. For other pins, the retention of their states is cancelled when the IOKEEP bit is cleared.
0	1	On-chip data-retention RAM	The states of the external memory control pins are not retained. After cancellation of deep standby mode, the retention of the external memory control pin states is cancelled. For other pins, the retention of their states is cancelled when the IOKEEP bit is cleared.
1	0	—	Setting prohibited.
1	1	On-chip data-retention RAM	The states of the external memory control pin are retained. The retention of the states of the external memory control pins and other pins is cancelled when the IOKEEP bit is cleared.

Table 52.5 External Memory Control Pins in Different Modes

Boot Mode 0 (CS0 Area: Bus Width: 16 Bits)	Boot Mode 1 (eSD Boot)	Boot Mode 2 (eMMC Boot)	Boot Mode 3 (Serial Flash Boot 3.3 V)	Boot Mode 4 (Octal-SPI Flash Boot)	Boot Mode 5 (SPIBSC: HyperFlash Boot 1)	Boot Mode 6 (OctaFlash Boot)	Boot Mode 7 (HyperFlash Boot 2)
A[20:1]	SD0_CLK			QSPI0_SPCLK		OM_SCLK	HM_CK
D[15:0]	SD0_CMD			QSPI0_IO0			HM_CK#
CS0	SD0_DAT0			QSPI0_IO1		OM_CS0#	HM_CS0#
RD	SD0_DAT1			QSPI0_IO2		OM_CS1#	HM_CS1#
CKIO	SD0_DAT2			QSPI0_IO3		OM_DQS	HM_RWDS
	SD0_DAT3			QSPI0_SSL		OM_SIO0	HM_DQ0
	SD0_DAT4			QSPI1_SPCLK		OM_SIO1	HM_DQ1
	SD0_DAT5			QSPI1_IO0		OM_SIO2	HM_DQ2
	SD0_DAT6			QSPI1_IO1		OM_SIO3	HM_DQ3
	SD0_DAT7			QSPI1_IO2		OM_SIO4	HM_DQ4
	SD0_RST#			QSPI1_IO3		OM_SIO5	HM_DQ5
				QSPI1_SSL		OM_SIO6	HM_DQ6
				RPC_RESET#		OM_SIO7	HM_DQ7
				RPC_WP#		OM_RESET#	HM_RESET#
				RPC_INT#			

When deep standby mode is canceled by interrupts (NMI or realtime clock alarm 0 or 1) or changes on the pins for canceling, the deep standby cancel source flag register (DSFR) can be used to confirm which source has canceled the mode. In case of a USB interrupt, it can be confirmed by the USB Deep standby cancel source flag register (USBDSFR). Pins retain the state immediately before the transition to deep standby mode.

However, in system activation through the external memory, the retention of the states of the external memory control pins is cancelled so that programs can be fetched after cancellation of deep standby mode. Other pins, after cancellation of deep standby mode, continue to retain the pin states until writing 0 to the IOKEEP bit in DSFR after reading 1 from the same bit. In system activation from the on-chip data-retention RAM (EBUSKEEPE bit = 1 and RAMBOOT bit = 1 setting), after cancellation of deep standby mode, both the external memory control pins and other pins continues to

retain the pin states until writing 0 to the IOKEEP bit in DSFR after reading 1 from the same bit. Reconfiguration of peripheral functions is required to return to the previous state of deep standby mode. Peripheral functions include all functions such as the clock pulse generator, interrupt controller, general I/O ports, and peripheral modules. After the reconfiguration, the retention of the pin state can be canceled and the LSI returns to the state prior to the transition to deep standby mode by reading 1 from the IOKEEP bit in DSFR and then writing 0 to it.

(4) Notes on Transition to Deep Standby Mode

If multiple canceling sources have been specified and multiple canceling sources are input, multiple cancel source flags will be set.

52.3.5 Module Standby Function

(1) Transition to Module Standby Function

Setting an MSTP bit in a standby control register to 1 halts supply of the clock signal to the corresponding on-chip peripheral module. This function can be used to reduce power consumption both while a program is running and in sleep mode. Disable a module before placing it in the module standby state. If interrupt requests are enabled, set the respective register in the target module or in the interrupt controller to disable interrupt requests. If DMA transfer requests are enabled, set the respective register in the target module to disable DMA transfer requests from the module and then set the respective register in the direct memory access controller to stop DMA transfers. In addition, do not attempt to access a module's registers while it is in the module standby state. The procedure for transitions to the module standby state depends on whether the STBREQ register has a bit for the corresponding module.

(a) Procedure for transition to module standby of modules for which the STBREQ register does not have a corresponding bit

1. Set the MSTP bit of the corresponding module to 1.

(b) Procedure for transition to module standby of modules for which the STBREQ register has a corresponding bit

1. Set the corresponding bit in the STBREQ register to 1 to generate a request to stop the module.
2. Confirm that the module is ready to be stopped by the corresponding bit in the STBACK register being set to 1.
3. Set the MSTP bit of the corresponding module to 1.

For states of registers in the module standby function, see section 55, Register States.

(2) Canceling Module Standby Function

Release from the module standby state can be achieved in two ways: starting the module by a power-on reset while the MSTP bit is set to 1, then clearing the MSTP bit to 0, or setting the MSTP bit to 0 to activate the module, setting the MSTP bit to 1 to place the module in the standby state, and then setting the MSTP bit to 0 again. If you use the latter approach, the procedure for release from module standby depends on whether the STBREQ register has a bit for the corresponding module.

- Release from the module standby state after the activation of the module by a power-on reset while the MSTP bit is set to 1

1. Clear the MSTP bit to 0.
2. After that, dummy-read the same register.

- Release from the module standby state after a transition to standby following activation of the module

(a) Procedure for release from module standby of modules for which the STBREQ register does not have a corresponding bit

1. Clear the MSTP bit to 0, then dummy-read the same register.

(b) Procedure for release from module standby of modules for which the STBREQ register has a corresponding bit

1. Clear the MSTP bit to 0, then dummy-read the same register.
2. Clear the corresponding bit in the STBREQ register to 0 to cancel the request to stop the module.
3. Confirm that the corresponding bit in the STBACK register has been cleared to 0, indicating the completion of release from standby.

52.3.6 Software Reset

Initialization equivalent to a power-on reset is only applied to the selected modules. The procedure for transitions to the software reset state varies according to whether the STBREQ register has a bit for the corresponding module. Follow the procedures below according to whether this is or is not the case. If DMA transfer requests are enabled, make the settings to stop the DMA transaction for the corresponding channel before transitions to the software reset state. For stopping the DMA transaction, see section 9.7.11, Transfer Status.

(1) Transition to Software Reset State

(a) Procedure for transition to the software reset state of modules for which the STBREQ register does not have a corresponding bit

1. Set the SRST bit of the corresponding module to 1, then dummy-read the same register.

(b) Procedure for transition to the software reset state of modules for which the STBREQ register has a corresponding bit

1. Set the corresponding bit in the STBREQ register to 1 to generate a request to stop the module.
2. Confirm that the corresponding bit in the STBACK register has been set to 1.
3. Set the SRST bit of the corresponding module to 1, then dummy-read the same register.

(2) Canceling Software Reset

(a) Procedure for release from the software reset state of modules for which the STBREQ register does not have a corresponding bit

1. Clear the SRST bit of the corresponding module to 0, then dummy-read the same register.

(b) Procedure for release from the software reset state of modules for which the STBREQ register has a corresponding bit

1. Clear the SRST bit of the corresponding module to 0, then dummy-read the same register.
2. Clear the corresponding bit in the STBREQ register to 0 to cancel the request to stop the module.
3. Confirm that the corresponding bit in the STBACK register has been cleared to 0.

52.3.7 Adjustment of XTAL Crystal Oscillator Gain

The gain of the crystal oscillator for XTAL and the realtime clock can be adjusted using the GAIN0 bit in XTALCTR. To modify the gain of the signal on the EXTAL or XTAL pin, PLL settling time is needed. The settling time is counted using the on-chip watchdog timer.

After a change to the value of the GAIN0 bit, the hardware automatically stops the bus master following the wait for completion of the issuing-finished request from the bus master and the change to the gain for the crystal oscillator in use with XTAL proceeds.

Since the change to the gain for the crystal oscillator in use with XTAL cannot start if completion of the issuing-finished request is not possible at this time, do not proceed with access to the registers of modules in the module-standby state and so on.

To suppress an unintended request from a bus master, use the software to stop bus masters before changing the gain of the crystal oscillator for XTAL.

1. The large gain is selected in the initial state.
2. Set the standby_mode_en bit of the power control register in the PL310 to 1. For the details of this register, see CoreLink Level 2 Cache Controller L2C-310 Technical Reference Manual issued by Arm Ltd.
3. Set the watchdog timer so that the specified settling time should be obtained and stop the watchdog timer.
   Specifically, the following settings are necessary:
   TME in WTCSR = 0: Stop the watchdog timer.
   CKS[3:0] in WTCSR: Division ratio for watchdog timer count clock
   WTCNT: Initial counter value
   (The watchdog timer starts counting on the set clock.)
4. Set the GAIN0 bit to the desired value.
5. The LSI is internally stopped and the watchdog timer starts counting. The clock is supplied only to the watchdog timer and other internal clocks are stopped. In this state, the CKIO pin continues to output an unstable clock. To avoid malfunction due to the unstable clock, modify the CKOEN2 bit in FRQCR appropriately. Since this state is equivalent to the software standby mode state, some registers of on-chip peripheral modules are initialized. For details, see section 55, Register States.
6. When an overflow occurs on the watchdog timer, the specified clock supply is started and the LSI starts operation. The watchdog timer stops after an overflow.

52.4 Usage Notes

52.4.1 Usage Notes on Setting Registers

When writing to the registers related to power-down modes, note the following.
When writing to the register related to power-down modes, the CPU, after executing a write instruction, executes the next instruction without waiting for the write operation to complete.
Therefore, to reflect the change specified by writing to the register while the next instruction is executed, insert a dummy read of the same register between the register write instruction and the next instruction.

52.4.2 Usage Notes when the Realtime Clocks are Not to be Used

When the realtime clocks are not to be used, make the required settings for the clocks that are not to be used by following the procedure listed below.

1. Set 0 in the RCKSEL bits in RTC control register 4 (RCR4) for RTC0 and RTC1, as described in section 16.2.21.
2. Set 0 in the RTCEN bits in RTC control register 3 (RCR3) for RTC0 and RTC1, as described in section 16.2.20. Check that the bits have actually been updated before proceeding with the next step.
3. Set 0 in the RTC1XT and RTC0XT bits in the RTCXTAL selection register (RTCXTALSEL), as described in section 52.2.36.
4. Set 1 in the MSTP53 and MSTP52 bits in the standby control register 5 (STBCR5), as described in section 52.2.5.

52.4.3 Usage Notes when USB 2.0 Host/Function Modules are Not to be Used

When the USB 2.0 host/function modules are not to be used, release the modules that are not to be used from the power-on reset state or deep standby state and then set up the registers by following the procedure shown below.
Note that, if the device in use is one in the 176-pin BGA package, be sure to perform the following processing for the channel 1 USB 2.0 host/function module.

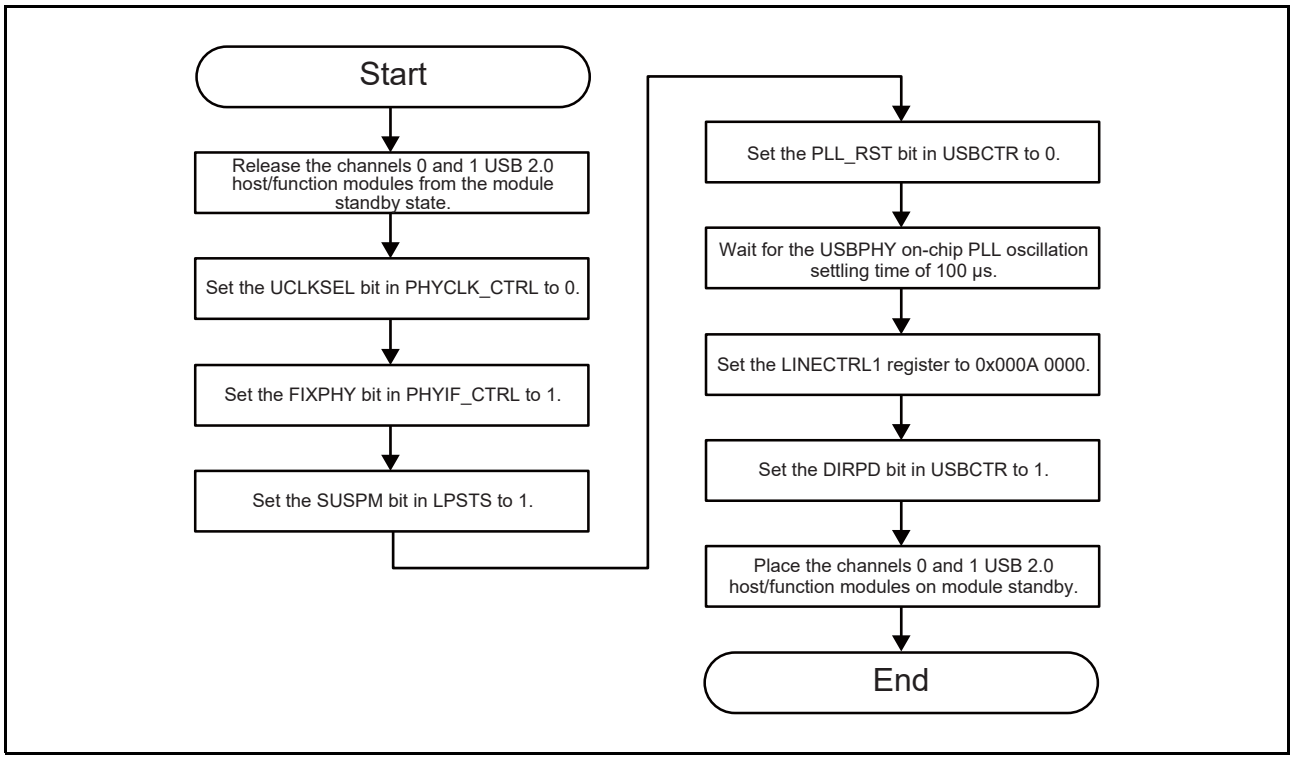


Figure 52.5 Flow of Settings when USB 2.0 Host/Function Modules are Not to be Used

For details on the registers, see section 32.2, Register Descriptions and section 33.2, Registers.

53. Debugger Interface

This LSI incorporates a debugger interface to support the boundary scan function and connection to the emulator.

53.1 Features

The debugger interface is a serial input/output interface which has a JTAG (Joint Test Action Group, IEEE Std.1149.1 and IEEE Standard Test Access Port and Boundary-Scan Architecture) interface and CoreSight debug interface.

This module has TAP controllers for the boundary scan and CoreSight debug function.

Driving the BSCANP pin high selects the boundary-scan TAP controller, and driving the BSCANP pin low selects the CoreSight debug TAP controller.

Figure 53.1 is a block diagram of this module and Table 53.1 shows the JTAG pin mode.

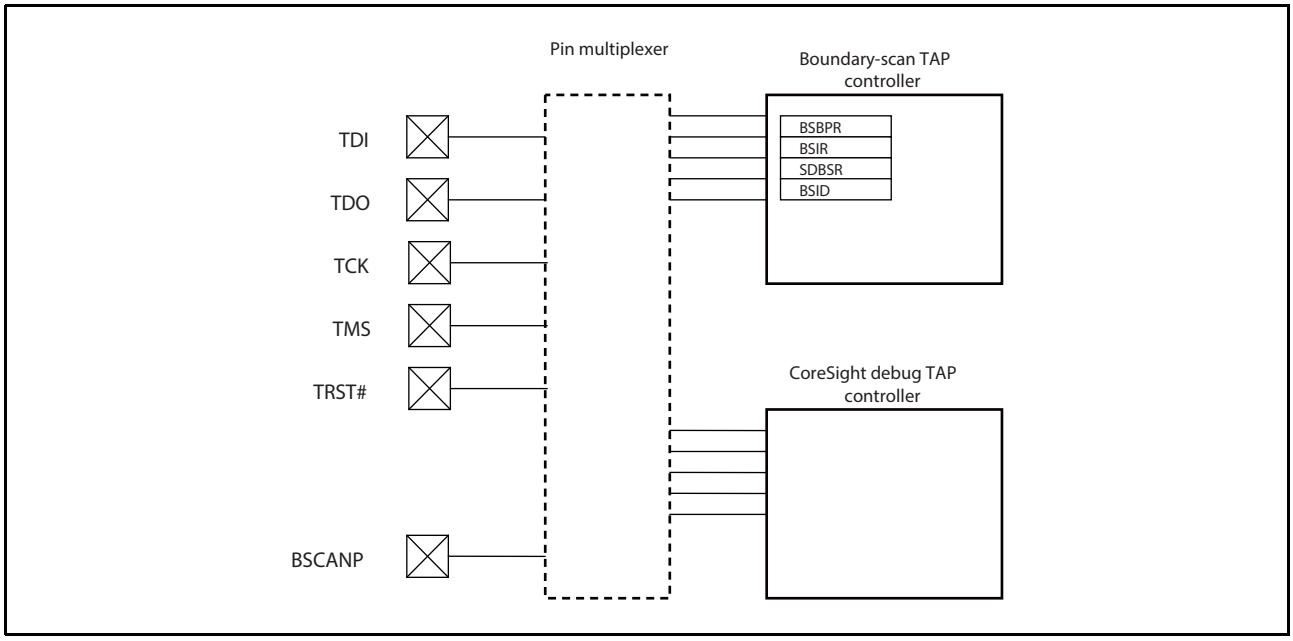


Figure 53.1 Block Diagram

Table 53.1 JTAG Pin Mode

BSCANP	JTAG Pin Mode
0	Normal operation (CoreSight debug mode)
1	Boundary-scan mode

The following lists the features of CoreSight.

- JTAG interface
  Supports JTAG and serial wire debug mode (SWD)
- Trace interface
  Outputs 4-bit × 264-Mbps (132 MHz DDR) trace data
  4-Kbyte Embedded Trace FIFO (ETF)
- SWV interface
  SW Trace output at 66 MHz (P1φ)
- Controls by ICE registers (Do not use these control other than for debugging purpose.)
- Control of resetting
  Disabling power shut-off in deep-standby mode during debugging (fake debug mode)

Figure 53.2 is a block diagram of CoreSight, Table 53.2 to Table 53.10 show the input/output pins of the cross-trigger interface (CTI), and Table 53.11 shows the address map for CoreSight. For details of CoreSight other than ICE registers, see the technical reference manual issued by Arm Ltd.

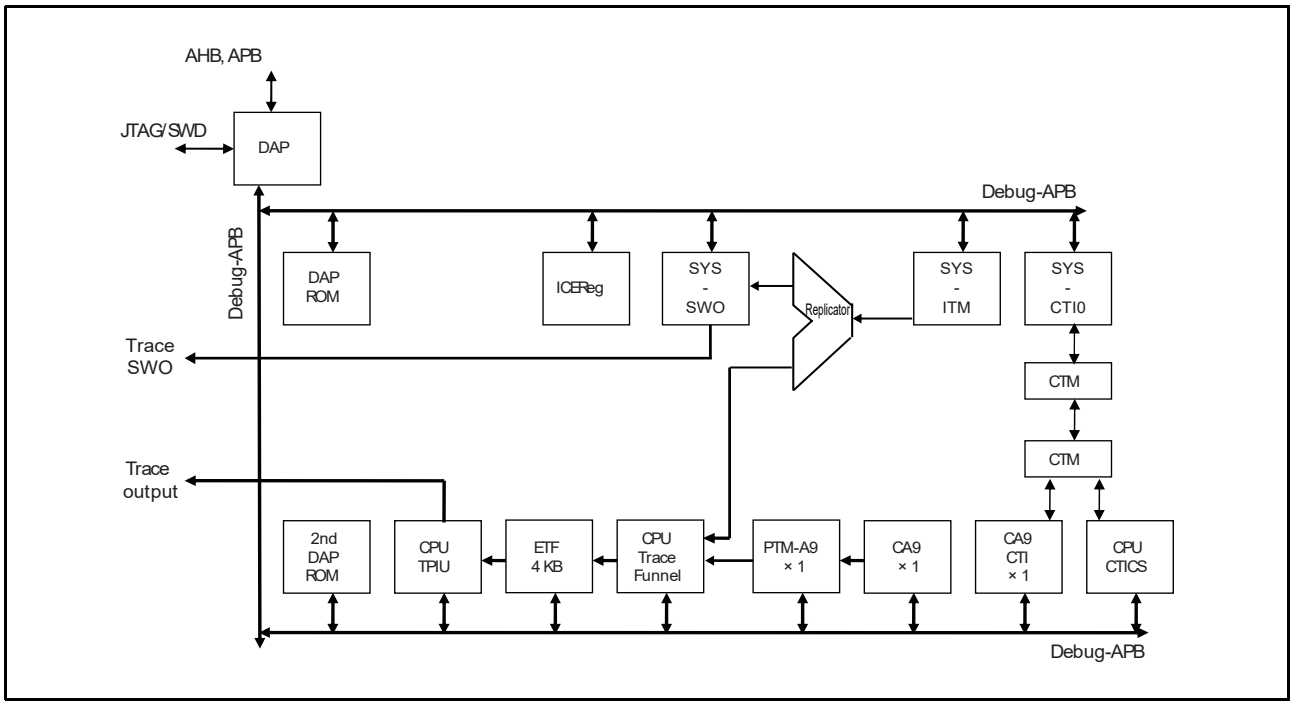


Figure 53.2 Block Diagram of CoreSight

Table 53.2 SYS CTI0 Trigger Input

Trigger Input Bit	Trigger Signal	Source Device
[7]	Not used	—
[6]	Not used	—
[5]	Not used	—
[4]	Not used	—
[3]	Not used	—
[2]	Not used	—
[1]	Not used	—
[0]	Not used	—

Table 53.3 SYS CTI0 Trigger Outputs

Trigger Output Bit	Trigger Signal	Source Device
[7]	Not used	—
[6]	Not used	—
[5]	Not used	—
[4]	TRIGOUT	—
[3]	Not used	SYS-ITM
[2]	Not used	—
[1]	Not used	—
[0]	Not used	—

Table 53.4 CA9 CTI Trigger Inputs

Trigger Input Bit	Trigger Signal	Source Device
[7]	Not used	—
[6]	TRIGGER	PTM-A9
[5]	COMMRX	CA9
[4]	COMMTX	CA9
[3]	EXTOUT[1]	PTM-A9
[2]	EXTOUT[0]	PTM-A9
[1]	PMUIRQ	CA9
[0]	DBGACK	CA9

Table 53.5 CA9 CTI Trigger Outputs

Trigger Output Bit	Trigger Signal	Destination Device
[7]	DBGRESTART	CA9
[6]	nCTIIRQ	INTC
[5]	Not used	—
[4]	EXTIN[3]	PTM-A9
[3]	EXTIN[2]	PTM-A9
[2]	EXTIN[1]	PTM-A9
[1]	EXTIN[0]	PTM-A9
[0]	EDBGRQ	CA9

Table 53.6 CPU CTICS Trigger Inputs

Trigger Input Bit	Trigger Signal	Source Device
[7]	Not used	—
[6]	Not used	—
[5]	Not used	—
[4]	Not used	—
[3]	ACQCOMP	ETF
[2]	FULL	ETF
[1]	Not used	—
[0]	Not used	—

Table 53.7 CPU CTICS Trigger Outputs

Trigger Output Bit	Trigger Signal	Destination Device
[7]	Not used	—
[6]	Not used	—
[5]	Not used	—
[4]	Not used	—
[3]	TRIGIN	CPU TPIU
[2]	FLUSHIN	CPU TPIU
[1]	TRIGIN	ETF
[0]	FLUSHIN	ETF

Table 53.8 CPU Trace Funnel

Port No.	Trace Source
[7]	reserved
[6]	reserved
[5]	reserved
[4]	SYS-ITM
[3]	reserved
[2]	reserved
[1]	reserved
[0]	reserved

Table 53.9 Address Map for CoreSight

System Address (Viewed from CPU)	Debug-APB Address (Viewed from Debugger)	Module
H'FC000000 to H'FC000FFF	H'80000000 to H'80000FFF	DAP ROM
H'FC001000 to H'FC001FFF	H'80001000 to H'80001FFF	Reserved
H'FC002000 to H'FC002FFF	H'80002000 to H'80002FFF	SYS-CTI0
H'FC003000 to H'FC003FFF	H'80003000 to H'80003FFF	Reserved
H'FC004000 to H'FC004FFF	H'80004000 to H'80004FFF	Reserved
H'FC005000 to H'FC005FFF	H'80005000 to H'80005FFF	SYS-ITM
H'FC006000 to H'FC006FFF	H'80006000 to H'80006FFF	SYS-SWO
H'FC007000 to H'FC007FFF	H'80007000 to H'80007FFF	Reserved
H'FC008000 to H'FC008FFF	H'80008000 to H'80008FFF	Reserved
H'FC009000 to H'FC009FFF	H'80009000 to H'80009FFF	Reserved
H'FC00A000 to H'FC00AFFF	H'8000A000 to H'8000AFFF	Reserved
H'FC00B000 to H'FC00BFFF	H'8000B000 to H'8000BFFF	Reserved
H'FC00C000 to H'FC00CFFF	H'8000C000 to H'8000CFFF	Reserved
H'FC00D000 to H'FC00DFFF	H'8000D000 to H'8000DFFF	Reserved
H'FC00E000 to H'FC00EFFF	H'8000E000 to H'8000EFFF	Reserved
H'FC00F000 to H'FC00FFFF	H'8000F000 to H'8000FFFF	ICE registers (ICEReg)
H'FC010000 to H'FC010FFF	H'80010000 to H'80010FFF	Reserved
H'FC011000 to H'FC011FFF	H'80011000 to H'80011FFF	Reserved
H'FC012000 to H'FC012FFF	H'80012000 to H'80012FFF	Reserved
H'FC013000 to H'FC013FFF	H'80013000 to H'80013FFF	Reserved
H'FC014000 to H'FC014FFF	H'80014000 to H'80014FFF	Reserved
H'FC015000 to H'FC015FFF	H'80015000 to H'80015FFF	Reserved
H'FC016000 to H'FC016FFF	H'80016000 to H'80016FFF	Reserved
H'FC017000 to H'FC017FFF	H'80017000 to H'80017FFF	Reserved
H'FC018000 to H'FC018FFF	H'80018000 to H'80018FFF	Reserved
H'FC019000 to H'FC019FFF	H'80019000 to H'80019FFF	Reserved
H'FC01A000 to H'FC01AFFF	H'8001A000 to H'8001AFFF	Reserved
H'FC01B000 to H'FC01BFFF	H'8001B000 to H'8001BFFF	Reserved
H'FC01C000 to H'FC01CFFF	H'8001C000 to H'8001CFFF	Reserved
H'FC01D000 to H'FC01DFFF	H'8001D000 to H'8001DFFF	Reserved
H'FC01E000 to H'FC01EFFF	H'8001E000 to H'8001EFFF	Reserved
H'FC01F000 to H'FC01FFFF	H'8001F000 to H'8001FFFF	Reserved
H'FC020000 to H'FC020FFF	H'80020000 to H'80020FFF	2nd DAP ROM
H'FC021000 to H'FC021FFF	H'80021000 to H'80021FFF	CPU-ETF
H'FC022000 to H'FC022FFF	H'80022000 to H'80022FFF	CPU-CTICS
H'FC023000 to H'FC023FFF	H'80023000 to H'80023FFF	CPU-TPIU
H'FC024000 to H'FC024FFF	H'80024000 to H'80024FFF	CPU-Trace Funnel
H'FC025000 to H'FC025FFF	H'80025000 to H'80025FFF	Reserved
H'FC026000 to H'FC026FFF	H'80026000 to H'80026FFF	Reserved
H'FC027000 to H'FC027FFF	H'80027000 to H'80027FFF	Reserved
H'FC028000 to H'FC028FFF	H'80028000 to H'80028FFF	Reserved
H'FC029000 to H'FC029FFF	H'80029000 to H'80029FFF	Reserved
H'FC02A000 to H'FC02AFFF	H'8002A000 to H'8002AFFF	Reserved
H'FC02B000 to H'FC02BFFF	H'8002B000 to H'8002BFFF	Reserved

Table 53.9 Address Map for CoreSight

System Address (Viewed from CPU)	Debug-APB Address (Viewed from Debugger)	Module
H'FC02C000 to H'FC02CFFF	H'8002C000 to H'8002CFFF	Reserved
H'FC02D000 to H'FC02DFFF	H'8002D000 to H'8002DFFF	Reserved
H'FC02E000 to H'FC02EFFF	H'8002E000 to H'8002EFFF	Reserved
H'FC02F000 to H'FC02FFFF	H'8002F000 to H'8002FFFF	Reserved
H'FC030000 to H'FC030FFF	H'80030000 to H'80030FFF	CA9-DBG (CPU0)
H'FC031000 to H'FC031FFF	H'80031000 to H'80031FFF	CA9-PMU (CPU0)
H'FC032000 to H'FC032FFF	H'80032000 to H'80032FFF	Reserved
H'FC033000 to H'FC033FFF	H'80033000 to H'80033FFF	Reserved
H'FC034000 to H'FC034FFF	H'80034000 to H'80034FFF	Reserved
H'FC035000 to H'FC035FFF	H'80035000 to H'80035FFF	Reserved
H'FC036000 to H'FC036FFF	H'80036000 to H'80036FFF	Reserved
H'FC037000 to H'FC037FFF	H'80037000 to H'80037FFF	Reserved
H'FC038000 to H'FC038FFF	H'80038000 to H'80038FFF	CA9 CTI (CPU0)
H'FC039000 to H'FC039FFF	H'80039000 to H'80039FFF	Reserved
H'FC03A000 to H'FC03AFFF	H'8003A000 to H'8003AFFF	Reserved
H'FC03B000 to H'FC03BFFF	H'8003B000 to H'8003BFFF	Reserved
H'FC03C000 to H'FC03CFFF	H'8003C000 to H'8003CFFF	PTM-A9 (CPU0)
H'FC03D000 to H'FC03DFFF	H'8003D000 to H'8003DFFF	Reserved
H'FC03E000 to H'FC03EFFF	H'8003E000 to H'8003EFFF	Reserved
H'FC03F000 to H'FC03FFFF	H'8003F000 to H'8003FFFF	Reserved

53.2 Input/Output Pins

Table 53.10 shows the pin configuration of the debugger interface.

Table 53.10 Pin Configuration

Name	Pin Name	Input/ Output	Function
Test clock	TCK/SWDCLK	Input	Data is input in serial to this module via the data input pin (TDI) or data is output via the data output pin (TDO) in synchronization with this signal. Functions as the SWDCLK pin in serial wire debug (SWD) mode.
Test mode select	TMS/SWDIO	Input, Input/ Output	Changing this signal in synchronization with the TCK signal determines the state of the TAP controller circuit. Its protocol conforms to the subset of the JTAG standard (IEEE standard 1149.1). Functions as the SWDIO pin in serial wire debug (SWD) mode.
Test reset	TRST#*1	Input	This signal is received asynchronously with the TCK signal. Asserting this signal resets this module. When power is supplied, this pin should be asserted for a given period regardless of whether or not this module function is used. For details on resetting, see section 53.5.2, Reset Signal Setting.
Test data input	TDI	Input	Data is sent to this module by changing this signal in synchronization with the TCK signal.
Test data output	TDO/SWO	Output	Data is read from this module by reading this signal in synchronization with the TCK signal. This pin also functions as an output pin in the serial wire debug (SWD) mode.
Boundary scan setting	BSCANP	Input	Input a high-level signal during boundary-scan testing. Input a low-level signal during normal use.
Clock output	TRACECLK	Output	Trace clock output pin
Enable output	TRACECTL	Output	Trace enable output pin
Data output	TRACEDATA3 to TRACEDATA0	Output	Trace data output pins

Note 1. When designing a board that can use an emulator, the RES# and TRST# pin circuits should be designed so that they can be set low and the TRST# pin can be controlled independently at power-on. When the TRST# pin is not in use, it should be either fixed to low level or connected to the RES# pin (or another pin which operates in the same manner as the RES# pin).

53.3 Registers for Boundary-Scan TAP Controller

The boundary-scan TAP controller has the following registers.

Table 53.11 List of Registers of Boundary-Scan TAP Controller

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Bypass register	BSBPR	—	—	—	—
Instruction register	BSIR	—	H'55	—	—
Boundary scan register	SDBSR	—	—	—	—
ID register	BSID	—	H'083B6447	—	—

53.3.1 Bypass Register (BSBPR)

BSBPR is a 1-bit register that cannot be accessed by the CPU. When BSIR is set to BYPASS mode, BSBPR is connected between the TDI and TDO pins. The initial value is undefined.

53.3.2 Instruction Register (BSIR)

BSIR is an 8-bit register. This register is initialized by TRST# assertion or in the TAP test-logic-reset state. BSIR cannot be accessed by the CPU.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	TI[7:0]	01010101	—	Test Instruction The instruction of this module is transferred to BSIR through a serial input from the TDI pin. For commands, see Table 53.12.

Table 53.12 Commands Supported for Boundary Scan TAP Controller

Bits 7 to 0								
TI7	TI6	TI5	TI4	TI3	TI2	TI1	TI0	Description
0	0	0	0	0	0	0	0	EXTEST
0	1	0	0	0	0	0	0	SAMPLE/PRELOAD
0	1	0	1	0	1	0	1	IDCODE (initial value)
1	1	0	1	0	0	0	0	CLAMP
1	0	0	0	0	0	0	0	HIGHZ
1	1	1	1	1	1	1	1	BYPASS
Other than above								Reserved

53.3.3 Boundary Scan Register (SDBSR)

SDBSR is a shift register, located on the PAD, for controlling the input/output pins of this LSI. SDBSR cannot be accessed by the CPU. The initial value is undefined. Using the EXTEST, SAMPLE/PRELOAD, CLAMP, and HIGHZ commands, a boundary scan test conforming to the JTAG standard can be carried out. Table 53.13 shows the correspondence between the pins of this LSI and bits of the boundary scan register.

Table 53.13 Correspondence between Pins of this LSI and Bits of Boundary Scan Registers

324-Pin Bit Number	272-/256-Pin Bit Number	176-Pin Bit Number	Pin Name*1	Type	324-Pin Bit Number	272-/256-Pin Bit Number	176-Pin Bit Number	Pin Name*1	Type
		From TDI			358	358	-	P8_0	OUTPUT
413	413	413	PD_0	OUTPUT*2	357	357	-	P8_0	CONTROL
412	412	412	PD_0	INPUT	356	356	-	P8_0	INPUT
411	411	411	PD_1	OUTPUT*2	355	-	-	P9_1	OUTPUT
410	410	410	PD_1	INPUT	354	-	-	P9_1	CONTROL
409	409	409	PD_2	OUTPUT*2	353	-	-	P9_1	INPUT
408	408	408	PD_2	INPUT	352	-	-	P9_0	OUTPUT
407	407	407	PD_3	OUTPUT*2	351	-	-	P9_0	CONTROL
406	406	406	PD_3	INPUT	350	-	-	P9_0	INPUT
405	405	405	PD_4	OUTPUT*2	349	-	-	P8_7	OUTPUT
404	404	404	PD_4	INPUT	348	-	-	P8_7	CONTROL
403	403	403	PD_5	OUTPUT*2	347	-	-	P8_7	INPUT
402	402	402	PD_5	INPUT	346	346	346	PE_4	OUTPUT
401	401	401	PD_6	OUTPUT*2	345	345	345	PE_4	CONTROL
400	400	400	PD_6	INPUT	344	344	344	PE_4	INPUT
399	399	399	PD_7	OUTPUT*2	343	343	-	PF_0	OUTPUT
398	398	398	PD_7	INPUT	342	342	-	PF_0	CONTROL
397	397	-	PB_0	OUTPUT	341	341	-	PF_0	INPUT
396	396	-	PB_0	CONTROL	340	340	-	PF_1	OUTPUT
395	395	-	PB_0	INPUT	339	339	-	PF_1	CONTROL
394	394	-	PA_7	OUTPUT	338	338	-	PF_1	INPUT
393	393	-	PA_7	CONTROL	337	337	-	PF_2	OUTPUT
392	392	-	PA_7	INPUT	336	336	-	PF_2	CONTROL
391	391	-	PA_6	OUTPUT	335	335	-	PF_2	INPUT
390	390	-	PA_6	CONTROL	334	-	-	P8_6	OUTPUT
389	389	-	PA_6	INPUT	333	-	-	P8_6	CONTROL
388	388	-	PA_5	OUTPUT	332	-	-	P8_6	INPUT
387	387	-	PA_5	CONTROL	331	-	-	P8_5	OUTPUT
386	386	-	PA_5	INPUT	330	-	-	P8_5	CONTROL
385	385	-	PA_4	OUTPUT	329	-	-	P8_5	INPUT
384	384	-	PA_4	CONTROL	328	-	-	P8_4	OUTPUT
383	383	-	PA_4	INPUT	327	-	-	P8_4	CONTROL
382	382	382	PE_0	OUTPUT	326	-	-	P8_4	INPUT
381	381	381	PE_0	CONTROL	325	-	-	P8_3	OUTPUT
380	380	380	PE_0	INPUT	324	-	-	P8_3	CONTROL
379	379	379	PE_1	OUTPUT	323	-	-	P8_3	INPUT
378	378	378	PE_1	CONTROL	322	322	322	PE_5	OUTPUT
377	377	377	PE_1	INPUT	321	321	321	PE_5	CONTROL
376	376	-	PA_3	OUTPUT	320	320	320	PE_5	INPUT
375	375	-	PA_3	CONTROL	319	319	319	PL_0	INPUT
374	374	-	PA_3	INPUT	318	318	318	PL_1	INPUT
373	373	-	PA_2	OUTPUT	317	317	317	PL_2	INPUT
372	372	-	PA_2	CONTROL	316	316	316	PL_3	INPUT
371	371	-	PA_2	INPUT	315	315	315	PL_4	INPUT
370	370	-	PA_1	OUTPUT	314	314	-	PF_3	OUTPUT
369	369	-	PA_1	CONTROL	313	313	-	PF_3	CONTROL
368	368	-	PA_1	INPUT	312	312	-	PF_3	INPUT
367	367	-	PA_0	OUTPUT	311	311	311	PE_6	OUTPUT
366	366	-	PA_0	CONTROL	310	310	310	PE_6	CONTROL
365	365	-	PA_0	INPUT	309	309	309	PE_6	INPUT
364	364	364	PE_2	OUTPUT	308	308	308	PH_0	OUTPUT
363	363	363	PE_2	CONTROL	307	307	307	PH_0	CONTROL
362	362	362	PE_2	INPUT	306	306	306	PH_0	INPUT
361	361	361	PE_3	OUTPUT	305	305	305	PH_1	OUTPUT
360	360	360	PE_3	CONTROL	304	304	304	PH_1	CONTROL
359	359	359	PE_3	INPUT	303	303	303	PH_1	INPUT

Table 53.13 Correspondence between Pins of this LSI and Bits of Boundary Scan Registers

324-Pin Bit Number	272-/256-Pin Bit Number	176-Pin Bit Number	Pin Name*1	Type	324-Pin Bit Number	272-/256-Pin Bit Number	176-Pin Bit Number	Pin Name*1	Type
302	302	-	PF_4	OUTPUT	235	-	-	P0_2	CONTROL
301	301	-	PF_4	CONTROL	234	-	-	P0_2	INPUT*4
300	300	-	PF_4	INPUT	233	-	-	P0_3	OUTPUT
299	299	299	PG_0	OUTPUT	232	-	-	P0_3	CONTROL
298	298	298	PG_0	CONTROL	231	-	-	P0_3	INPUT*4
297	297	297	PG_0	INPUT	230	-	-	P0_4	OUTPUT
296	296	296	PG_1	OUTPUT	229	-	-	P0_4	CONTROL
295	295	295	PG_1	CONTROL	228	-	-	P0_4	INPUT*4
294	294	294	PG_1	INPUT	227	-	-	P0_5	OUTPUT
293	293	293	PG_2	OUTPUT	226	-	-	P0_5	CONTROL
292	292	292	PG_2	CONTROL	225	-	-	P0_5	INPUT*4
291	291	291	PG_2	INPUT	224	-	-	P0_6	OUTPUT
290	290	-	PF_5	OUTPUT	223	-	-	P0_6	CONTROL
289	289	-	PF_5	CONTROL	222	-	-	P0_6	INPUT*4
288	288	-	PF_5	INPUT	221	221	221	PJ_1	OUTPUT
287	287	-	PF_6	OUTPUT	220	220	220	PJ_1	CONTROL
286	286	-	PF_6	CONTROL	219	219	219	PJ_1	INPUT
285	285	-	PF_6	INPUT	218	218	218	PJ_2	OUTPUT
284	284	284	PG_3	OUTPUT	217	217	217	PJ_2	CONTROL
283	283	283	PG_3	CONTROL	216	216	216	PJ_2	INPUT
282	282	282	PG_3	INPUT	215	215	215	PJ_3	OUTPUT
281	281	281	PG_4	OUTPUT	214	214	214	PJ_3	CONTROL
280	280	280	PG_4	CONTROL	213	213	213	PJ_3	INPUT
279	279	279	PG_4	INPUT	212	212	-	PK_5	OUTPUT
278	278	278	PG_5	OUTPUT	211	211	-	PK_5	CONTROL
277	277	277	PG_5	CONTROL	210	210	-	PK_5	INPUT
276	276	276	PG_5	INPUT	209	209	-	PH_5	OUTPUT
275	275	275	PG_6	OUTPUT	208	208	-	PH_5	CONTROL
274	274	274	PG_6	CONTROL	207	207	-	PH_5	INPUT
273	273	273	PG_6	INPUT	206	206	-	PH_6	OUTPUT
272	-	-	P8_2	OUTPUT	205	205	-	PH_6	CONTROL
271	-	-	P8_2	CONTROL	204	204	-	PH_6	INPUT
270	-	-	P8_2	INPUT	203	203	203	PJ_4	OUTPUT
269	-	-	P8_1	OUTPUT	202	202	202	PJ_4	CONTROL
268	-	-	P8_1	CONTROL	201	201	201	PJ_4	INPUT
267	-	-	P8_1	INPUT	200	200	200	PJ_5	OUTPUT
266	266	-	PH_2	OUTPUT	199	199	199	PJ_5	CONTROL
265	265	-	PH_2	CONTROL	198	198	198	PJ_5	INPUT
264	264	-	PH_2	INPUT	197	197	197	PK_0	OUTPUT
263	263	-	PF_7	OUTPUT	196	196	196	PK_0	CONTROL
262	262	-	PF_7	CONTROL	195	195	195	PK_0	INPUT
261	261	-	PF_7	INPUT	194	194	194	PK_1	OUTPUT
260	260	260	PG_7	OUTPUT	193	193	193	PK_1	CONTROL
259	259	259	PG_7	CONTROL	192	192	192	PK_1	INPUT
258	258	258	PG_7	INPUT	191	191	191	PK_2	OUTPUT
257	257	257	PH_3	OUTPUT	190	190	190	PK_2	CONTROL
256	256	256	PH_3	CONTROL	189	189	189	PK_2	INPUT
255	255	255	PH_3	INPUT	188	188	188	PK_3	OUTPUT
254	254	254	PH_4	OUTPUT	187	187	187	PK_3	CONTROL
253	253	253	PH_4	CONTROL	186	186	186	PK_3	INPUT
252	252	252	PH_4	INPUT	185	185	185	PK_4	OUTPUT
251	251	251	PJ_0	OUTPUT	184	184	184	PK_4	CONTROL
250	250	250	PJ_0	CONTROL	183	183	183	PK_4	INPUT
249	249	249	PJ_0	INPUT	182	182	182	P3_5	OUTPUT
248	248	-	PJ_6	OUTPUT	181	181	181	P3_5	CONTROL
247	247	-	PJ_6	CONTROL	180	180	180	P3_5	INPUT
246	246	-	PJ_6	INPUT	179	-	-	P1_0	OUTPUT
245	245	-	PJ_7	OUTPUT	178	-	-	P1_0	CONTROL
244	244	-	PJ_7	CONTROL	177	-	-	P1_0	INPUT*4
243	243	-	PJ_7	INPUT	176	-	-	P1_1	OUTPUT
242	-	-	P0_0	OUTPUT	175	-	-	P1_1	CONTROL
241	-	-	P0_0	CONTROL	174	-	-	P1_1	INPUT*4
240	-	-	P0_0	INPUT*4	173	173	173	P3_1	OUTPUT
239	-	-	P0_1	OUTPUT	172	172	172	P3_1	CONTROL
238	-	-	P0_1	CONTROL	171	171	171	P3_1	INPUT
237	-	-	P0_1	INPUT*4	170	170	170	P3_2	OUTPUT
236	-	-	P0_2	OUTPUT	169	169	169	P3_2	CONTROL

Table 53.13 Correspondence between Pins of this LSI and Bits of Boundary Scan Registers

324-Pin Bit Number	272-/256-Pin Bit Number	176-Pin Bit Number	Pin Name*1	Type	324-Pin Bit Number	272-/256-Pin Bit Number	176-Pin Bit Number	Pin Name*1	Type
168	168	168	P3_2	INPUT	101	101	-	PC_3	INPUT
167	167	167	P3_3	OUTPUT	100	100	-	PC_4	OUTPUT
166	166	166	P3_3	CONTROL	99	99	-	PC_4	CONTROL
165	165	165	P3_3	INPUT	98	98	-	PC_4	INPUT
164	164	164	P3_4	OUTPUT	97	97	-	PC_5	OUTPUT
163	163	163	P3_4	CONTROL	96	96	-	PC_5	CONTROL
162	162	162	P3_4	INPUT	95	95	-	PC_5	INPUT
161	-	-	P1_2	OUTPUT	94	94	-	PC_6	OUTPUT
160	-	-	P1_2	CONTROL	93	93	-	PC_6	CONTROL
159	-	-	P1_2	INPUT*4	92	92	-	PC_6	INPUT
158	-	-	P1_3	OUTPUT	91	91	-	PC_7	OUTPUT
157	-	-	P1_3	CONTROL	90	90	-	PC_7	CONTROL
156	-	-	P1_3	INPUT*4	89	89	-	PC_7	INPUT
155	-	-	P1_4	OUTPUT	88	88	88	P5_0	INPUT
154	-	-	P1_4	CONTROL	87	87	87	P5_1	INPUT
153	-	-	P1_4	INPUT*4	86	86	86	P5_2	INPUT
152	-	-	P2_0	OUTPUT	85	85	85	P5_3	INPUT
151	-	-	P2_0	CONTROL	84	84	84	P5_4	INPUT
150	-	-	P2_0	INPUT*4	83	83	83	P5_5	INPUT
149	149	-	P3_0	OUTPUT	82	82	82	P5_6	INPUT
148	148	-	P3_0	CONTROL	81	81	81	P5_7	INPUT
147	147	-	P3_0	INPUT	80	80	-	P6_4	OUTPUT
146	146	-	PC_0	OUTPUT	79	79	-	P6_4	CONTROL
145	145	-	PC_0	CONTROL	78	78	-	P6_4	INPUT
144	144	-	PC_0	INPUT	77	77	-	P6_5	OUTPUT
143	-	-	P2_1	OUTPUT	76	76	-	P6_5	CONTROL
142	-	-	P2_1	CONTROL	75	75	-	P6_5	INPUT
141	-	-	P2_1	INPUT*4	74	74	-	P6_6	OUTPUT
140	-	-	P2_2	OUTPUT	73	73	-	P6_6	CONTROL
139	-	-	P2_2	CONTROL	72	72	-	P6_6	INPUT
138	-	-	P2_2	INPUT*4	71	71	-	P6_7	OUTPUT
137	137	-	PC_1	OUTPUT	70	70	-	P6_7	CONTROL
136	136	-	PC_1	CONTROL	69	69	-	P6_7	INPUT
135	135	-	PC_1	INPUT	68	68	68	P6_0	OUTPUT
134	134	-	PC_2	OUTPUT	67	67	67	P6_0	CONTROL
133	133	-	PC_2	CONTROL	66	66	66	P6_0	INPUT
132	132	-	PC_2	INPUT	65	65	65	P6_1	OUTPUT
131	-	-	P2_3	OUTPUT	64	64	64	P6_1	CONTROL
130	-	-	P2_3	CONTROL	63	63	63	P6_1	INPUT
129	-	-	P2_3	INPUT*4	62	-	-	P7_0	OUTPUT
128	128	128	NMI	INPUT	61	-	-	P7_0	CONTROL
127	127	127	P4_0	OUTPUT	60	-	-	P7_0	INPUT
126	126	126	P4_0	CONTROL	59	-	-	P7_1	OUTPUT
125	125	125	P4_0	INPUT	58	-	-	P7_1	CONTROL
124	124	124	P4_1	OUTPUT	57	-	-	P7_1	INPUT
123	123	123	P4_1	CONTROL	56	56	-	P7_2	OUTPUT
122	122	122	P4_1	INPUT	55	55	-	P7_2	CONTROL
121	121	121	P4_2	OUTPUT	54	54	-	P7_2	INPUT
120	120	120	P4_2	CONTROL	53	-	-	P7_3	OUTPUT
119	119	119	P4_2	INPUT	52	-	-	P7_3	CONTROL
118	118	118	P4_3	OUTPUT	51	-	-	P7_3	INPUT
117	117	117	P4_3	CONTROL	50	-	-	P7_4	OUTPUT
116	116	116	P4_3	INPUT	49	-	-	P7_4	CONTROL
115	115	115	P4_4	OUTPUT	48	-	-	P7_4	INPUT
114	114	114	P4_4	CONTROL	47	47	47	P6_2	OUTPUT
113	113	113	P4_4	INPUT	46	46	46	P6_2	CONTROL
112	112	112	P4_5	OUTPUT	45	45	45	P6_2	INPUT
111	111	111	P4_5	CONTROL	44	44	44	P6_3	OUTPUT
110	110	110	P4_5	INPUT	43	43	43	P6_3	CONTROL
109	109	109	P4_6	OUTPUT	42	42	42	P6_3	INPUT
108	108	108	P4_6	CONTROL	41	-	-	P7_5	OUTPUT
107	107	107	P4_6	INPUT	40	-	-	P7_5	CONTROL
106	106	106	P4_7	OUTPUT	39	-	-	P7_5	INPUT
105	105	105	P4_7	CONTROL	38	-	-	P9_7	OUTPUT
104	104	104	P4_7	INPUT	37	-	-	P9_7	CONTROL
103	103	-	PC_3	OUTPUT	36	-	-	P9_7	INPUT
102	102	-	PC_3	CONTROL	35	-	-	P9_6	OUTPUT

Table 53.13 Correspondence between Pins of this LSI and Bits of Boundary Scan Registers

324-Pin Bit Number	272-/256-Pin Bit Number	176-Pin Bit Number	Pin Name*1	Type	324-Pin Bit Number	272-/256-Pin Bit Number	176-Pin Bit Number	Pin Name*1	Type
34	-	-	P9_6	CONTROL	16	-	-	P9_3	CONTROL
33	-	-	P9_6	INPUT	15	-	-	P9_3	INPUT
32	32	-	P7_6	OUTPUT	14	-	-	P9_2	OUTPUT
31	31	-	P7_6	CONTROL	13	-	-	P9_2	CONTROL
30	30	-	P7_6	INPUT	12	-	-	P9_2	INPUT
29	29	-	P7_7	OUTPUT	11	11	-	PB_4	OUTPUT
28	28	-	P7_7	CONTROL	10	10	-	PB_4	CONTROL
27	27	-	P7_7	INPUT	9	9	-	PB_4	INPUT
26	-	-	P9_5	OUTPUT	8	8	-	PB_3	OUTPUT
25	-	-	P9_5	CONTROL	7	7	-	PB_3	CONTROL
24	-	-	P9_5	INPUT	6	6	-	PB_3	INPUT
23	-	-	P9_4	OUTPUT	5	5	-	PB_2	OUTPUT
22	-	-	P9_4	CONTROL	4	4	-	PB_2	CONTROL
21	-	-	P9_4	INPUT	3	3	-	PB_2	INPUT
20	20	-	PB_5	OUTPUT	2	2	-	PB_1	OUTPUT
19	19	-	PB_5	CONTROL	1	1	-	PB_1	CONTROL
18	18	-	PB_5	INPUT	0	0	-	PB_1	INPUT
17	-	-	P9_3	OUTPUT					

Note 1. The pin names are as listed in the “Port function/dedicated function” column of Table 1.4 in section 1.6, List of Pins.
Note 2. Open-drain pin. Setting the data bit for the pin to 0 makes the pin output a low-level signal and setting the data bit for the pin to 1 places the pin in the hi-Z state.
Note 3. Bits of type “control” are active low. When the control bit is at the low level, the corresponding pin becomes an output.
Note 4. This pin is a TTL level input in the case of a boundary scan.

53.3.4 ID Register (BSID)

BSID is a 32-bit read-only register that cannot be accessed by the CPU. BSID can only be read from the pins when the IDCODE command is set. The device ID can be read from the CPU by access to the BSID register as described in section 5.11.1, BSID Register (BSID).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DID[31:16]															
Initial value:	0	0	0	0	1	0	0	0	0	0	1	1	1	0	1	1
R/W:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DID[15:0]															
Initial value:	0	1	1	0	0	1	0	0	0	1	0	0	0	1	1	1
R/W:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DID[31:0]	H'083B6447	—	Device ID This is the ID register stipulated by JTAG. Note that the higher-order four bits may be changed depending on the version of the chip.

53.4 ICE Registers

The debugger interface has the following ICE registers.
ICE registers are exclusively for use in debugging. Do not use these registers other than for debugging.

Table 53.14 Configuration of ICE Registers

Register Name	Abbreviation	R/W	Initial Value	Address		Access Size
				Viewed from CPU	Viewed from Debugger	
Mode reset control register	ICEREGMDRSTCTL*1	R/W	H'0000111E	H'FC00F000	H'8000F000	32
JTAG trace select register	ICEREGJTTRCSEL*1	R/W	H'00000000	H'FC00F004	H'8000F004	32
Clock power control register	ICEREGCLKPWRCTRL*1	R/W	H'00000000	H'FC00F014	H'8000F014	32
Lock access register	ICEREGLOCKACCES	W	—	H'FC00FFB0	H'8000FFB0	32

Note 1. When the CPU is to write a value to these registers, set the ICEREGLOCKACCESS register to "H'C5ACCE55" beforehand.

53.4.1 Mode Reset Control Register (ICEREGMDRSTCTL)

ICEREGMDRSTCTL is used to set the debug mode and to perform software reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RSTRB_CPU0_DERSTZ	—	—	—	RSTRB_CPU0_CPURSTZ	—	—	—	—	—	RSTRB_CPU_PRSTDBGZ	RSTRB_CPU_SYSRSTZ	—
Initial value:	0	0	0	1	0	0	0	1	0	0	0	1	1	1	1	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved The read value is always 0.
12	RSTRB_CPU0_DERSTZ	1	R/W	NEON Reset (low-active) Resets the NEON unit in the CPU.
11 to 9	—	All 0	R	Reserved The read value is always 0.
8	RSTRB_CPU0_CPURSTZ	1	R/W	CPU Reset (low-active) Resets all systems on the CPU block other than debugging resources.
7 to 5	—	All 0	R	Reserved The read value is always 0.
4	—	1	R/W	Reserved The read value is always 1. The write value should always be 1.
3	—	1	R/W	Reserved The read value is always 1. The write value should always be 1.
2	RSTRB_CPU_PRSTDBGZ	1	R/W	CPU Subsystem Debug Peripheral Reset (low-active) Resets the debug-APB block in the CPU subsystem.
1	RSTRB_CPU_SYSRSTZ	1	R/W	CPU Subsystem Reset (low-active) Resets the CPU subsystem other than those for debugging.
0	—	0	R	Reserved The read value is always 0.

53.4.2 JTAG Trace Select Register (ICEREGJTTRCSEL)

ICEREGJTTRCSEL is used to control trace data output to the respective pins and debug enable signal supplied to the CPU.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TRCCK_DIV_SEL[1:0]		—	—	—	—	TRCMUX_SEL	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R	R	R	R	R/W	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved The read value is always 0.
26, 25	TRCCK_DIV_SEL [1:0]	0	R/W	Sets the divider ratio of the trace clock output. 00: Trace clock /4 (33-MHz DDR) 01: Trace clock /2 (66-MHz DDR) 10: Trace clock (132-MHz DDR) 11: Setting prohibited.
24 to 21	—	All 0	R	Reserved The read value is always 0.
20	TRCMUX_ SEL	0	R/W	Sets the priority of the function multiplexed on the pins to which trace output is assigned. 0: The function of the pins depends on the general I/O port setting. 1: The pins act as trace outputs regardless of the general I/O port setting.
19 to 0	—	All 0	R	Reserved The read value is always 0.

53.4.3 Clock Power Control Register (ICEREGCLKPWRCTRL)

ICEREGCLKPWRCTRL is used to set the fake debug mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	FAKEDBG CTRL	—	—	—	—	—	—	FAKE DBG
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved The read value is always 0.
7	FAKEDBGCTRL	0	R/W	Enables or disables the fake debug mode*1. 0: The power control is determined by the standby control register 1 (STBCR1). 1: The power control is determined by this register.
6 to 1	—	All 0	R	Reserved The read value is always 0.
0	FAKEDBG	0	R/W	Selects whether or not to stop the power supply in deep standby mode. 0: The power supply is actually stopped in deep standby mode. 1: The power supply is not stopped even in deep standby mode.

Note 1. In fake debug mode, stopping the power supply to any module, including the debugger interface, is disabled; the supply of power to the modules is continued. Using fake debug mode can be helpful in the debugging of software for a system which is released from the deep standby mode after a transition to deep standby mode. Do not use fake debug mode for purposes other than debugging. To use fake debug mode, set the FAKEDBGCTRL and FAKEDBG bits to 1.

53.4.4 Lock Access Register (ICEREGLOCKACCESS)

ICEREGLOCKACCESS is used to enable access to ICE registers by the CPU.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ICEREGLOCKACCESS															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ICEREGLOCKACCESS															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ICEREGLOCK ACCESS	—	W	To enable access to ICE registers by the CPU, first write H'C5ACCE55 to this register.

53.5 Operation

53.5.1 TAP Controller

Figure 53.3 shows the internal states of the TAP controller. State transitions basically conform to the JTAG standard.

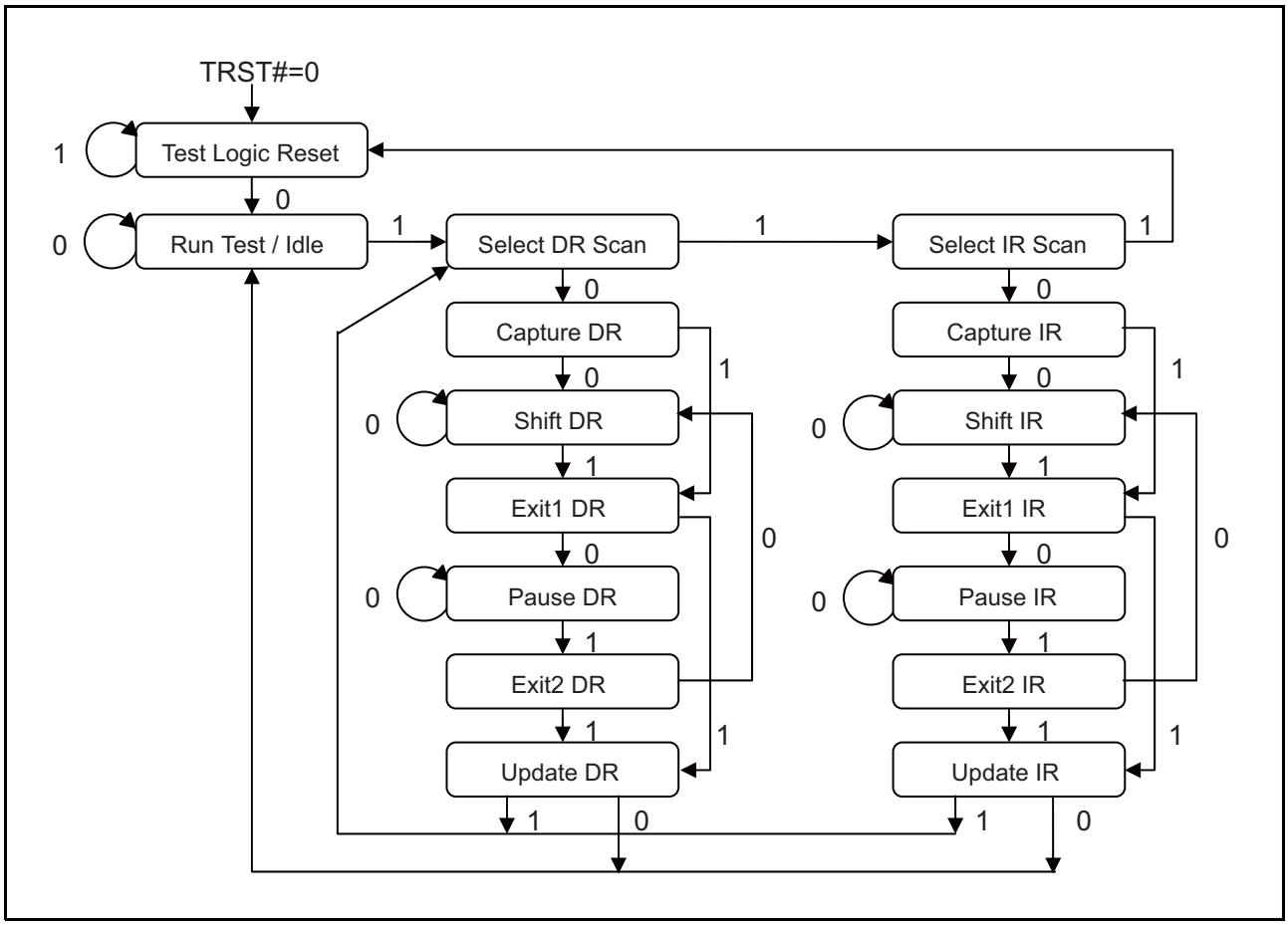


Figure 53.3 State Transitions of TAP Controller

Note: State transition occur according to the TMS value at the rising edge of the TCK signal. The TDI value sampled at the rising edge of the TCK signal and shifted at the falling edge of the TCK signal. TDO value is output at the rising edge of the TCK signal. The TDO signal is in a Hi-Z state for TAP controller states other than Shift-DR and Shift-IR. A transition to the Test-Logic-Reset state by clearing TRST# to 0 is performed asynchronously with the TCK signal.

53.5.2 Reset Signal Setting

Table 53.15 Reset Signal Setting

RES#	TRST#	Chip State
L	L	The chip is power-on reset and this module is reset.*1
	H	The chip is power-on reset.*1
H	L	This module is only reset.
	H	Normal operation

Note 1. By asserting the RES# and TRST# signals, the CPU and CoreSight enter the reset state. Set the debugging function while asserting the RES# signal after negating the TRST# signal.

53.6 Boundary Scan

Commands can be set in BSIR by this module to place this module's pins in boundary scan mode stipulated by JTAG.

53.6.1 Supported Instructions

This LSI supports three mandatory instructions stipulated by JTAG standard (BYPASS, SAMPLE/PRELOAD, and EXTEST) and three option instructions (IDCODE, CLAMP, and HIGHZ).

(1) BYPASS

BYPASS is a mandatory instruction that operates the bypass register. This instruction is used to shorten the shift path to speed up serial data transfer involving other LSI on the printed circuit board. This LSI's system circuits are not affected by execution of this instruction.

(2) SAMPLE/PRELOAD

SAMPLE/PRELOAD is used to input data from this LSI’s internal circuit to the boundary scan register, outputs data from the scan path, and loads data onto the scan path. While this instruction is being executed, signals input to this LSI pins are transmitted directly to the internal circuit, and the values of the internal circuit are directly output externally from the output pins. This LSI's system circuits are not affected by execution of this instruction.

In a SAMPLE operation, a snapshot of a value to be transferred from an input pin to the internal circuit, or a value to be transferred from the internal circuit to an output pin, is latched into the boundary scan register and read from the scan path. Snapshot latching is performed in synchronization with the rising edge of the TCK signal in the Capture-DR state. Snapshot latching does not affect normal operation of this LSI.

In a PRELOAD operation, an initial value is set in the parallel output latch of the boundary scan register from the scan path prior to the EXTEST instruction. Without a PRELOAD operation, when the EXTEST instruction was executed, an undefined value would be output from the output pin until completion of the initial scan sequence (transfer to the output latch) (with the EXTEST instruction, the parallel output latch value is constantly output to the output pin).

(3) EXTEST

This instruction is provided to test external circuit when this LSI is mounted on a printed circuit board. When this instruction is executed, output pins are used to output test data (previously set by the SAMPLE/PRELOAD instruction) from the boundary scan register to the printed circuit board, and input pins are used to latch test results into the boundary scan register from the printed circuit board. If testing is carried out by using the EXTEST instruction N times, the Nth test data is scanned-in when test data (N-1) is scanned out. The data loaded into the output pin boundary scan register in the Capture-DR state are not used for external circuit testing (data are replaced by the shift operation).

(4) IDCODE

Commands can be set in BSIR via pins of this module to place the module’s pins in the IDCODE mode stipulated by JTAG. When this module is initialized (TRST# is asserted or TAP is in the Test-Logic-Reset state), the IDCODE mode is entered.

(5) CLAMP, HIGHZ

Commands can be set in BSIR via pins of this module to place the module’s pins in the CLAMP or HIGHZ mode stipulated by JTAG.

53.6.2 Points for Attention

Boundary scan mode does not cover the following signals.

Category / Module	Clock	Reset, Interrupt	Debugger interface	USB 2.0 Host function module
Pin Name	AUDIO_X1	RES#	TRST#	DP0
	AUDIO_X2		JP0_1	DM0
	USB_X1		JP0_0	RREF0
	USB_X2		TMS	DP1
	XTAL		TCK	DM1
	EXTAL		BSCANP	RREF1
	CKIO			
	RTC_X1			
	RTC_X2			

Category / Module	HyperBus controller Octa Memory controller	SPI Multi I/O bus controller	SD/MMC Host interface	MIPI CSI-2 Interface
Pin Name	HM_CK#	QSPI0_IO0	SD0_CLK	CSI_CLKP
	HM_CK / OM_SCLK	QSPI0_IO1	SD0_CMD	CSI_CLKN
	HM_CS0# / OM_CS0#	QSPI0_IO2	SD0_DAT0	CSI_DATA0P
	HM_CS1# / OM_CS1#	QSPI0_IO3	SD0_DAT1	CSI_DATA0N
	HM_DQ0 / OM_SIO0	QSPI0_SPCLK	SD0_DAT2	CSI_DATA1P
	HM_DQ1 / OM_SIO1	QSPI0_SSL	SD0_DAT3	CSI_DATA1N
	HM_DQ2 / OM_SIO2	QSPI1_IO0	SD0_DAT4	
	HM_DQ3 / OM_SIO3	QSPI1_IO1	SD0_DAT5	
	HM_DQ4 / OM_SIO4	QSPI1_IO2	SD0_DAT6	
	HM_DQ5 / OM_SIO5	QSPI1_IO3	SD0_DAT7	
	HM_DQ6 / OM_SIO6	QSPI1_SPCLK	SD0_RST#	
	HM_DQ7 / OM_SIO7	QSPI1_SSL	SD1_CLK	
	HM_RESET# / OM_RESET#	RPC_INT#	SD1_CMD	
	HM_RWDS / OM_DQS	RPC_RESET#	SD1_DAT0	
		RPC_WP#	SD1_DAT1	
			SD1_DAT2	
			SD1_DAT3	

53.7 Usage Notes

1. Once a command of this module is set, it will not be modified unless another command is reissued. If the same command is given continuously, the command must be set after a command (BYPASS, etc.) that does not affect LSI operations is once set.
2. Regardless of whether or not this module is to be used, it should be initialized by asserting the TRST# signal when power is supplied and on release from deep standby mode (excluding fake debug mode) through assertion of the RES# signal.
3. Be sure to wait for at least 200 ns after negating the TRST# signal before starting TAP controller operation.
4. Be sure to fix the TMS pin to the high level until 200 ns have elapsed after negation of the TRST# signal.

54. Trusted Secure IP

This LSI incorporates a Trusted Secure IP module to provide security functions. The module consists of an access management circuit, encryption engine, and random number generator. In combination with the Trusted Secure IP driver, the Trusted Secure IP can prevent eavesdropping (confidentiality), falsification of information (integrity), and impersonation (authenticity).
Key information to be used in encrypting and decrypting data is only stored within the Trusted Secure IP, and any external access can be shut out to obtain a system with strong security.

54.1 Overview

Table 54.1 summarizes the specifications of the Trusted Secure IP. Figure 54.1 shows a block diagram of the Trusted Secure IP.

Table 54.1 Specifications of Trusted Secure IP (1/2)

Item	Description
Access control	Access management circuit • In case of irregular access to the Trusted Secure IP due to a falsified program or runaway execution of a program, this circuit blocks all subsequent access and stops the output of data from the Trusted Secure IP.
Encryption engine	AES: Compliant with NIST FIPS PUB 197 algorithm • Key sizes: 128, 192, or 256 bits • Block sizes: 128 bits • Block cipher mode of operation ECB, CBC, CTR: Compliant with NIST SP 800-38A CMAC: Compliant with NIST SP 800-38B CCM: Compliant with NIST SP 800-38C GCM: Compliant with NIST SP 800-38D XTS: Compliant with NIST SP 800-38E GCTR • Number of cycles for execution*1 ECB, CBC, CTR, CMAC, GCTR, XTS: 11 cycles of Bφ for 128-bit keys, 13 cycles of Bφ for 192-bit keys, 15 cycles of Bφ for 256-bit keys CCM: 22 cycles of Bφ for 128-bit keys, 26 cycles of Bφ for 192-bit keys, 30 cycles of Bφ for 256-bit keys AES-GCM • AES-GCM is realized by combining AES-GCTR and GHASH. RSA • Key sizes: Up to 2048 bits • Block sizes: Up to 2048 bits • Number of cycles for execution: Approximately 1,300,000 cycles of Bφ when the CRT is used*1 TDES • Key sizes: 56 bits, 2 × 56 bits, or 3 × 56 bits • Block sizes: 64 bits • Block cipher mode of operation: ECB, CBC • Number of cycles for execution*1 16 cycles of Bφ for 56-bit keys, 32 cycles of Bφ for 2 × 56-bit keys, 48 cycles of Bφ for 3 × 56-bit keys ARC4 • Key sizes: 2048 bits • Block sizes: 64 bits • Number of cycles for execution: 16 cycles of Bφ*1 HASH Support for SHA1, SHA224/SHA256/MD5, GHASH • Block sizes: 512bits • Number of cycles for execution*1 SHA1: 80 cycles of Bφ SHA224/SHA256/MD5: 64 cycles of Bφ GHASH: 9 cycles of Bφ

Table 54.1 Specifications of Trusted Secure IP (2/2)

Item	Description
Encryption engine	Key management • Keys are only valid within the Trusted Secure IP. • Only key generation information is output from the Trusted Secure IP. • Keys can be regenerated by the input of key generation information to the Trusted Secure IP. Endian • Big or little
Generation of random numbers	32-bit true random number generator • The Trusted Secure IP driver can assemble 32-bit true random numbers to generate 128- or 256-bit true random numbers. • The generated 128-bit and 256-bit true random numbers are used as keys in encrypting and decrypting data.
Unique ID	• An ID unique to the LSI (unique ID) is accessible from the access management circuit through the dedicated bus. • Combining the unique ID with the key generation information prevents the illicit copying of data to another LSI.
Interrupt sources	10
Low power consumption	Setting of the module stop state is possible.

Note 1. This does not include the overhead for calling functions of the Trusted Secure IP driver.

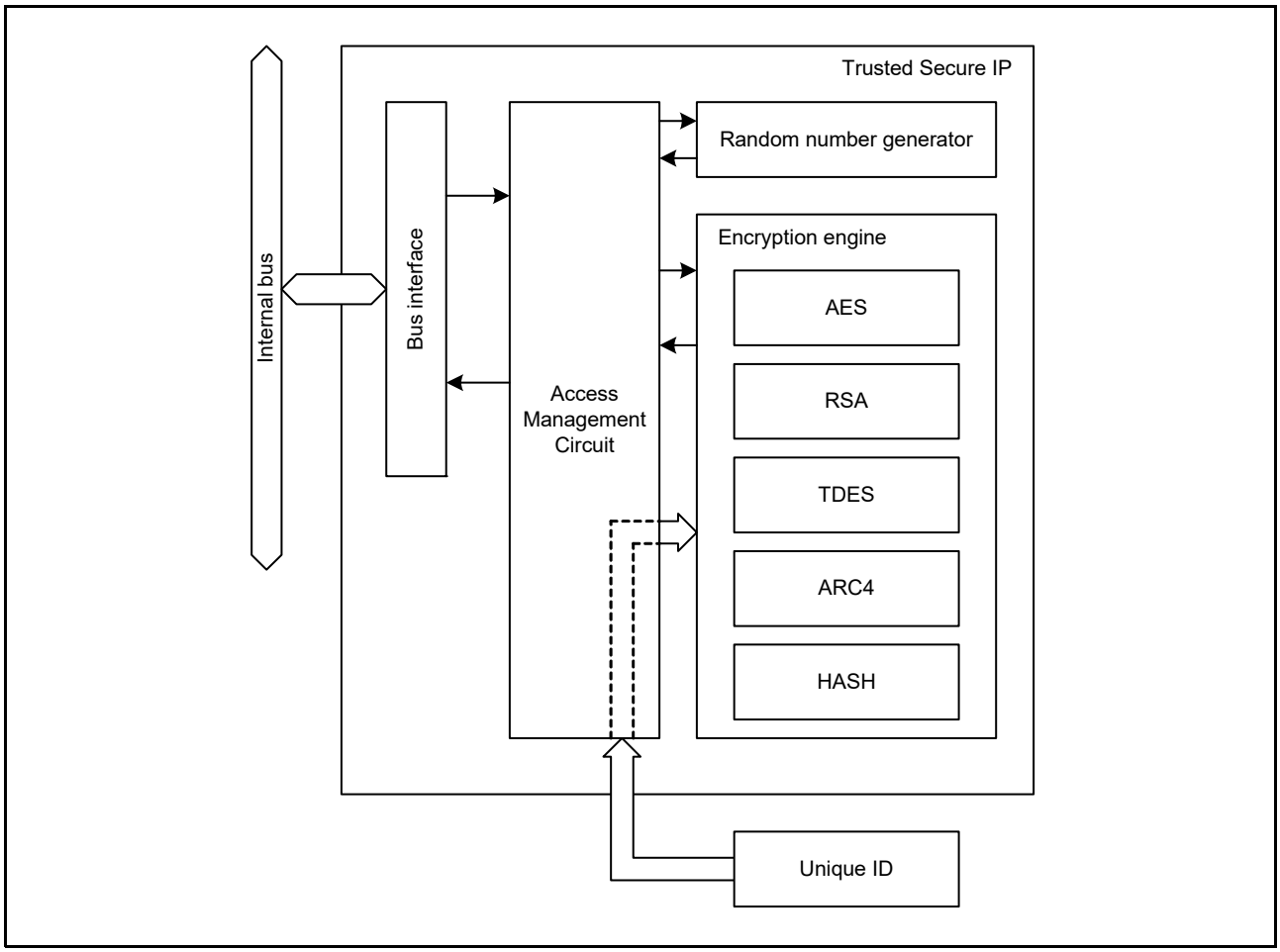


Figure 54.1 Trusted Secure IP Block Diagram

54.2 Operation

54.2.1 Operating Modes and State Transitions

Figure 54.2 shows the state transitions of the Trusted Secure IP.
Use of the Trusted Secure IP security functions is only possible through use of the Trusted Secure IP driver provided by Renesas Electronics, in accordance with the state transitions as shown in the figure below.
When irregular access to the Trusted Secure IP (access that violates the defined procedure) due to a falsified program or a program entering runaway execution, etc. is attempted, the access management circuit does not accept any subsequent access and stops the output of any data from the Trusted Secure IP.

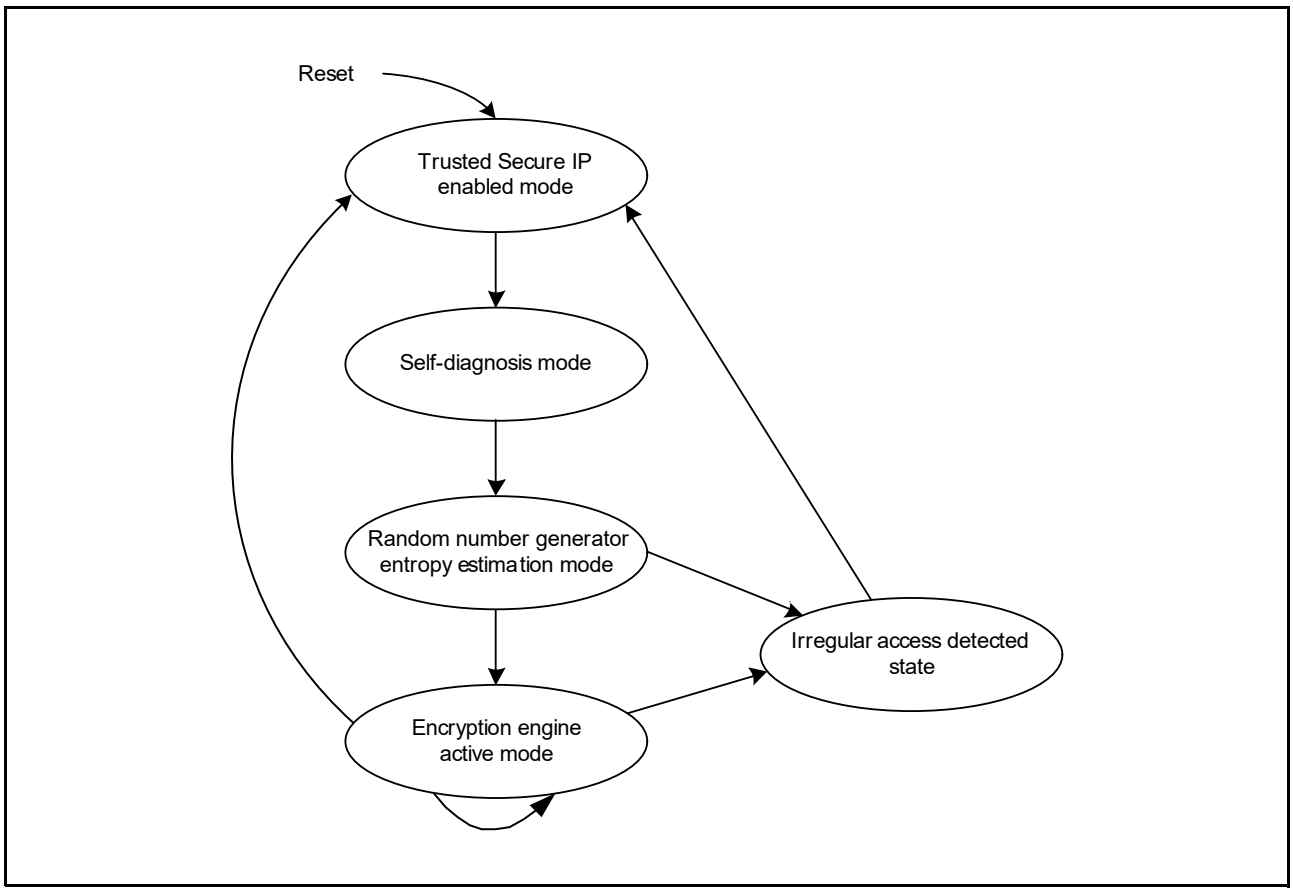


Figure 54.2 Trusted Secure IP Operating Modes and State Transitions

Many of the security functions that the Trusted Secure IP offers are applicable only in the encryption engine active mode. The operations that can be performed in this mode are given below.

(1) Key Installation
(2) Encryption and decryption
(3) Key generation
(4) Random number generation

54.2.2 Encryption Engine

Figure 54.3 shows processes of the encryption engine integrated in the Trusted Secure IP.
The encryption engine, using the key generation information, performs plaintext to ciphertext encryption and ciphertext to plaintext decryption by hardware.
In no part of the encryption or decryption process, is key data or intermediate data ever exposed outside of the Trusted Secure IP.

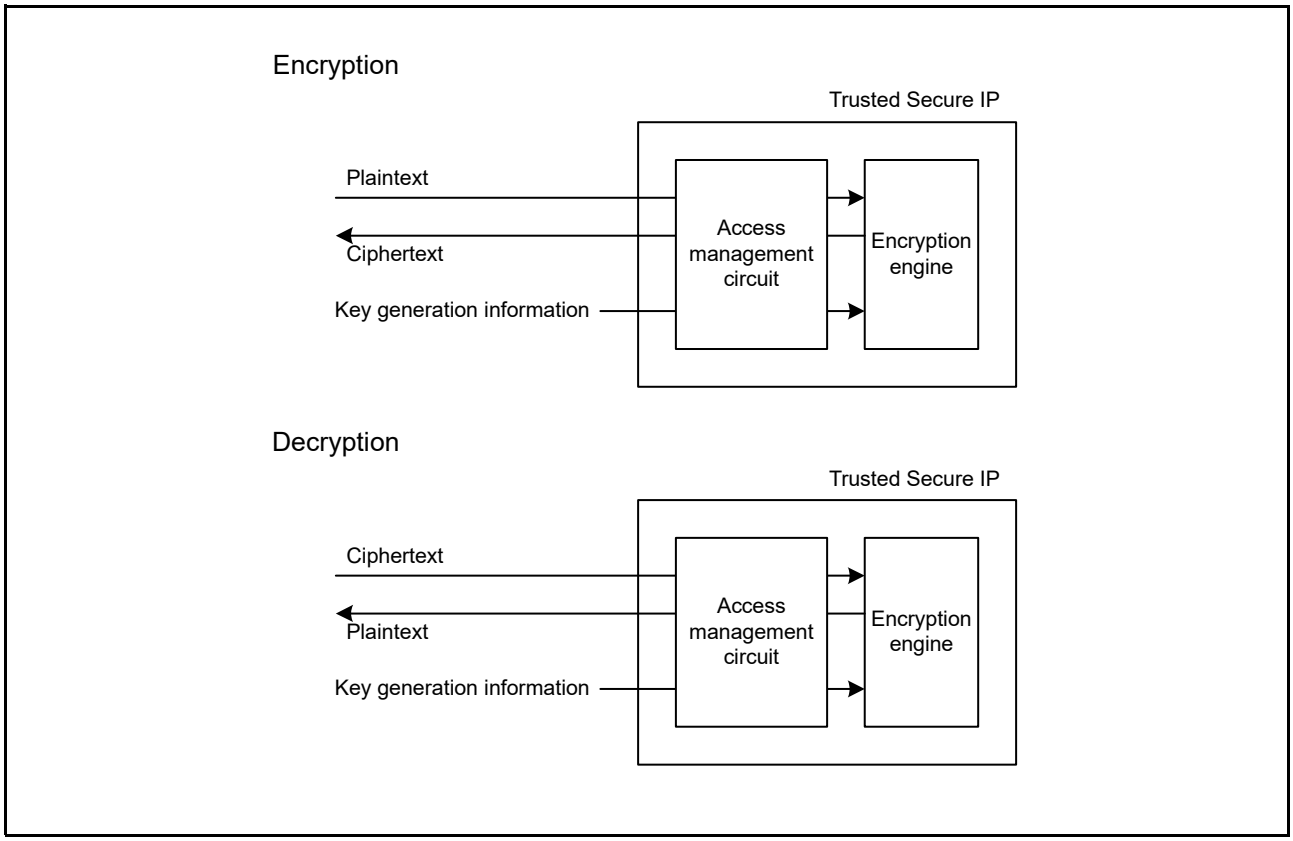


Figure 54.3 Encryption and Decryption processes by Encryption Engine

54.2.3 Key Installation

The key installation is the operation that safely converts the user key to the key generation information and stores it in flash memory. The procedure for installing the key data are given below.

(1) The user uses the key (Key-2) used for encrypting the user key to encrypt the user key (Key-1) producing eKey-1.
(2) The user sends the encrypted user key (eKey-1) to the Trusted Secure IP over the serial interface.
(3) The key generation information of the Key-2 (Index-2) contained in the Trusted Secure IP driver is used to recover the Key-2, which is then used to decrypt the user key.
(4) The user key is converted to user key generation information (Index-1) using the unique ID and a random number, and stored in flash memory.

The installation process and flow chart are given in Figure 54.4 and Figure 54.5, respectively.
Once the key data is installed, the user key generation information (Index-1) can then be used to perform encryption or decryption.

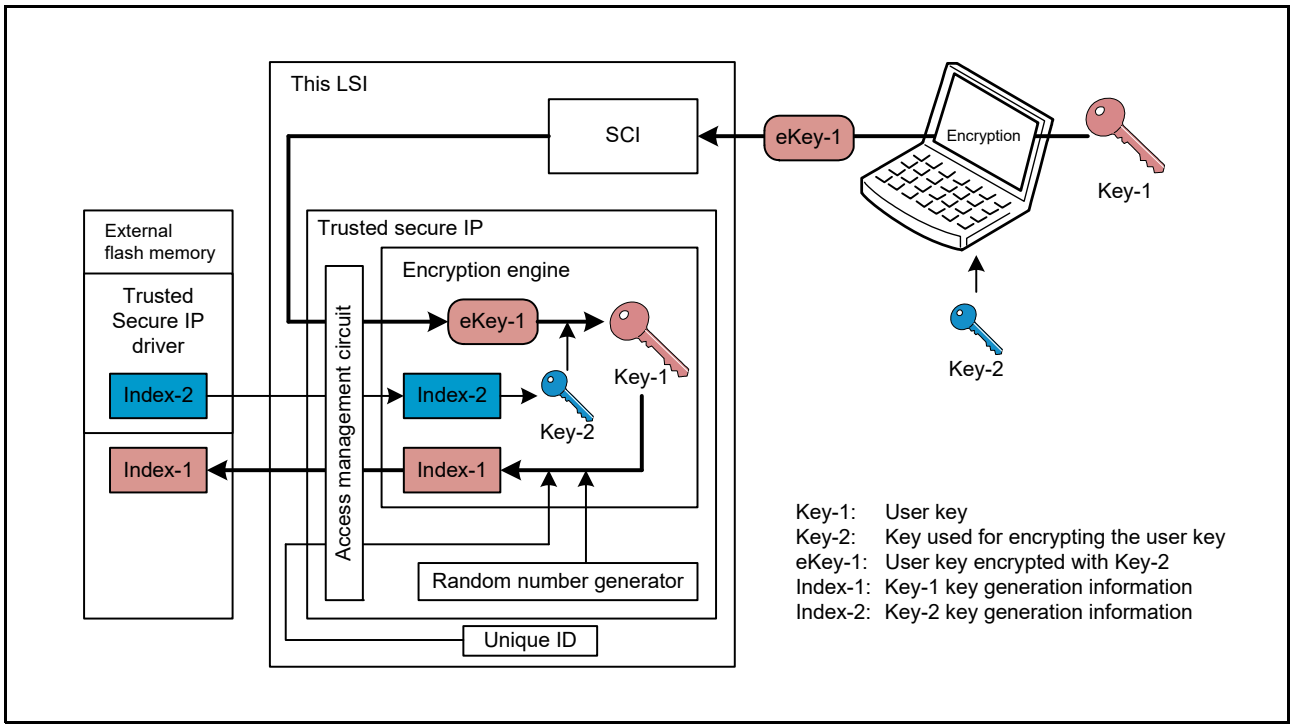


Figure 54.4 Key Installation Process

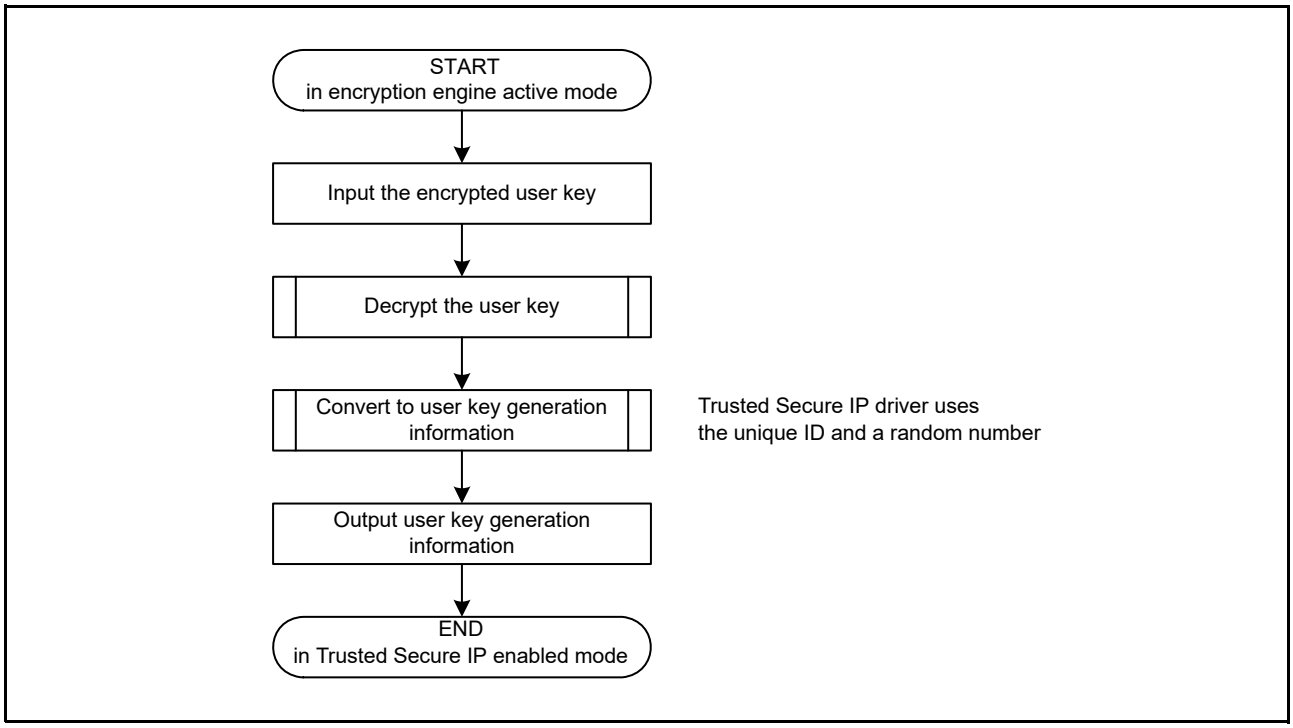


Figure 54.5 Key Installation Flow Chart

54.2.4 Encryption and Decryption

The procedures for encrypting and decrypting data are given below.

(1) Input the key generation information into the Trusted Secure IP, and recover the key data.
(2) Input the data to encrypt or decrypt into the Trusted Secure IP. This converts plaintext into ciphertext, and ciphertext into plaintext.
(3) Read the converted data.

The encryption engine has an input buffer and an output buffer, enabling encryption/decryption to proceed in parallel with data input/output.
Figure 54.6, Figure 54.7, and Figure 54.8 show the timing diagram, encryption flow, and decryption flow, respectively.

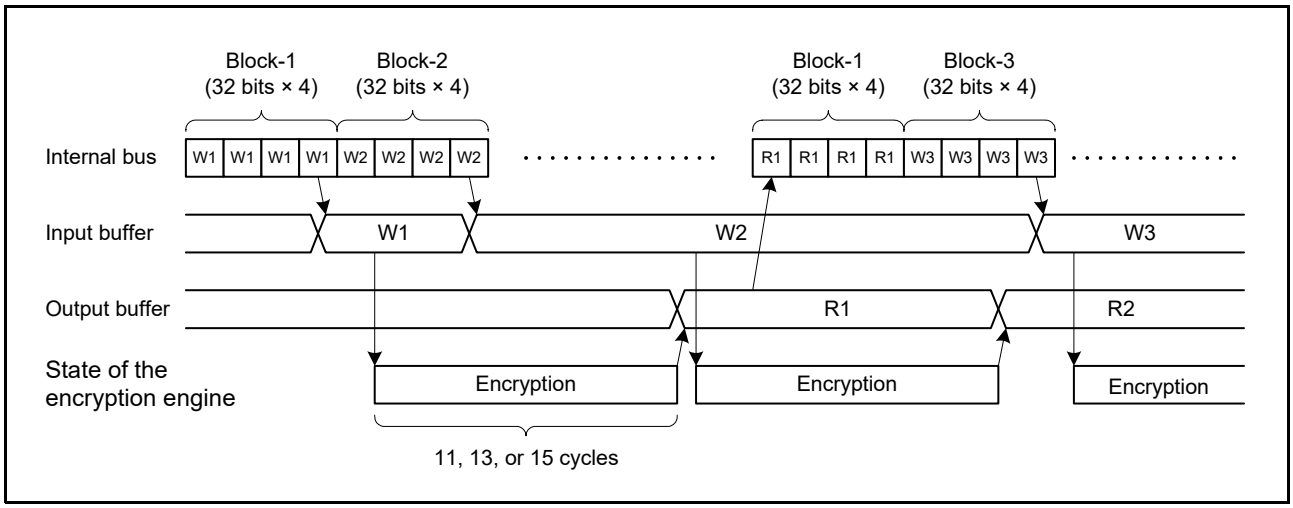


Figure 54.6 Encryption and Decryption Timing Diagram

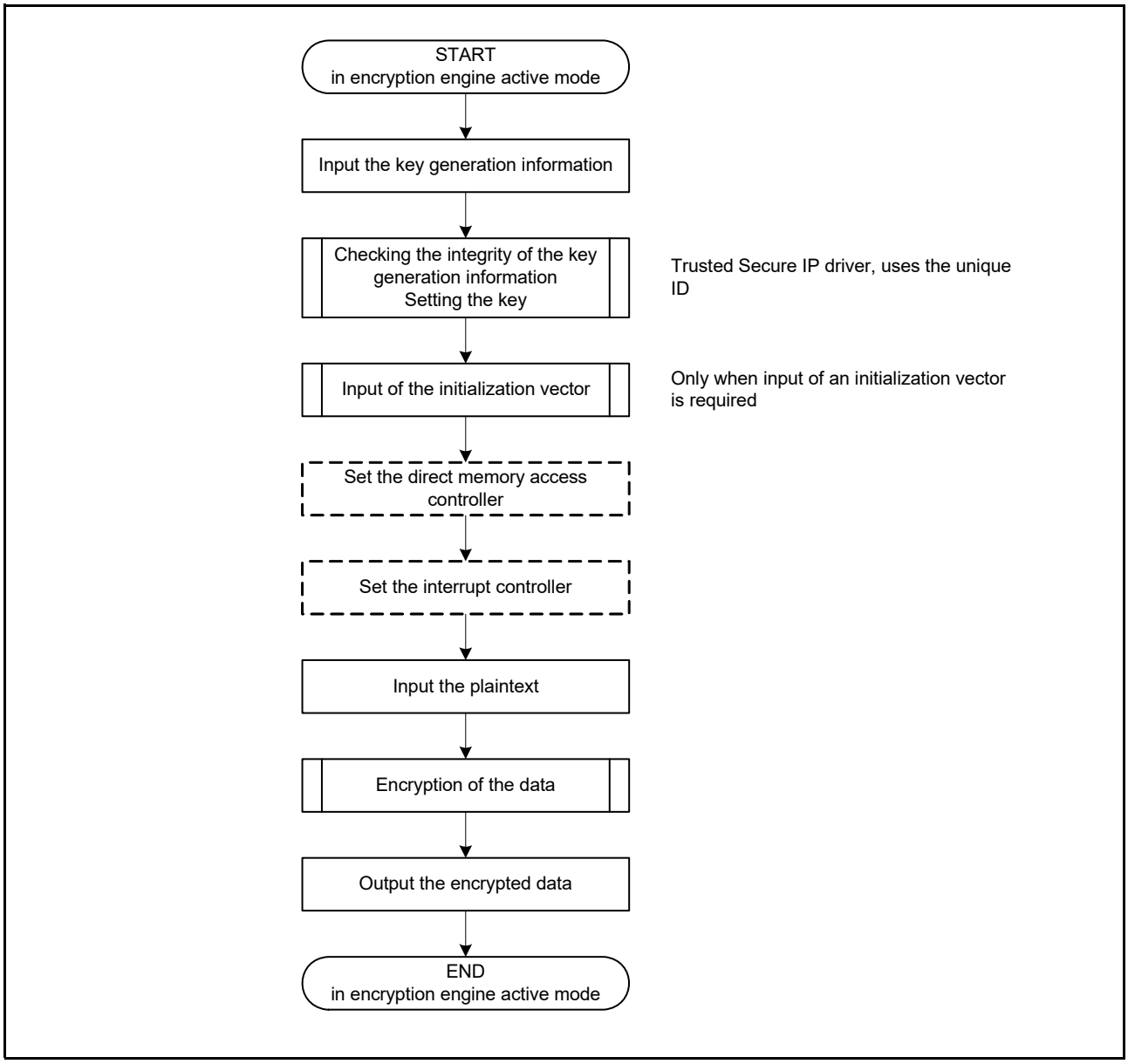


Figure 54.7 Encryption Flow Chart

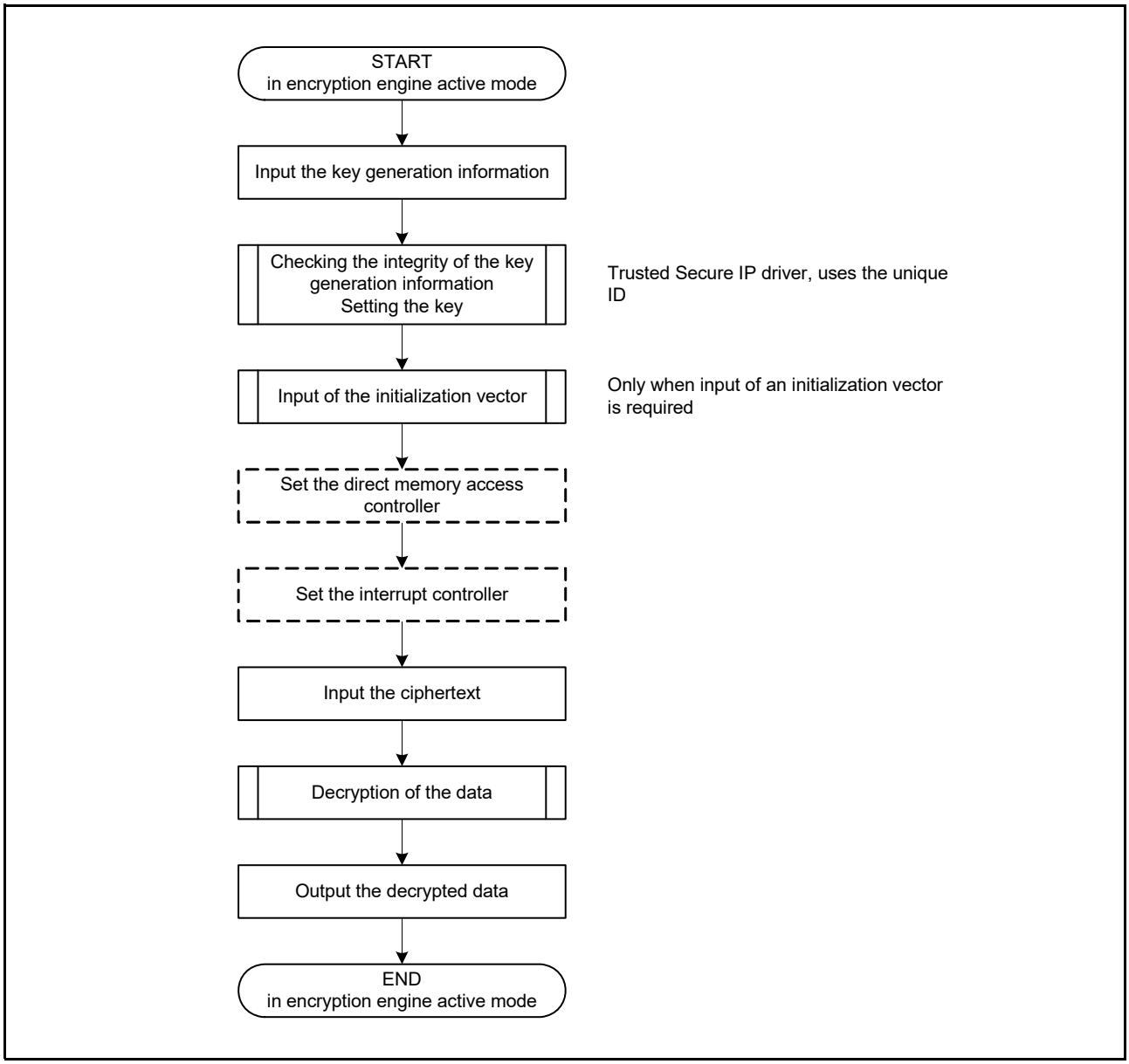


Figure 54.8 Decryption Flow Chart

54.2.5 Generating Key Generation Information (by Using Random Numbers)

Figure 54.9 shows the generating flow for the key generation information by using random numbers.

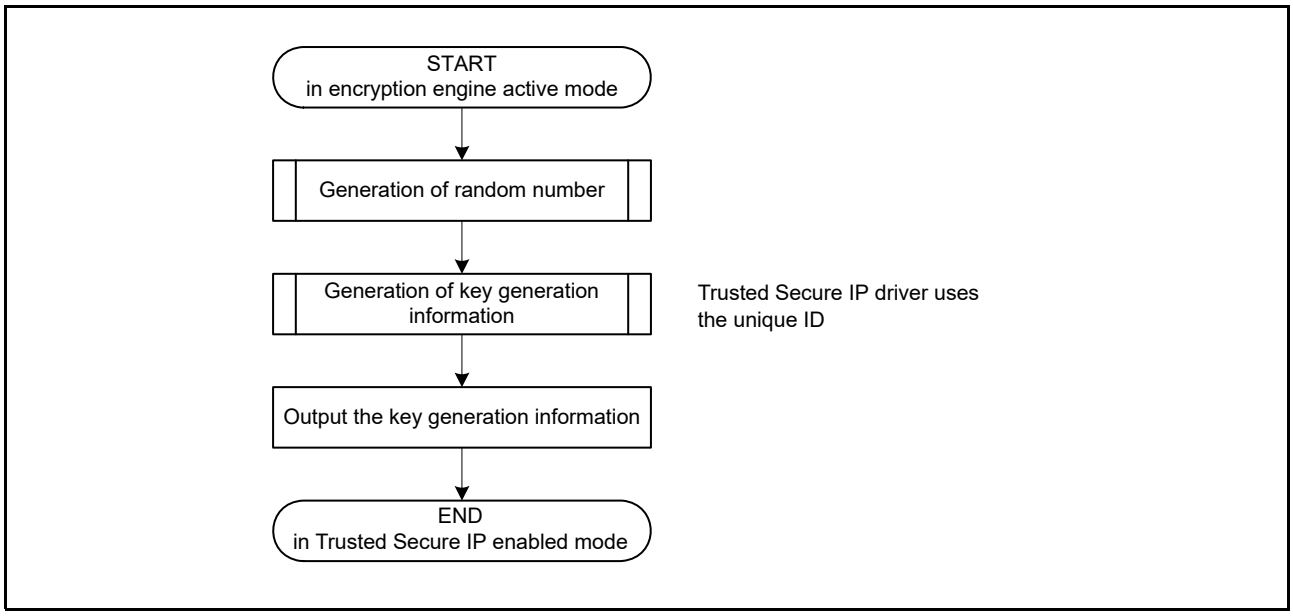


Figure 54.9 Key Generation Information Generating Flow Chart (Using Random Numbers)

54.2.6 Random Number Generation

Figure 54.10 shows the random number generation flow.

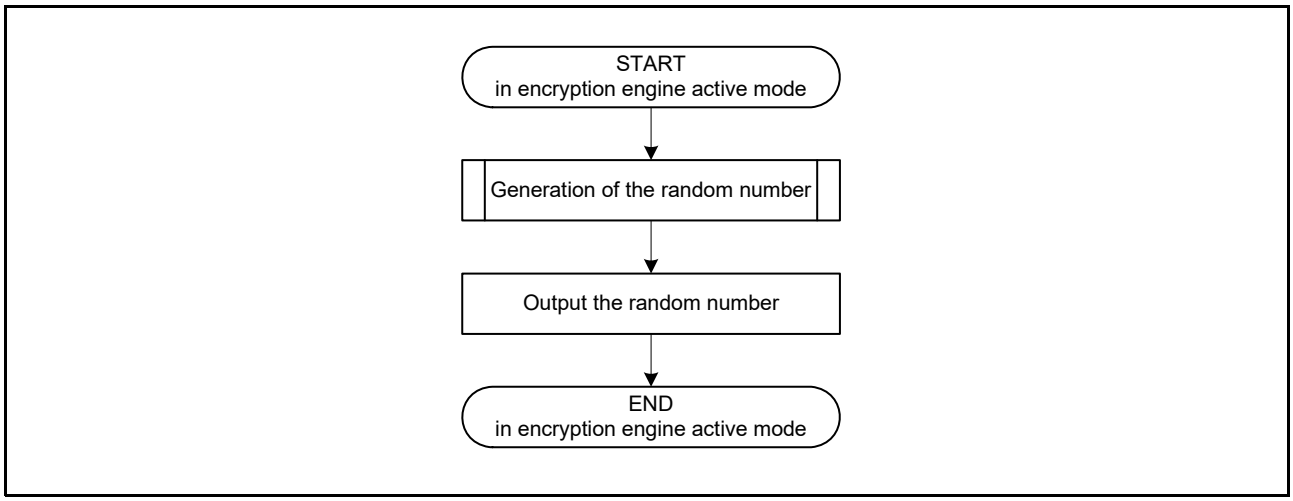


Figure 54.10 Random Number Generation Flow Chart

54.3 Interrupt

Table 54.2 lists the interrupt sources.
Trusted Secure IP driver uses interrupts caused by these interrupt sources. Do not change the setting of the interrupt controller corresponding to these interrupt sources.

Table 54.2 Trusted Secure IP Interrupt Sources

Name	Interrupt Source
PROC_BUSY	Procedure completion interrupt
ROMOK	Falsification detection interrupt
LONG_PLG	Calculation completion interrupt
WRRDY0	Write ready 0
WRRDY1	Write ready 1
WRRDY4	Write ready 4
RDRDY0	Read ready 0
RDRDY1	Read ready 1
IWRRDY	Integration write ready
IRDRDY	Integration read ready

54.4 Usage Notes

54.4.1 Setting the Module Stop Function

Operation of the Trusted Secure IP module is enabled or disabled by the MSTP107 bit of standby control register 10 (STBCR10), as stated in section 52.2.10, Standby Control Register 10 (STBCR10). After a reset, the Trusted Secure IP is stopped. After exiting the module stop state, the Trusted Secure IP can be accessed. Refer to section 52, Power-Down Modes for details.

54.4.2 Trusted Secure IP Driver

Use of the Trusted Secure IP requires the Trusted Secure IP driver provided by Renesas Electronics. Please contact our sales office for information regarding the Trusted Secure IP driver.

55. Register States

Table 55.1 Register States

Module	Register Name	Power-On Reset	Deep Standby	Software Standby	Module Standby	Sleep
Secondary cache	All registers	Initialized	Initialized	Retained	—	Retained
LSI internal bus	All registers	Initialized	Initialized	Retained	—	Retained
Clock pulse generator	FRQCR	Initialized*1	Initialized	Retained	—	Retained
	CKIOSEL	Initialized	Initialized	Retained	—	Retained
	SCLKSEL	Initialized*2	Initialized*2	Retained	—	Retained
Interrupt controller	All registers	Initialized	Initialized	Retained	—	Retained
Bus state controller	RTCSR	Initialized	Initialized	Retained	—	Retained*3
	RTCNT	Initialized	Initialized	Retained	—	Retained*4
	All registers other than above	Initialized	Initialized	Retained	—	Retained
Direct memory access controller	All registers	Initialized	Initialized	Retained	—	Retained*5
Multi-function timer pulse unit 3	All registers	Initialized	Initialized	Retained	Retained	Retained
Port output enable 3	All registers	Initialized	Initialized	Retained	—	Retained
General purpose PWM timer	All registers	Initialized	Initialized	Retained	Retained	Retained
Port output enable for GPT	All registers	Initialized	Initialized	Retained	—	Retained
OS timer	All registers	Initialized	Initialized	Retained	Retained	Retained
Watchdog timer	WRCSR	Initialized*1	Initialized	Retained	—	Retained
	PESR	Initialized*1	Initialized	Retained	—	Retained
	All registers other than above	Initialized	Initialized	Retained	—	Retained
Realtime clock	R64CNT	Retained*4	Retained*4	Retained*4	Retained*4	Retained*4
	RSECCNT					
	BCNT0					
	RMINCNT					
	BCNT1					
	RHRCNT					
	BCNT2					
	RWKCNT					
	BCNT3					
	RDAYCNT					
	RMONCNT					
	RYRCNT					
	RSECAR					
	BCNT0AR					
	RMINAR					
	BCNT1AR					
	RHRAR					
	BCNT2AR					
	RWKAR					
	BCNT3AR					
	RDAYAR					
	BCNT0AER					

Table 55.1 Register States

Module	Register Name	Power-On Reset	Deep Standby	Software Standby	Module Standby	Sleep
Realtime clock	RMONAR	Retained*4	Retained*4	Retained*4	Retained*4	Retained*4
	BCNT1AER					
	RYRAR					
	BCNT2AER					
	RYRAREN					
	BCNT3AER					
	RCR1	Retained*6	Retained*6	Retained	Retained	Retained
	RCR2	Retained*7	Retained*7	Retained	Retained	Retained
	RCR3	Retained	Retained	Retained	Retained	Retained
	RCR4	Retained	Retained	Retained	Retained	Retained
	RFRH	Retained	Retained	Retained	Retained	Retained
	RFRL	Retained	Retained	Retained	Retained	Retained
	RADJ	Retained	Retained	Retained	Retained	Retained
Serial communication interface with FIFO	All registers	Initialized	Initialized	Retained	Retained	Retained
Serial communication interface	All registers	Initialized	Initialized	Retained	Retained	Retained
Renesas serial peripheral interface	All registers	Initialized	Initialized	Retained	Retained	Retained
SPI multi I/O bus controller	All registers	Initialized*2	Initialized*2	Retained	Retained	Retained
HyperBus controller	All registers	Initialized*2	Initialized*2	Retained	Retained	Retained
Octa memory controller	All registers	Initialized*2	Initialized*2	Retained	Retained	Retained
I^2C bus interface	All registers	Initialized	Initialized	Retained	Retained	Retained
Serial sound interface	All registers	Initialized	Initialized	Retained	Retained	Retained
CANFD interface	All registers	Initialized	Initialized	Retained	Retained	Retained
Renesas SPDIF interface	All registers	Initialized	Initialized	Retained	Retained	Retained
Ethernet controller	All registers	Initialized	Initialized	Retained	Retained	Retained
PTP module for the Ethernet controller	All registers	Initialized	Initialized	Retained	Retained	Retained
DMA controller for the Ethernet controller	All registers	Initialized	Initialized	Retained	Retained	Retained
A/D converter	All registers	Initialized	Initialized	Retained	Retained	Retained
NAND flash controller	All registers	Initialized	Initialized	Retained	Retained	Retained
USB 2.0 host module	All registers	Initialized	Initialized	Retained	Retained	Retained
USB 2.0 function module	All registers	Initialized	Initialized	Retained	Retained	Retained
Video display controller 6	All registers	Initialized	Initialized	Retained	Retained	Retained
LVDS output interface	All registers	Initialized	Initialized	Retained	Retained	Retained
Image renderer	All registers	Initialized	Initialized	Retained	Retained	Retained
2D drawing engine	All registers	Initialized	Initialized	Retained	Retained	Retained
Sprite engine	All registers	Initialized	Initialized	Retained	Retained	Retained
JPEG codec unit	All registers	Initialized	Initialized	Retained	Retained	Retained
Capture engine unit	All registers	Initialized	Initialized	Retained	Retained	Retained
MIPI CSI-2 interface	All registers	Initialized	Initialized	Retained	Retained	Retained
Video input module	All registers	Initialized	Initialized	Retained	Retained	Retained
SD/MMC host interface	All registers	Initialized	Initialized	Retained	Retained	Retained
GPIO	All registers	Initialized*2	Initialized*2	Retained	—	Retained

Table 55.1 Register States

Module	Register Name	Power-On Reset	Deep Standby	Software Standby	Module Standby	Sleep
Power-down modes	DSFR, USBBSFR	Initialized*1	Retained	Retained	—	Retained
	XTALCTR	Initialized*1	Retained*8	Retained*8	—	Retained
	RTCXTALSEL	—	Retained	Retained	—	Retained
	All registers other than above	Initialized*2	Initialized*2	Retained	—	Retained
Debugger interface	Registers in CA9 PMU*9	Initialized	Initialized	Retained	—	Retained
	All registers other than above*10	Retained	Initialized*11	Retained	Retained	Retained
OTP	All registers	Initialized*12	Initialized*12	Retained	—	Retained
DRP	All registers	Initialized	Initialized	Retained	Retained	Retained
Trusted Secure IP	All registers	Initialized	Initialized	Retained	Retained	Retained

Note 1. Retained on internal power-on reset by the watchdog timer.
Note 2. Note that the value differs from the initial value as set by the boot program. For details, see section 3, Boot Mode.
Note 3. Flag processing continues.
Note 4. Counting up continues.
Note 5. Transfer can proceed.
Note 6. The CIE bit is initialized.
Note 7. The RESET and ADJ30 bits are initialized.
Note 8. Initialized when the realtime clock is not using the EXTAL pin.
Note 9. Access to these registers from the I/O area (SLV7) is not possible while TRST# is asserted.
Note 10. Initialized on assertion of TRST#.
Note 11. Retained in FAKE debug mode.
Note 12. The initial values of some registers for the OTP memory depend on values written to the memory.

56. Electrical Characteristics

56.1 Absolute Maximum Ratings

Table 56.1 Absolute Maximum Ratings

Item		Symbol	Value	Unit
Power supply voltage (I/O)		PVcc	–0.3 to 4.2	V
Power supply voltage (1.8-/3.3-V switchable I/O)		PVcc_SPI PVcc_SD0 PVcc_SD1	–0.3 to 4.2	V
Power supply voltage (1.8-V I/O)		PVcc_HO	–0.3 to 4.2	V
Power supply voltage (Internal)		Vcc	–0.3 to 1.6	V
PLL power supply voltage		PLLVcc	–0.3 to 1.6	V
Analog power supply voltage		AVcc	–0.3 to 4.2	V
USB transceiver analog power supply voltage (I/O)		USBAPVcc1 USBAPVcc0	–0.3 to 4.2	V
USB transceiver digital power supply voltage (I/O)		USBDPVcc1 USBDPVcc0	–0.3 to 4.2	V
LVDS analog power supply voltage		LVDSAPVcc	–0.3 to 4.2	V
LVDS PLL power supply voltage		LVDSPLLVcc	–0.3 to 1.6	V
MIPI analog power supply voltage		MIPIAVcc18	–0.3 to 2.6	V
Input voltage	1.8-V I/O input pins	V_{in}	–0.3 to 1.8-V power supply (PVcc_HO, MIPIAVcc18) + 0.3	V
	1.8-/3.3-V switchable I/O input pins	V_{in}	–0.3 to 1.8-/3.3-V power supply (PVcc_SPI, PVcc_SD0, PVcc_SD1) + 0.3	V
	Other input pins	V_{in}	–0.3 to 3.3-V power supply (PVcc, AVcc, USBAPVcc1, USBAPVcc0, USBDPVcc1, USBDPVcc0, LVDSAPVcc) + 0.3	V
Operating temperature	Ambient temperature	Ta	–40 to +85	°C
	Junction temperature	Tj	–40 to +125	
Storage temperature		T_{stg}	–55 to +125	°C

Caution: Permanent damage to the LSI may result if absolute maximum ratings are exceeded

56.2 Power-On/Power-Off Sequence

The 1.2-V power supply (Vcc, PLLVcc, and LVDSPLLVcc), 1.8-V power supply (PVcc_HO and MIPIAVcc18), 1.8-/3.3-V switchable power supply (PVcc_SPI, PVcc_SD0, PVcc_SD1), and 3.3-V power supply (PVcc, AVcc, USBAPVcc1, USBAPVcc0, USBDPVcc1, USBDPVcc0, and LVDSAPVcc) can be turned on and off in any order.
When turning on the power, be sure to drive both the TRST# and RES# pins low; otherwise, the output pins and input/output pins output undefined levels, resulting in system malfunction.
When turning off the power, drive the TRST# and RES# pins low if the undefined output may cause a problem.

56.3 DC Characteristics

- Conditions used to obtain DC characteristics (2) in Table 56.2 other than current consumption
  Vcc = 1.14 to 1.26 V, PVcc = USBDPVcc0 = USBDPVcc1 = AVcc = 3.0 to 3.6 V,
  PVcc_SPI = 3.0 to 3.6 V/1.7 to 1.9 V, PVcc_SD0 = 3.0 to 3.6 V/1.7 to 1.9 V,
  PVcc_SD1 = 3.0 to 3.6 V/1.7 to 1.9 V, PLLVcc = 1.14 to 1.26 V,
  USBAPVcc0 = USBAPVcc1 = 3.0 to 3.6 V, LVDSAPVcc = 3.0 to 3.6 V,
  LVDSPLLVcc = 1.14 to 1.26 V, PVcc_HO = MIPIAVcc18 = 1.7 to 1.9 V,
  Vss = AVss = USBVss = 0 V
  Ta = –40 to +85°C, Tj = –40 to +125°C

- Conditions used to obtain DC characteristics (2) in Table 56.2 for current consumption
  Vcc = 1.20 V, PVcc = USBDPVcc0 = USBDPVcc1 = AVcc = 3.3 V,
  PVcc_SPI = 3.3 V/1.8 V, PVcc_SD0 = 3.3 V/1.8 V, PVcc_SD1 = 3.3 V/1.8 V,
  PLLVcc = 1.20 V, USBAPVcc0 = USBAPVcc1 = 3.3 V, LVDSAPVcc = 3.3 V,
  LVDSPLLVcc = 1.20 V, PVcc_HO = MIPIAVcc18 = 1.8 V,
  Vss = AVss = USBVss = 0 V
  Ta = –40 to +85°C, Tj = –40 to +125°C
  Iφ = 528.0 MHz, Gφ = 264.00 MHz, Bφ = 132.00 MHz, P1φ = 66.00 MHz, P0φ = 33.00 MHz

Table 56.2 DC Characteristics (1) [Common Items]

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Power supply voltage		PVcc	3.0	3.3	3.6	V	
		PVcc_SPI PVcc_SD0 PVcc_SD1	1.7	1.8	1.9	V	1.8-V power supply selection
			3.0	3.3	3.6	V	3.3-V power supply selection
		PVcc_HO	1.7	1.8	1.9	V	
		Vcc	1.14	1.20	1.26	V	
PLL power supply voltage		PLLVcc	1.14	1.20	1.26	V	
Analog power supply voltage		AVcc	3.0	3.3	3.6	V	
USB power supply voltage		USBAPVcc1 USBAPVcc0 USBDPVcc1 USBDPVcc0	3.0	3.3	3.6	V	
LVDS analog power supply voltage		LVDSAPVcc	3.0	3.3	3.6	V	
LVDS PLL power supply voltage		LVDSPLLVcc	1.14	1.20	1.26	V	
MIPI analog power supply voltage		MIPIAVcc18	1.7	1.8	1.9	V	
Input leakage current	All input pins (except MIPI CSI-2 interface Related Pins*1)	$\lvert I_{in} \rvert$	—	—	1.0	μA	V_{in} = 0.5 to PVcc – 0.5 V
	MIPI CSI-2 interface Related Pins*1				10	μA	V_{in} = 0.5 to MIPIAVcc18 – 0.5 V
Three-state leakage current	All input/output pins, all output pins (except PD_0 to PD_7) (off state)	$\lvert I_{STI} \rvert$	—	—	1.0	μA	V_{in} = 0.5 to PVcc – 0.5 V
	PD_0 to PD_7		—	—	10	μA	
Input capacitance	USB 2.0 Host/Function Module-Related Pins*2	C_{in}	—	—	20	pF	
	All input/output pins and all input pins other than above		—	—	10	pF	

Note 1. CSI_DATA0P, CSI_DATA0N, CSI_DATA1P, CSI_DATA1N, CSI_CLKP, CSI_CLKN pins
Note 2. DP1, DP0, DM1, DM0 pins

Table 56.2 DC Characteristics (2) [Current Consumption]

Item	Power Supply	Symbol	Typ.	Max.	Unit	Test Conditions
Current consumption in normal operation	Vcc	Icc	—	994	mA	When the DRP is not in use.
			—	400	mA	When the DRP is in use, these values are added to that of the above columns, respectively.
	PLLVcc	PLLIcc	—	21.8	mA	
	LVDSPLLVcc	LVDSPLLIcc	—	2.1	mA	
	MIPIAVcc18	MIPIAIcc18	—	10	mA	During MIPI transfer
	PVcc_HO	PIcc_HO	15*1	—	mA	PVcc_HO = 1.8 V HM_CK, HM_CK# / OM_SCLK = 132 MHz
	PVcc_SPI	PIcc_SPI	15*1	—	mA	PVcc_SPI = 3.3 V QSPI0_SPCLK, QSPI1_SPCLK = 66 MHz
			15*1	—	mA	PVcc_SPI = 1.8 V QSPI0_SPCLK, QSPI1_SPCLK = 132 MHz
	PVcc_SD0	PIcc_SD0	15*1	—	mA	PVcc_SD0 = 3.3 V SD0_CLK = 33 MHz
			11*1	—	mA	PVcc_SD0 = 1.8 V SD0_CLK = 132 MHz
	PVcc_SD1	PIcc_SD1	15*1	—	mA	PVcc_SD1 = 3.3 V SD1_CLK = 33 MHz
			11*1	—	mA	PVcc_SD1 = 1.8 V SD1_CLK = 132 MHz
	PVcc	PIcc	100*1	—	mA	
	USBDPVcc0 + USBDPVcc1	UDPIcc	52*1	—	mA	In USB high-speed operation (1ch)
			93*1	—	mA	In USB high-speed operation (2ch)
			19*1	—	mA	In USB full-speed operation (1ch)
			29*1	—	mA	In USB full-speed operation (2ch)
	USBAPVcc0 + USBAPVcc1	UAPIcc	—	11	mA	When 1-ch is in use
			—	18	mA	When 2-ch is in use
	LVDSAPVcc	LVDSAPIcc	—	30	mA	During LVDS transfer
	AVcc	AIcc	—	1	mA	During A/D conversion
Current consumption in sleep mode	Vcc	I_{sleep}	—	850	mA	
	For the other power supply, the current consumption is the same as in normal operation.					
Current consumption in software standby mode	Vcc + PLLVcc + LVDSPLLVcc	I_{sstby}	30	500	mA	
	MIPIAVcc18 + PVcc_SPI + PVcc_HO + PVcc_SD0 + PVcc_SD1	$PI_{sstby18\_33}$	0.4	3.4	µA	
	PVcc + AVcc + USBAPVcc1 + USBAPVcc0 + USBDPVcc1 + USBDPVcc0 + LVDSAPVcc	PI_{sstby}	3	20	µA	When the USB host/function is not in use
			4.5	5	mA	When the USB host/function is in use

Item	Power Supply	Symbol	Typ.	Max.	Unit	Test Conditions
Current consumption in deep standby mode	Vcc + PLLVcc + LVDSPLLVcc	I_{dstby}	16	155	µA	RAM 0 Kbytes retained
			22	170	µA	RAM 16 Kbytes retained
			28	185	µA	RAM 32 Kbytes retained
			40	215	µA	RAM 64 Kbytes retained
			64	275	µA	RAM 128 Kbytes retained
	MIPIAVcc18 + PVcc_SPI + PVcc_HO + PVcc_SD0 + PVcc_SD1	$PI_{dstby18\_33}$	0.4	2.1	µA	
	PVcc + AVcc + USBAPVcc1 + USBAPVcc0 + USBDPVcc1 + USBDPVcc0 + LVDSAPVcc	PI_{dstby}	2.4	7	µA	RTC is not operating
			7.7	12.5	µA	RTC_X1 selected
			1.3	—	mA	EXTAL 12 MHz selected, small gain
			1.5	—	mA	EXTAL 24 MHz selected, small gain

Note 1. Reference value. The actual operating current greatly depends on the system (such as slow rising/falling edges caused by IO load and toggle frequency). Be sure to determine the value using the actual system.

Table 56.2 DC Characteristics (3) [Except 1.8-/3.3-V Switchable I/O Interface, I²C Bus Interface, and USB 2.0 Host/ Function Module-Related Pins]

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input high voltage*1		V_{IH}	2.2	—	PVcc + 0.3	V	
Input low voltage*1		V_{IL}	–0.3	—	0.8	V	
Schmitt trigger input characteristics		VT^+	PVcc × 0.665	—	PVcc + 0.3	V	
		VT^-	–0.3	—	0.8	V	
		$VT^+ – VT^-$	0.2	—	—	V	
Output high voltage		V_{OH}	PVcc – 0.5	—	—	V	I_{OH} = –8.0 mA*2, –2.0 mA*3
Output low voltage		V_{OL}	—	—	0.4	V	I_{OL} = 8.0 mA*2, 2.0 mA*3
RAM standby voltage	Software standby mode (large-capacity on-chip RAM)	V_{RAMS}	0.85	—	—	V	Measured with Vcc as parameter
	Deep standby mode (only the on-chip RAM for data retention)	V_{RAMD}	1.14	—	—	V	

Note 1. Except for Schmitt trigger function
Note 2. This is the value when high driving ability is set with a pin for which high driving ability is selectable.
Note 3. This is the value when normal driving ability is set with a pin for which high driving ability is selectable or the value of the pin to which normal driving ability is fixed.

Table 56.2 DC Characteristics (4) [1.8-/3.3-V Switchable I/O Interface*]

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input high voltage	V_{IH}	1.27	—	PVcc_* + 0.3	V	PVcc_SPI with 1.8-V power supply selection
				2		PVcc_SD0 and PVcc_SD1 with 1.8-V power supply selection
		PVcc_* × 0.625	—	PVcc_* + 0.3	V	3.3-V power supply selection
Input low voltage	V_{IL}	–0.3	—	0.58	V	1.8-V power supply selection
		–0.3	—	PVcc_* × 0.25	V	3.3-V power supply selection
Output high voltage	V_{OH}	1.4	—	—	V	1.8-V power supply selection, I_{OH} = –2.0 mA
		PVcc_* × 0.75	—	—	V	3.3-V power supply selection, I_{OH} = –2.0 mA
Output low voltage	V_{OL}	—	—	0.45	V	1.8-V power supply selection, I_{OL} = 2.0 mA
		—	—	PVcc_* × 0.125	V	3.3-V power supply selection, I_{OL} = 2.0 mA

Note: PVcc_SPI, PVcc_SD0, PVcc_SD1

Table 56.2 DC Characteristics (5) [1.8-V I/O Interface*]

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input high voltage	V_{IH}	1.27	—	PVcc_* + 0.3	V	
Input low voltage	V_{IL}	–0.3	—	0.58	V	
Output high voltage	V_{OH}	1.4	—	—	V	I_{OH} = –2.0 mA
Output low voltage	V_{OL}	—	—	0.45	V	I_{OL} = 2.0 mA

Note: PVcc_HO

Table 56.2 DC Characteristics (6) [I^2C Bus Interface-Related Pins*]

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input high voltage	V_{IH}	PVcc × 0.7	—	PVcc + 0.3	V	
Input low voltage	V_{IL}	–0.3	—	PVcc × 0.3	V	
Schmitt trigger input characteristics	$V_{IH} – V_{IL}$	PVcc × 0.05	—	—	V	
Output low voltage	V_{OL}	—	—	0.4	V	I_{OL} = 3.0 mA (RIICnFER.FMPE = 0)
		—	—	0.4	V	I_{OL} = 20.0 mA (RIICnFER.FMPE = 1)

Note: The PD_0 to PD_7 pins are open-drain pins.

Table 56.2 DC Characteristics (7) [USB 2.0 Host/Function Module-Related Pins*]

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Reference resistance	R_{REF}	2.2 kΩ ± 1%				

Note: RREF0, RREF1 pins.

Table 56.2 DC Characteristics (8) [USB 2.0 Host/Function Module-Related Pins* (Low-Speed, Full-Speed, and High-Speed Common Items)]

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
DP pull-up resistance (when function is selected)	R_{pu}	0.900	—	1.575	kΩ	In idle mode
		1.425	—	3.090	kΩ	In transmit/receive mode
DP and DM pull-down resistance (when host is selected)	R_{pd}	14.25	—	24.80	kΩ	

Note: DP1, DP0, DM1, and DM0 pins

Table 56.2 DC Characteristics (9) [USB 2.0 Host/Function Module-Related Pins* (Low-Speed and Full-Speed)]

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input high voltage	V_{IH}	2.0	—	—	V	
Input low voltage	V_{IL}	—	—	0.8	V	
Differential input sensitivity	V_{DI}	0.2	—	—	V	| (DP) – (DM) |
Differential common mode range	V_{CM}	0.8	—	2.5	V	
Output high voltage	V_{OH}	2.8	—	3.6	V	I_{OH} = –200 µA
Output low voltage	V_{OL}	0.0	—	0.3	V	I_{OL} = 2 mA
Output signal crossover voltage	V_{CRS}	1.3	—	2.0	V	C_L = 50 pF (full-speed) C_L = 200 to 600 pF (low-speed)

Note: DP1, DP0, DM1, and DM0 pins

Table 56.2 DC Characteristics (10) [USB 2.0 Host/Function Module-Related Pins* (High-Speed)]

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Squelch detection threshold voltage (differential voltage)	V_{HSSQ}	100	—	150	mV	
Common mode voltage range	V_{HSCM}	–50	—	500	mV	
Idle state	V_{HSOI}	–10.0	—	10.0	mV	
Output high voltage	V_{HSOH}	360	—	440	mV	
Output low voltage	V_{HSOL}	–10.0	—	10.0	mV	
Chirp J output voltage (difference)	V_{CHIRPJ}	700	—	1100	mV	
Chirp K output voltage (difference)	V_{CHIRPK}	–900	—	–500	mV	

Note: DP1, DP0, DM1, and DM0 pins

Table 56.2 DC Characteristics (11) [LVDS-Related Pins*]

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Differential Output Voltage	VOD	250	350	450	mV	R_L = 100Ω
Difference VOD between 'H' and 'L'	ΔVOD	—	—	50	mV	R_L = 100Ω
Offset (Common Mode) Voltage	VOS	1.125	1.25	1.375	V	R_L = 100Ω
Difference VOS between 'H' and 'L'	ΔVOS	—	—	50	mV	R_L = 100Ω

Note: TXCLKOUTP, TXCLKOUTM, TXOUT2P to TXOUT0P, and TXOUT2M to TXOUT0M pins

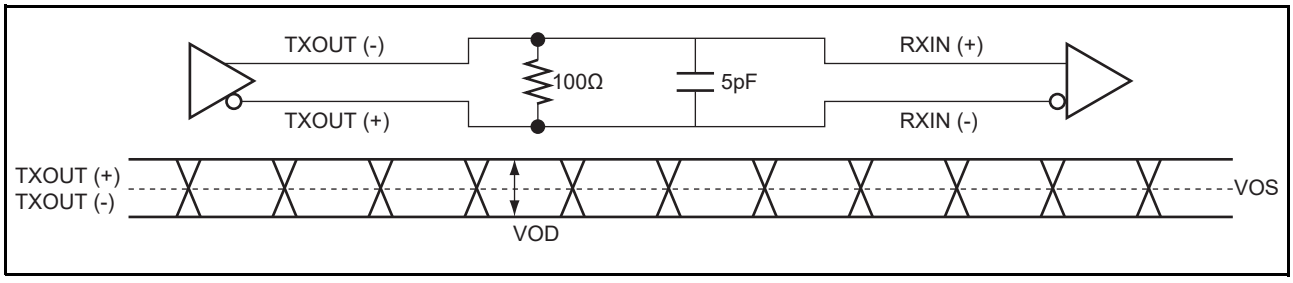


Figure 56.1 LVDS output waveform

Table 56.2 DC Characteristics (12) [MIPI CSI-2 interface-Related Pins*]

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input signal voltage range	V_{PIN}	–50	—	1350	mV	
Ground shift	V_{GNDSH}	–50	—	50	mV	
transient pin voltage level	$V_{PIN(absmax)}$	–0.15	—	1.45	V	
Differential input high threshold	V_{IDTH}	—	—	70	mV	HS Receiver
Differential input low threshold	V_{IDTL}	–70	—	—	mV	HS Receiver
Single-ended input high voltage	V_{IHHS}	—	—	460	mV	HS Receiver
Single-ended input low voltage	V_{ILHS}	–40	—	—	mV	HS Receiver
Input common mode voltage	$V_{CMRX(DC)}$	70	—	330	mV	HS Receiver
Differential input impedance	Z_{ID}	80	—	125	Ω	HS Receiver
Input high voltage	V_{IH}	880	—	—	mV	LP Receiver
Input low voltage	V_{IL}	—	—	550	mV	LP Receiver
Input hysteresis	V_{HYST}	—	50	—	mV	LP Receiver, Reference value

Note: CSI_DATA0P, CSI_DATA0N, CSI_DATA1P, CSI_DATA1N, CSI_CLKP, CSI_CLKN pins

Table 56.3 Permissible Output Currents

Item			Symbol	Min.	Typ.	Max.	Unit
Permissible output low current (per pin)	PD_0 to PD_7	High-speed mode	IOL	—	—	20	mA
		Normal mode				6	mA
	The CKIO, SD0_CLK, and SD1_CLK output pins, and output pins driven by the PVcc_SPI, or PVcc_HO power supply pin	High drive				12	mA
		Normal drive				8	mA
	SD0_CMD, SD0_DAT7 to SD0_DAT0, SD1_CMD, SD1_DAT3 to SD1_DAT0					6	mA
	Output pins other than above	High drive*1				8	mA
		Normal drive*2				2	mA
Permissible output high current (per pin)	The CKIO, SD0_CLK, and SD1_CLK output pins, and output pins driven by the PVcc_SPI, or PVcc_HO power supply pin	High drive	– IOH	—	—	12	mA
		Normal drive				8	mA
	SD0_CMD, SD0_DAT7 to SD0_DAT0, SD1_CMD, SD1_DAT3 to SD1_DAT0					6	mA
	Output pins other than above	High drive*1				8	mA
		Normal drive*2				2	mA
Permissible output current (total)			Σ IO	—	—	150	mA

Caution: To protect the LSI's reliability, the output current values should not exceed the values in Table 56.3.

Note 1. This is the value when high driving ability is set with a pin for which high driving ability is selectable.
Note 2. This is the value when normal driving ability is set with a pin for which high driving ability is selectable or the value of the pin to which normal driving ability is fixed.

56.4 AC Characteristics

Signals input to this LSI are basically handled as signals in synchronization with a clock. The setup and hold times for input pins must be followed.

Conditions for AC characteristics: Vcc = 1.14 to 1.26 V, PVcc = USBDPVcc0 = USBDPVcc1 = AVcc = 3.0 to 3.6 V, PVcc_SPI = 3.0 to 3.6 V/1.7 to 1.9 V, PVcc_SD0 = 3.0 to 3.6 V/1.7 to 1.9 V, PVcc_SD1 = 3.0 to 3.6 V/1.7 to 1.9 V, PLLVcc = 1.14 to 1.26 V, USBAPVcc0 = USBAPVcc1 = 3.0 to 3.6 V, LVDSAPVcc = 3.0 to 3.6 V, LVDSPLLVcc = 1.14 to 1.26 V, PVcc_HO = MIPIAVcc18 = 1.7 to 1.9 V,
Vss = AVss = USBVss = 0 V,
Ta = –40 to +85°C, Tj = –40 to +125°C

Table 56.4 Operating Frequency

Item		Symbol	Min.	Max.	Unit	Remarks
Operating frequency	CPU clock (Iφ)	f	55	528	MHz	
	Image processing clock (Gφ)		55	264	MHz	
	Internal bus clock (Bφ)		27.5	132	MHz	
	Peripheral clock 1 (P1φ)		27.5	66	MHz	
	Peripheral clock 0 (P0φ)		27.5	33	MHz	

56.4.1 Clock Timing

Table 56.5 Clock Timing

Item	Symbol	Min.	Max.	Unit	Figure
EXTAL clock input frequency (when the clock is supplied to USB 2.0 host/function module)	f_{EX}	12MHz ± 100ppm 24MHz ± 100ppm			Figure 56.2
EXTAL clock input frequency (when the clock isn't supplied to USB 2.0 host/function module)		10 20	12 24	MHz	
EXTAL clock input cycle time (when the clock isn't supplied to USB 2.0 host/function module)	t_{EXcyc}	83.34 41.67	100 50	ns	
AUDIO_X1 clock input frequency (crystal resonator connected)	f_{EX}	10.00	50.00	MHz	
AUDIO_X1 clock input cycle time (crystal resonator connected)	t_{EXcyc}	20.00	100.00	ns	
AUDIO_X1, AUDIO_CLK clock input frequency (external clock input)	f_{EX}	1.00	50.00	MHz	
AUDIO_X1, AUDIO_CLK clock input cycle time (external clock input)	t_{EXcyc}	20.00	1000.00	ns	
USB_X1 clock input frequency (when the 48-MHz clock is supplied to USB 2.0 host/function module)	f_{EX}	48MHz ± 100ppm			
EXTAL, AUDIO_X1, AUDIO_CLK, USB_X1 clock input low pulse width	t_{EXL}	0.4	0.6	t_{EXcyc}	
EXTAL, AUDIO_X1, AUDIO_CLK, USB_X1 clock input high pulse width	t_{EXH}	0.4	0.6	t_{EXcyc}	
EXTAL, AUDIO_X1, AUDIO_CLK, USB_ X1 clock input rise time	t_{EXr}	—	4	ns	
EXTAL, AUDIO_X1, AUDIO_CLK, USB_ X1 clock input fall time	t_{EXf}	—	4	ns	
CKIO clock output frequency	f_{OP}	27.5*1	132*1	MHz	
CKIO clock output cycle time	t_{cyc}	7.58*1	36.36*1	ns	Figure 56.3 (1), Figure 56.3 (2)
CKIO clock output low pulse width 1	t_{CKOL1}	t_{cyc} / 2 – t_{CKOr1}	—	ns	Figure 56.3 (1)
CKIO clock output high pulse width 1	t_{CKOH1}	t_{cyc} / 2 – t_{CKOf1}	—	ns	
CKIO clock output rise time 1	t_{CKOr1}	—	3	ns	
CKIO clock output fall time 1	t_{CKOf1}	—	3	ns	
CKIO clock output low pulse width 2	t_{CKOL2}	t_{cyc} / 2 – t_{CKOr2}	—	ns	Figure 56.3 (2)
CKIO clock output high pulse width 2	t_{CKOH2}	t_{cyc} / 2 – t_{CKOf2}	—	ns	
CKIO clock output rise time 2	t_{CKOr2}	—	2	ns	
CKIO clock output fall time 2	t_{CKOf2}	—	2	ns	
CKIO clock output low pulse width 3	t_{CKOL3}	t_{cyc} / 2 – t_{CKOr3}	—	ns	Figure 56.3 (3)
CKIO clock output high pulse width 3	t_{CKOH3}	t_{cyc} / 2 – t_{CKOf3}	—	ns	
CKIO clock output rise time 3	t_{CKOr3}	—	1*2	ns	
CKIO clock output fall time 3	t_{CKOf3}	—	1*2	ns	
On-chip PLL circuit oscillation settling time	t_{POSC}	1	—	ms	Figure 56.4, Figure 56.5
On-chip oscillation circuit oscillation settling time (RTC_X1)	t_{ROSC}	—	3*3	s	Figure 56.7
On-chip oscillation circuit oscillation settling time (other than above)		—	4*3	ms	Figure 56.4, Figure 56.5 , Figure 56.7
Mode hold time	t_{MDH}	200	—	ns	Figure 56.4, Figure 56.5

Note 1. The range of CKIO clock output frequency and cycle time is determined by register settings. See, Table 6.4 and Table 6.5.
Note 2. Output load: 15 pF

Note 3. Settings for values smaller than the above specifications may be possible, as long as the values are confirmed through evaluation by the manufacturer of the oscillator.

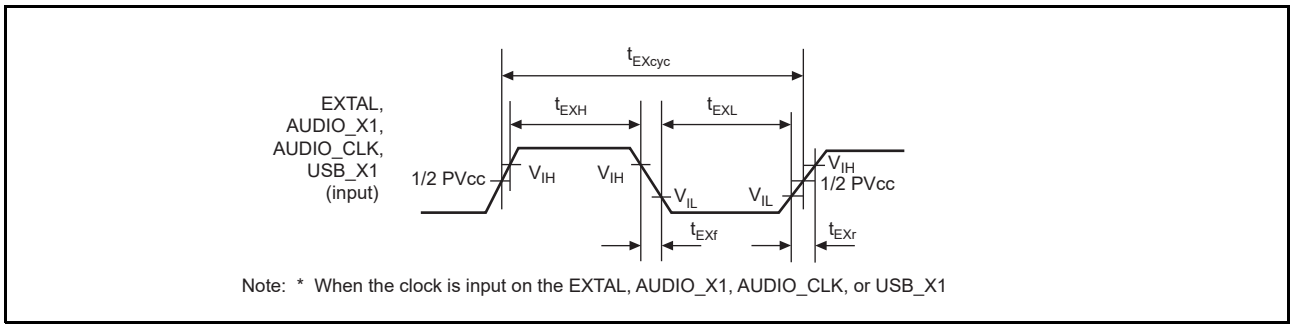


Figure 56.2 EXTAL, AUDIO_X1, AUDIO_CLK, and USB_X1 Clock Input Timing

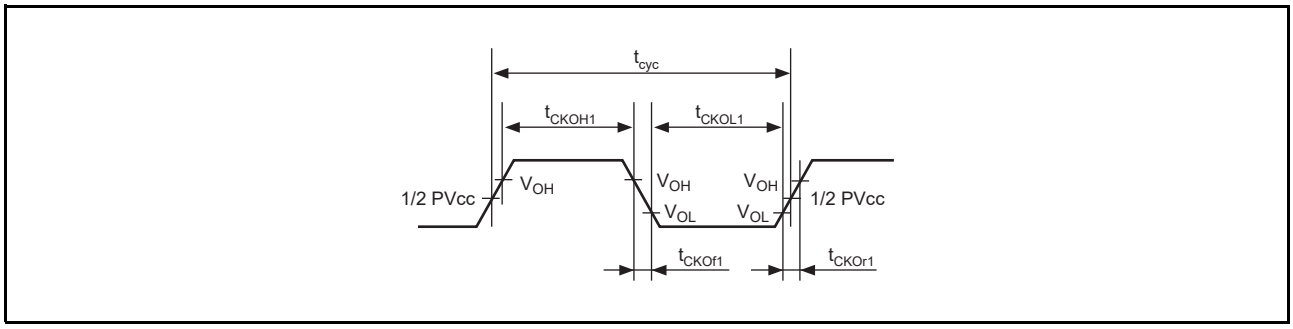


Figure 56.3 (1) CKIO Clock Output Timing 1

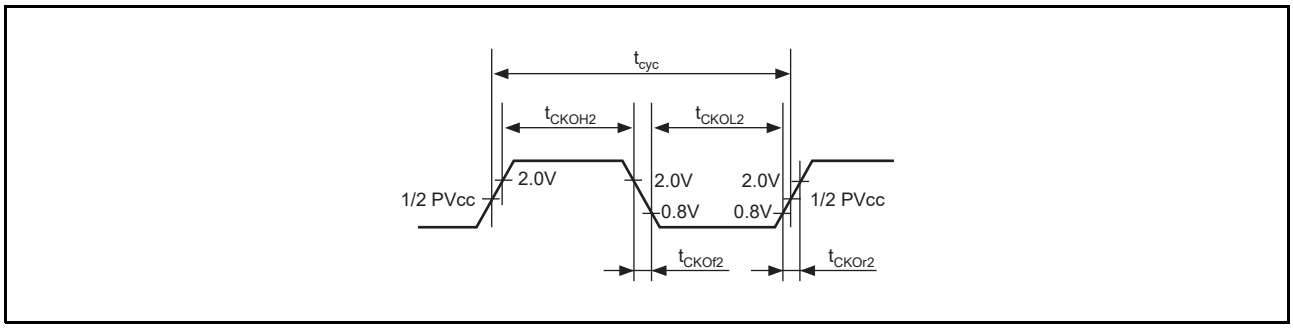


Figure 56.3 (2) CKIO Clock Output Timing 2

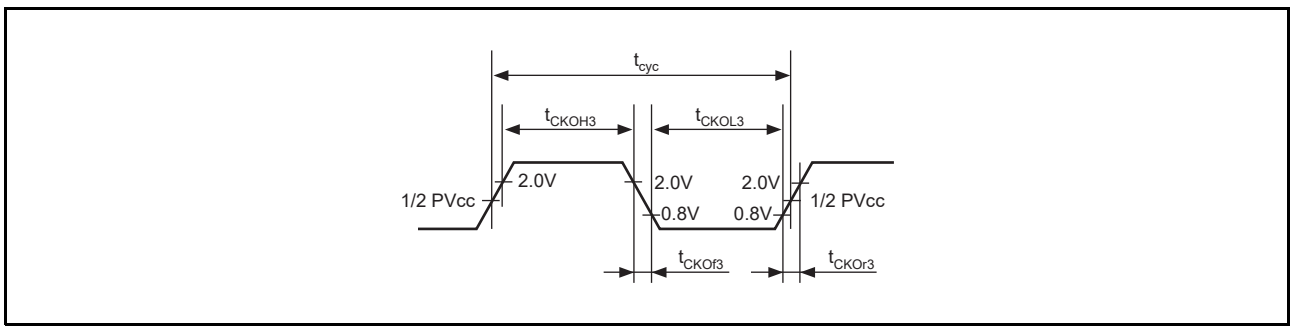


Figure 56.3 (3) CKIO Clock Output Timing 3

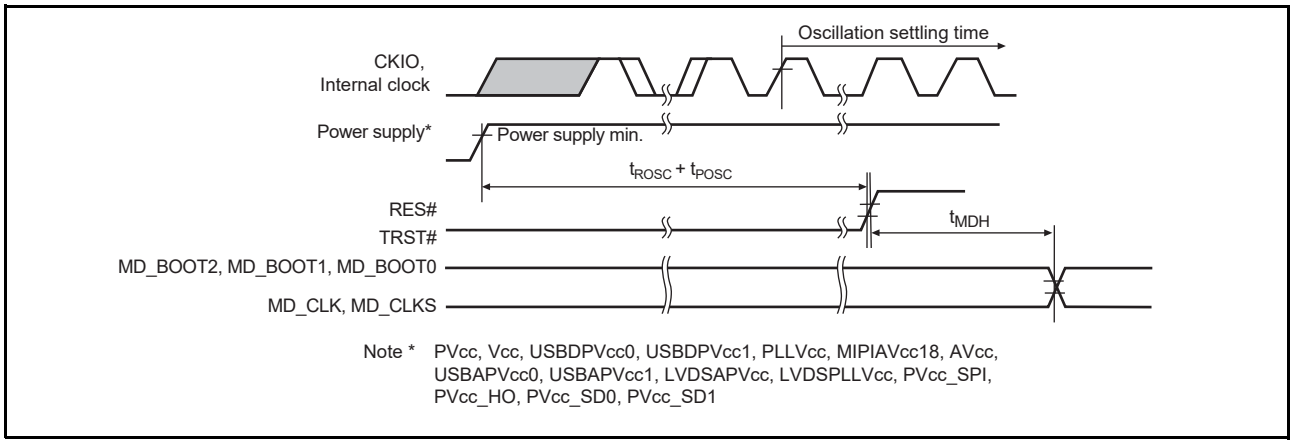


Figure 56.4 Power-On Oscillation Settling Time

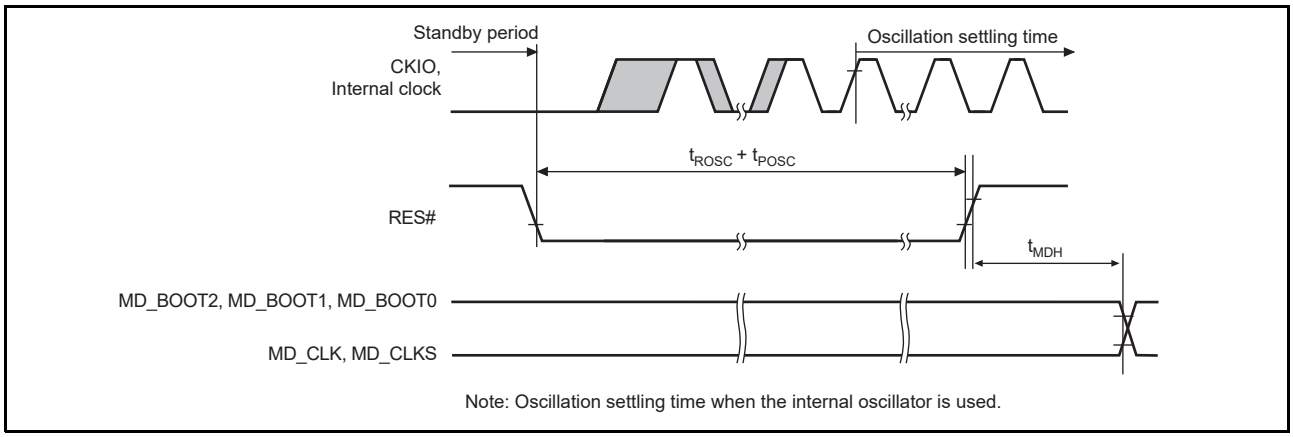


Figure 56.5 Oscillation Settling Time on Return from Standby (Return by Reset)

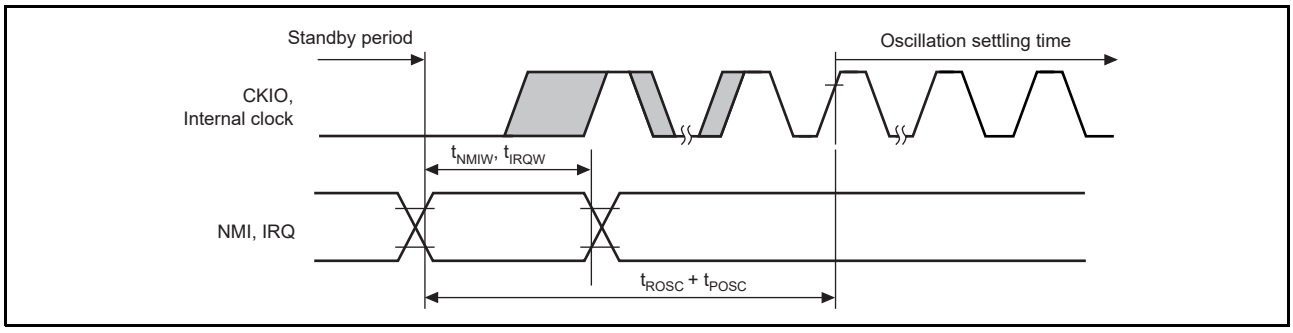
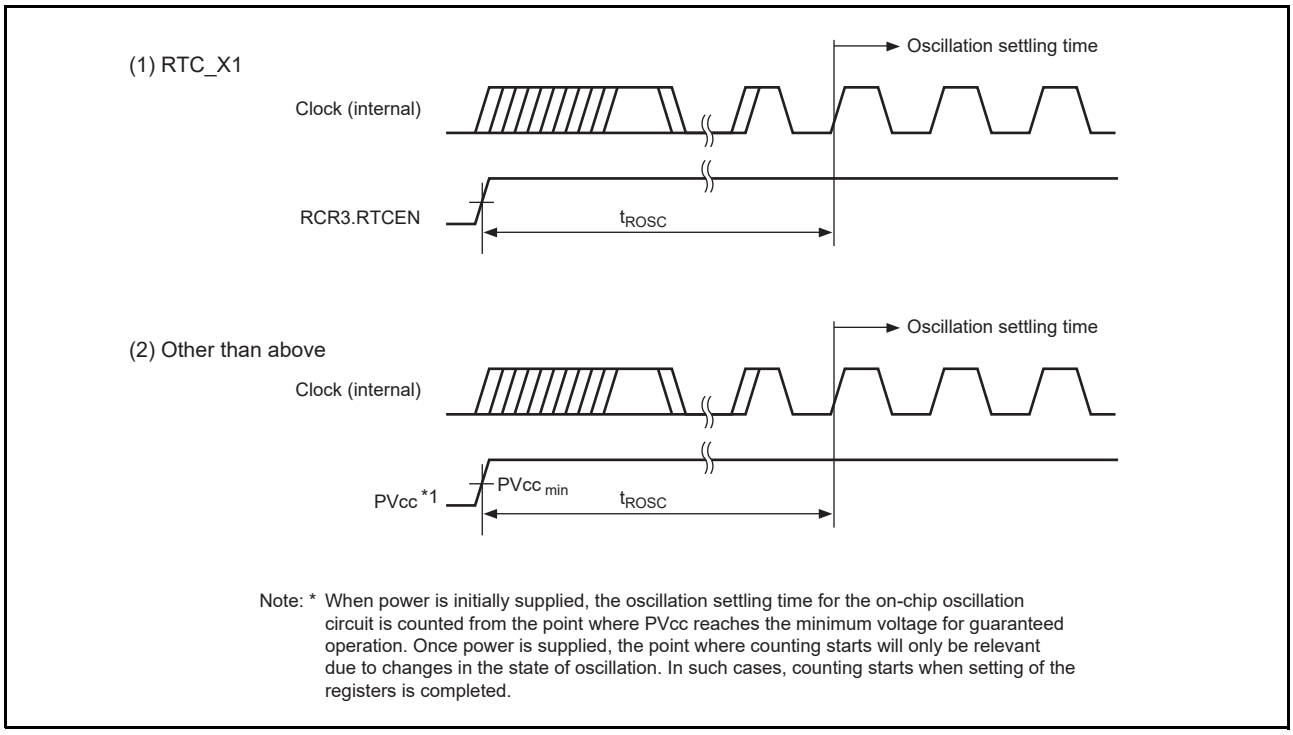


Figure 56.6 Oscillation Settling Time on Return from Standby (Return by NMI or IRQ)



Note: * When power is initially supplied, the oscillation settling time for the on-chip oscillation circuit is counted from the point where PVcc reaches the minimum voltage for guaranteed operation. Once power is supplied, the point where counting starts will only be relevant due to changes in the state of oscillation. In such cases, counting starts when setting of the registers is completed.

Figure 56.7 On-chip Oscillation Circuit Oscillation Settling Time

56.4.2 Control Signal Timing

Table 56.6 Control Signal Timing

Item		Symbol	Min.	Max.	Unit	Figure
RES# pulse width	Exit from standby mode	t_{RESW}	10	—	ms	Figure 56.8 (1)
	Other than above		20	—	tcyc*2	
TRST# pulse width		t_{TRSW}	20	—	tcyc*2	
NMI pulse width		t_{NMIW}	20	—	tcyc*2	Figure 56.3 (2) , Figure 56.6
IRQ pulse width		t_{IRQW}	20	—	tcyc*2	
TINT pulse width		t_{TINTW}	20	—	tcyc*2	
TINT input wait timing in the same group*1		t_{TINTGW}	20	—	tcyc*2	Figure 56.8 (2)
RES# input rise time		t_{RSr}	—	500	µs	Figure 56.8 (3)

Note 1. This restriction applies only to the same group of pin interrupts. For details, see section 51.4.4.2, Restriction on the Input from Pins in the Same Group.
Note 2. tcyc indicates the cycle when the setting of the CKIOSEL bits in the CKIO select register is 2'b01 (initial value).

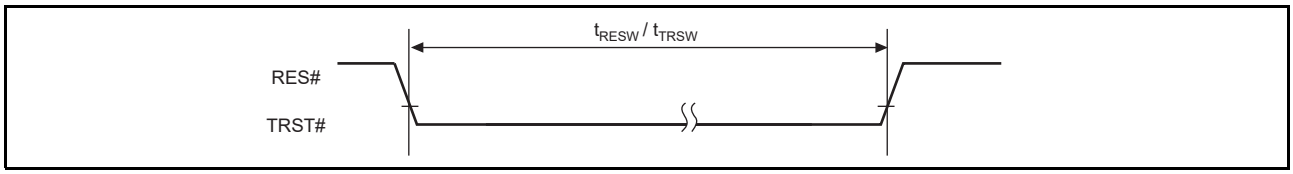


Figure 56.8 (1) Reset Input Timing 1

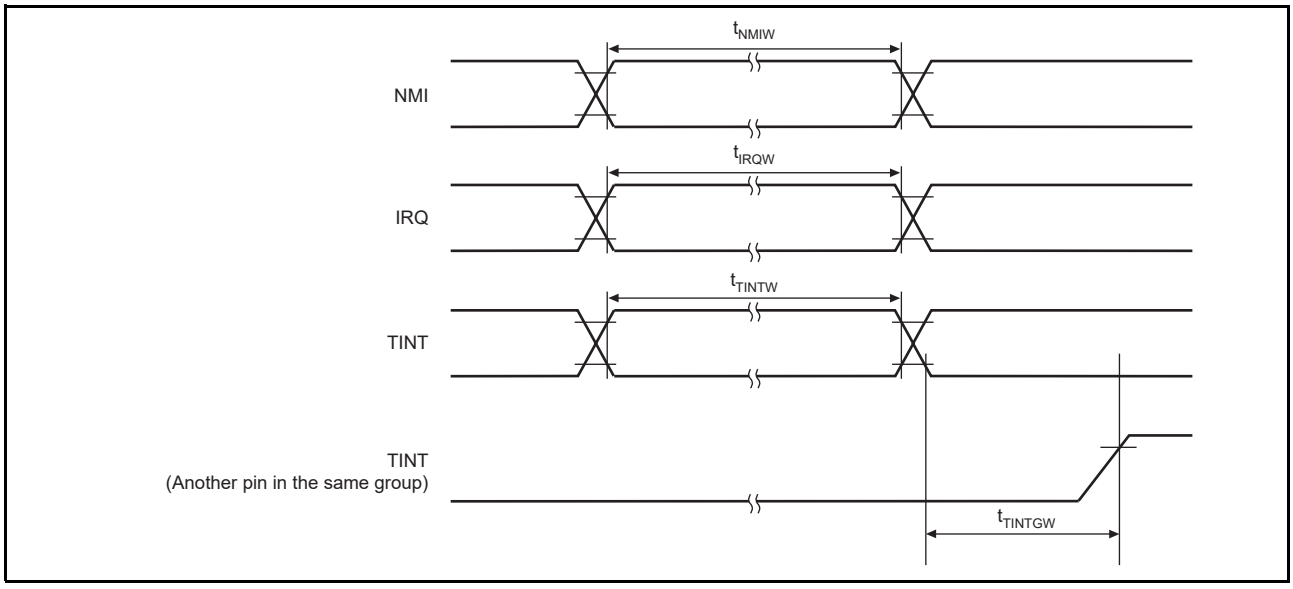


Figure 56.8 (2) Interrupt Signal Input Timing

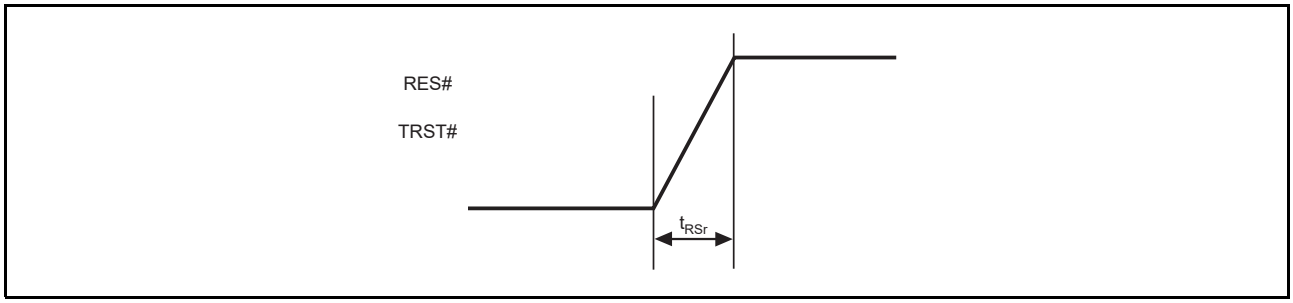


Figure 56.8 (3) Reset Input Timing 2

56.4.3 SPI Multi I/O Bus Controller, Octa Memory/HyperBus™ Controller Reset Output Timing

Table 56.7 SPI Multi I/O Bus Controller,Octa Memory/HyperBus™ Controller Reset Output Timing

Item	Symbol	Min.	Max.	Unit	Figure
Memory reset pulse width	t_{RP}	2500	—	t_{p0cyc}	Figure 56.9 to Figure 56.12
Memory reset (negate) to memory access time	t_{RH}	500	—	t_{p0cyc}	

Note: t_{p0cyc} indicates the peripheral clock 0 (P0φ) cycle.

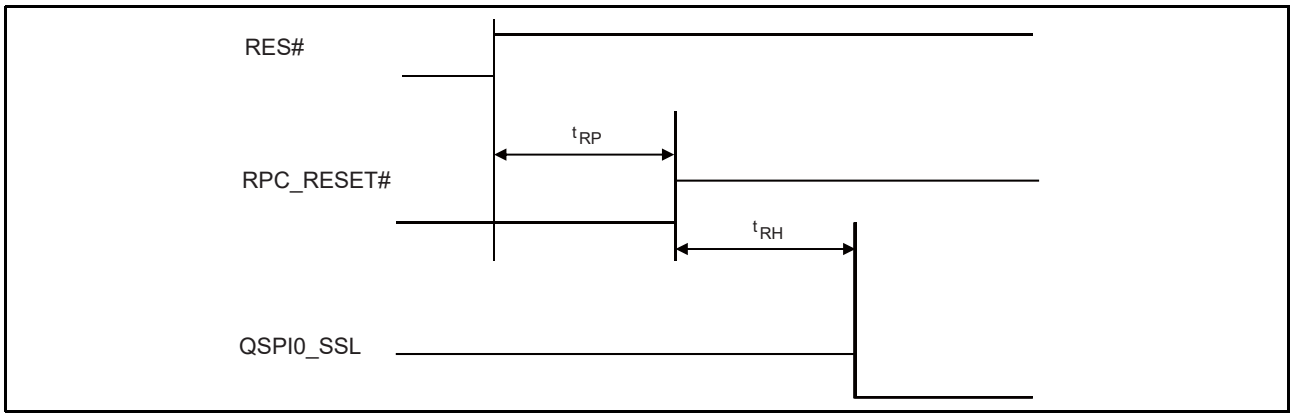


Figure 56.9 SPI Multi I/O Bus Controller Reset Output Timing after a power-on reset (Boot Modes 3, 4, and 5)

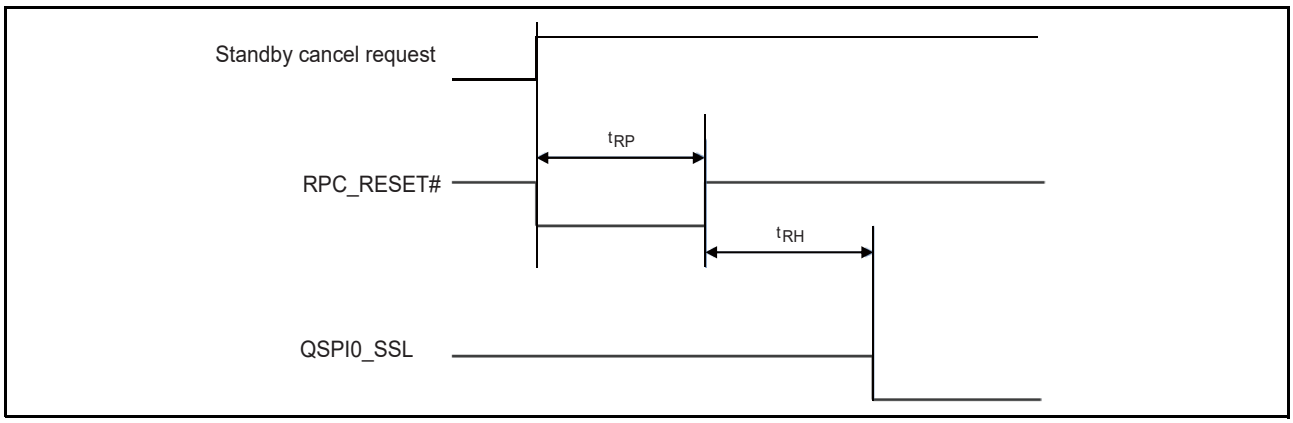


Figure 56.10 SPI Multi I/O Bus Controller Reset Output Timing after recovery from deep standby (Boot Modes 3, 4, and 5)

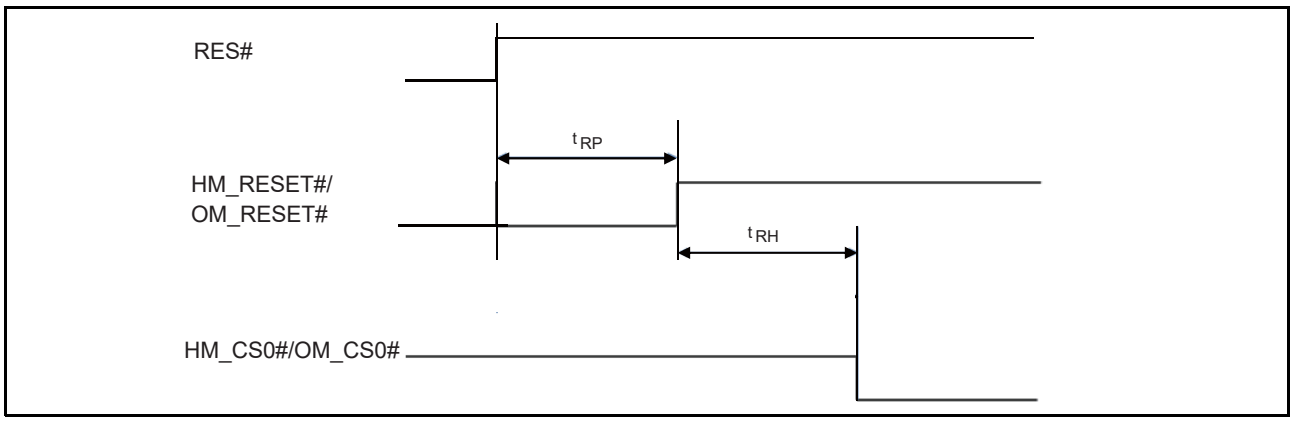


Figure 56.11 Octa Memory/HyperBus™ Controller Reset Output Timing after a power-on reset (Boot Modes 6 and 7)

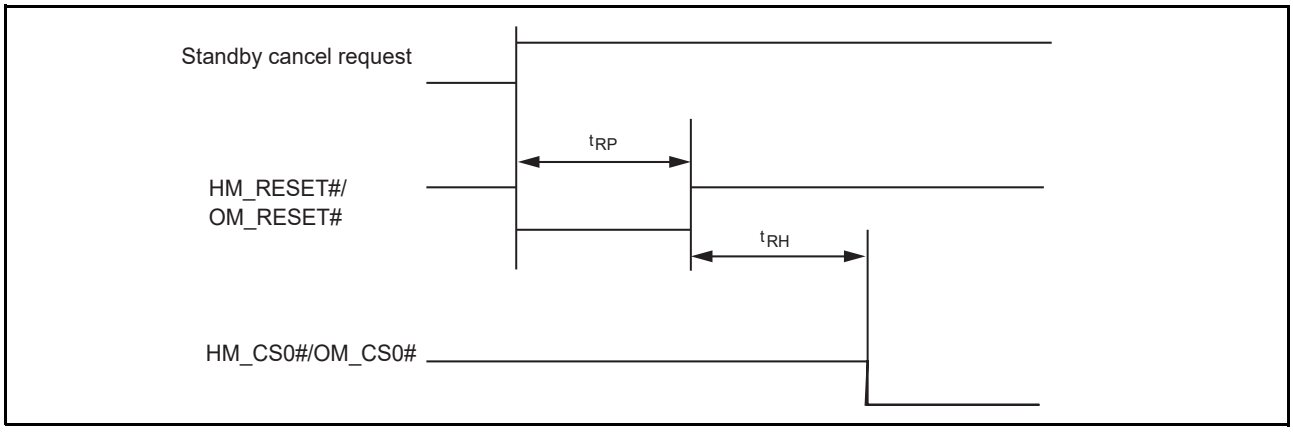


Figure 56.12 Octa Memory/HyperBus™ Controller Reset Output Timing after recovery from deep standby (Boot Modes 6 and 7)

56.4.4 Bus Timing

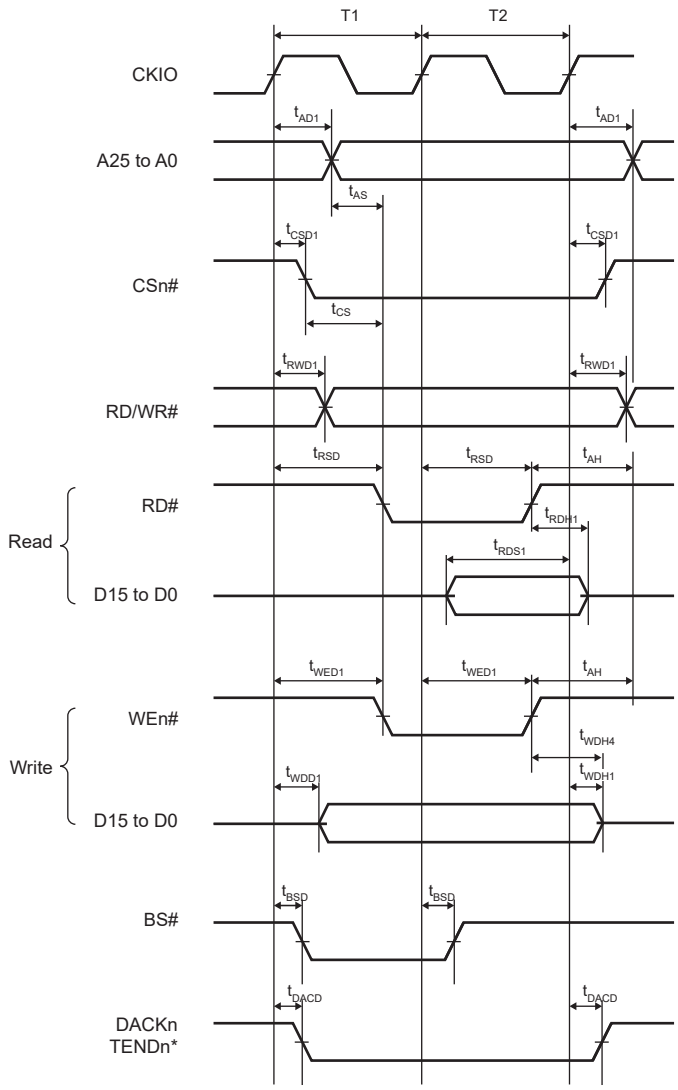
Table 56.8 Bus Timing

Item	Symbol	CKIO = 132 MHz*1 Min.	Max.	Unit	Figure
Address delay time 1	t_{AD1}	2.0	4.8	ns	Figure 56.13 to Figure 56.37
Address delay time 2	t_{AD2}	1 / 2 t_{cyc}	1 / 2 t_{cyc} + 8 *3	ns	Figure 56.20
Address setup time	t_{AS}	0	—	ns	Figure 56.13 to Figure 56.16, Figure 56.20
Chip enable setup time	t_{cs}	0	—	ns	Figure 56.13 to Figure 56.16, Figure 56.20
Address hold time	t_{AH}	0	—	ns	Figure 56.13 to Figure 56.16
BS delay time	t_{BSD}	—	4.8	ns	Figure 56.13 to Figure 56.34
CS delay time 1	t_{CSD1}	2.0	4.8	ns	Figure 56.13 to Figure 56.37
Read write delay time 1	t_{RWD1}	2.0	4.8	ns	Figure 56.13 to Figure 56.37
Read strobe delay time	t_{RSD}	1 / 2 t_{cyc}	1 / 2 t_{cyc} + 12 *3	ns	Figure 56.13 to Figure 56.20
Read data setup time 1	t_{RDS1}	1 / 2 t_{cyc} + 5 *3	—	ns	Figure 56.13 to Figure 56.19
Read data setup time 2	t_{RDS2}	2.0	—	ns	Figure 56.21 to Figure 56.24, Figure 56.29 to Figure 56.31
Read data setup time 3	t_{RDS3}	1 / 2 t_{cyc} + 5 *3	—	ns	Figure 56.20
Read data hold time 1	t_{RDH1}	0	—	ns	Figure 56.13 to Figure 56.19
Read data hold time 2	t_{RDH2}	2.5	—	ns	Figure 56.21 to Figure 56.24, Figure 56.29 to Figure 56.31
Read data hold time 3	t_{RDH3}	0	—	ns	Figure 56.20
Write enable delay time 1	t_{WED1}	1 / 2 t_{cyc}	1 / 2 t_{cyc} + 12 *3	ns	Figure 56.13 to Figure 56.18
Write enable delay time 2	t_{WED2}	—	4.8	ns	Figure 56.19
Write data delay time 1	t_{WDD1}	—	4.8	ns	Figure 56.13 to Figure 56.19
Write data delay time 2	t_{WDD2}	—	4.8	ns	Figure 56.25 to Figure 56.28, Figure 56.32 to Figure 56.34
Write data hold time 1	t_{WDH1}	2.0	—	ns	Figure 56.13 to Figure 56.19
Write data hold time 2	t_{WDH2}	2.0	—	ns	Figure 56.25 to Figure 56.28, Figure 56.32 to Figure 56.34
Write data hold time 4	t_{WDH4}	0	—	ns	Figure 56.13 to Figure 56.17
WAIT setup time	t_{WTS}	1 / 2 t_{cyc} + 5.0 *3	—	ns	Figure 56.14 to Figure 56.20
WAIT hold time	t_{WTH}	1 / 2 t_{cyc} + 3.5 *3	—	ns	Figure 56.14 to Figure 56.20
RAS delay time 1	t_{RASD1}	2.0	4.8	ns	Figure 56.21 to Figure 56.37
CAS delay time 1	t_{CASD1}	2.0	4.8	ns	Figure 56.21 to Figure 56.37
DQM delay time 1	t_{DQMD1}	2.0	4.8	ns	Figure 56.21 to Figure 56.34
CKE delay time 1	t_{CKED1}	2.0	4.8	ns	Figure 56.36
AH delay time	t_{AHD}	1 / 2 t_{cyc}	1 / 2 t_{cyc} + 12 *3	ns	Figure 56.17
Multiplexed address delay time	t_{MAD}	—	12*3	ns	Figure 56.17
Multiplexed address hold time	t_{MAH}	1.0*3	—	ns	Figure 56.17
Address setup time for AH	t_{AVVH}	1 / 2 t_{cyc} – 2	—	ns	Figure 56.17
DACK, TEND delay time	t_{DACD}	Refer to the direct memory access controller timing		ns	Figure 56.13 to Figure 56.34

Note 1. The maximum value (f_{max}) of CKIO (external bus clock) depends on the number of wait cycles and the system configuration of your board.

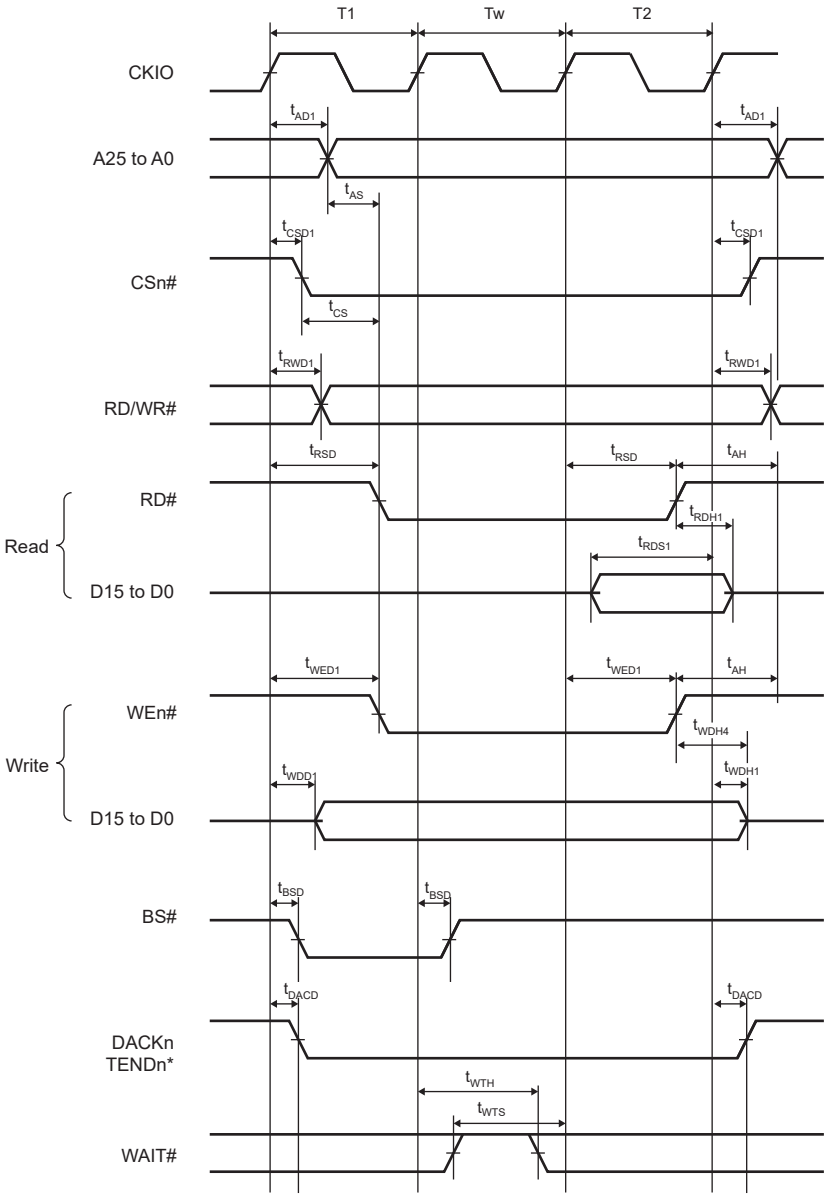
Note 2. 1/2 t_{cyc} indicated in minimum and maximum values for the item of delay, setup, and hold times represents a half cycle from the rising edge with a clock. That is, 1/2 tcyc describes a reference of the falling edge with a clock.

Note 3. These values specified are for 66 MHz max operation. For CKIO = 110 to 132 MHz operation, some wait cycle insertion and input timing design are required depending on the system configuration.



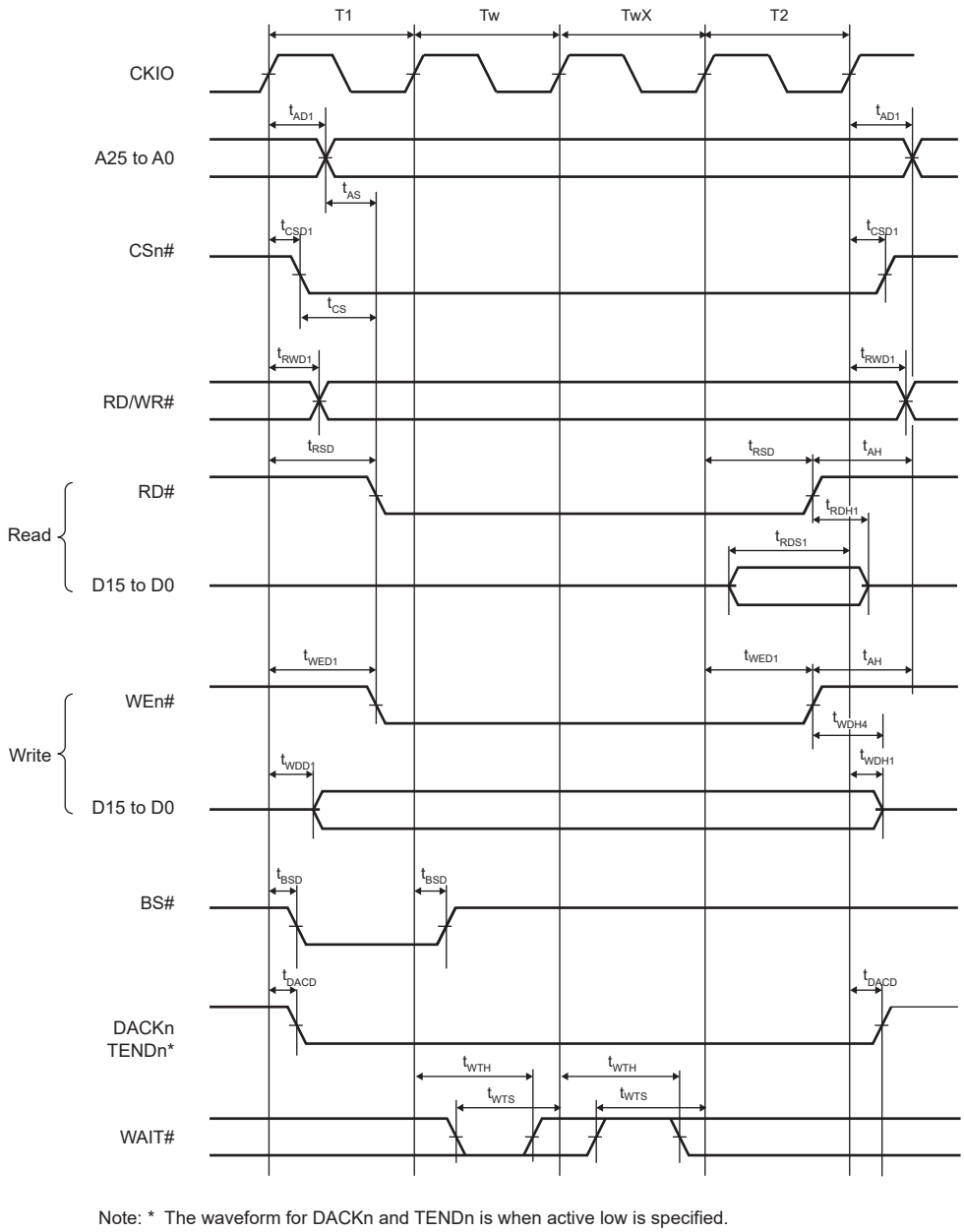
Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 56.13 Basic Bus Timing for Normal Space (No Wait)



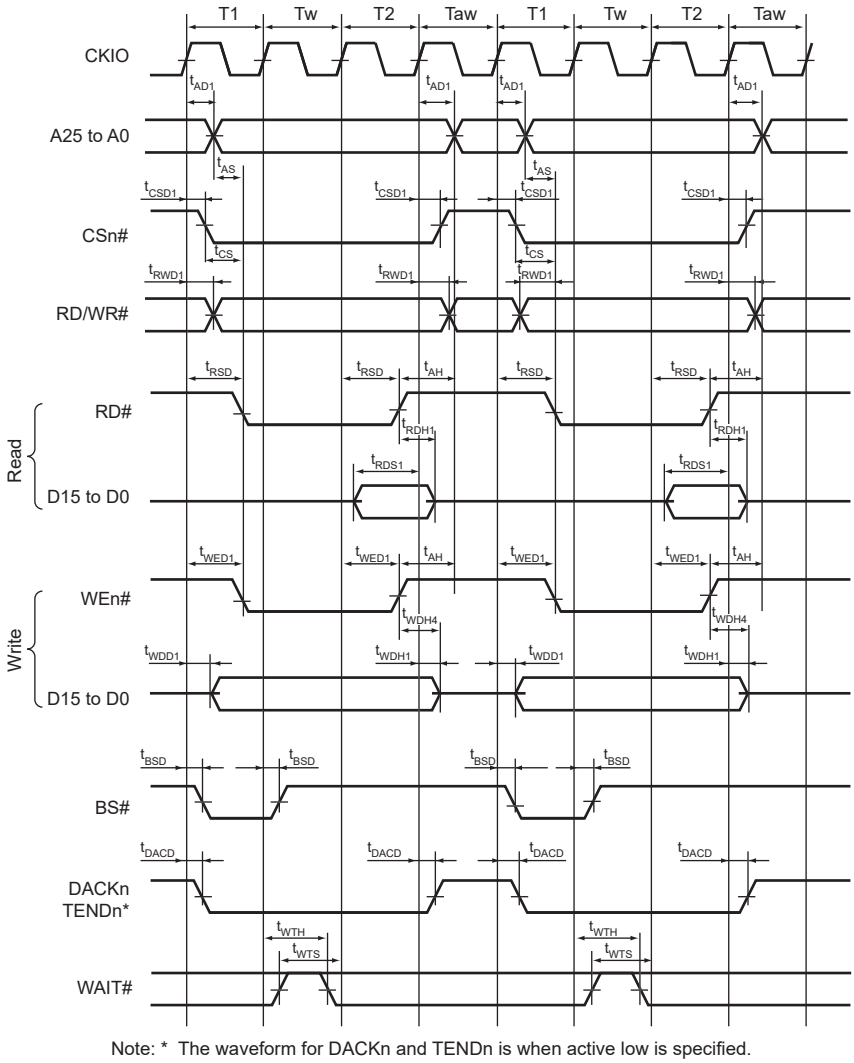
Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 56.14 Basic Bus Timing for Normal Space (One Software Wait Cycle)



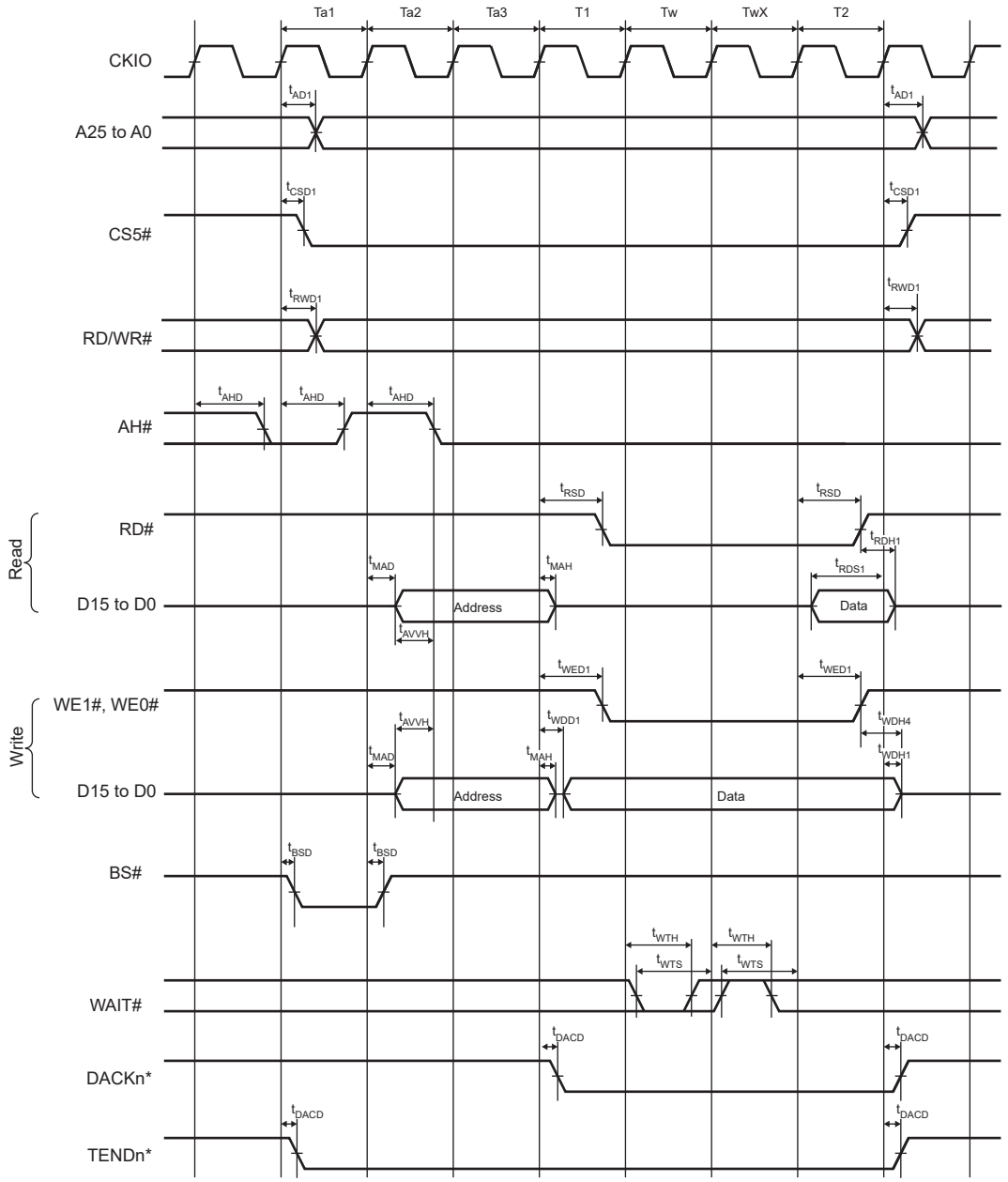
Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 56.15 Basic Bus Timing for Normal Space (One Software Wait Cycle, One External Wait Cycle)



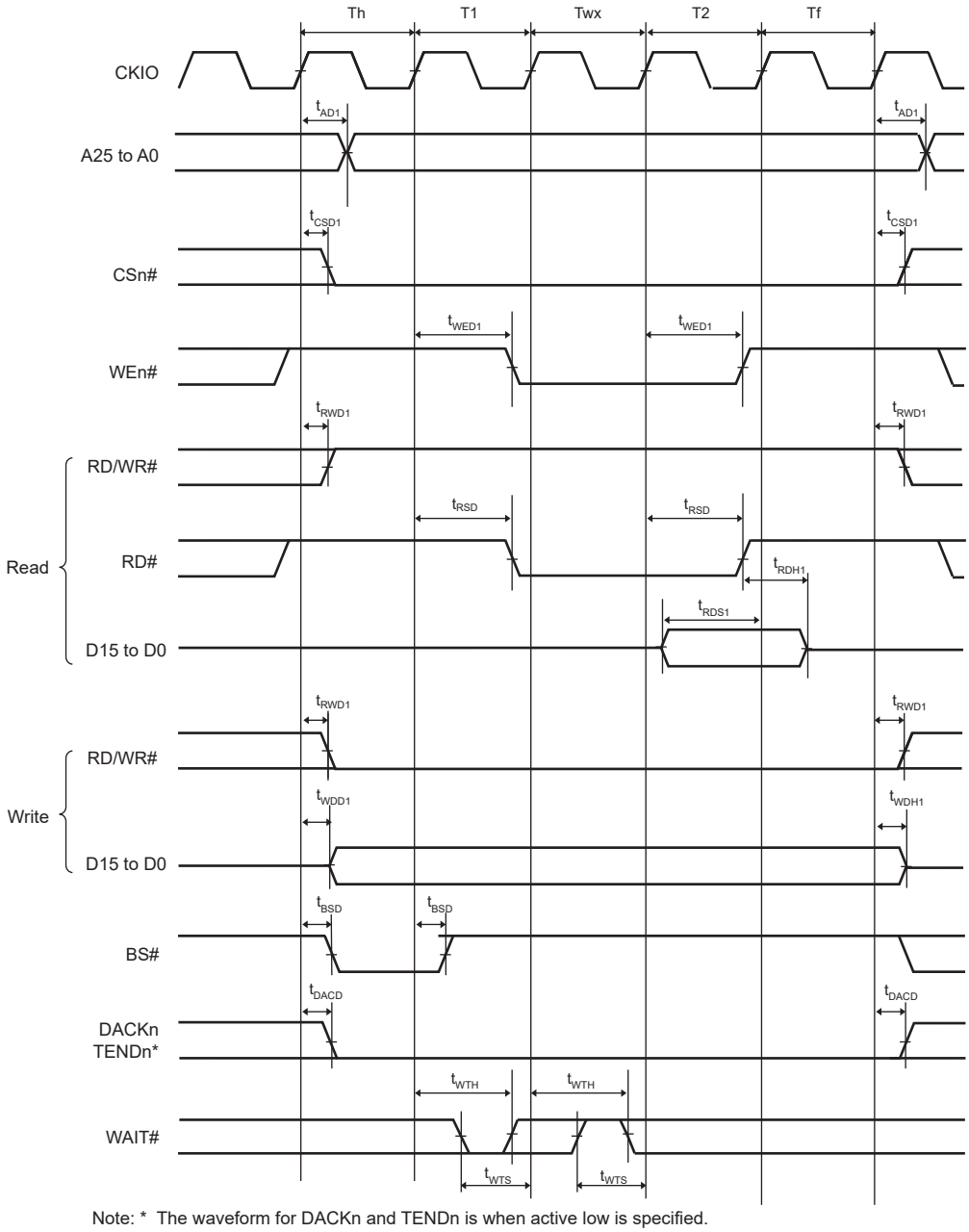
Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 56.16 Basic Bus Timing for Normal Space (One Software Wait Cycle, External Wait Cycle Valid (WM Bit = 0), No Idle Cycle)



Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 56.17 MPX-I/O Interface Bus Cycle (Three Address Cycles, One Software Wait Cycle, One External Wait Cycle)



Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 56.18 Bus Cycle of SRAM with Byte Selection (SW = 1 Cycle, HW = 1 Cycle, One Asynchronous External Wait Cycle, BAS = 0 (Write Cycle UB#/LB# Control))

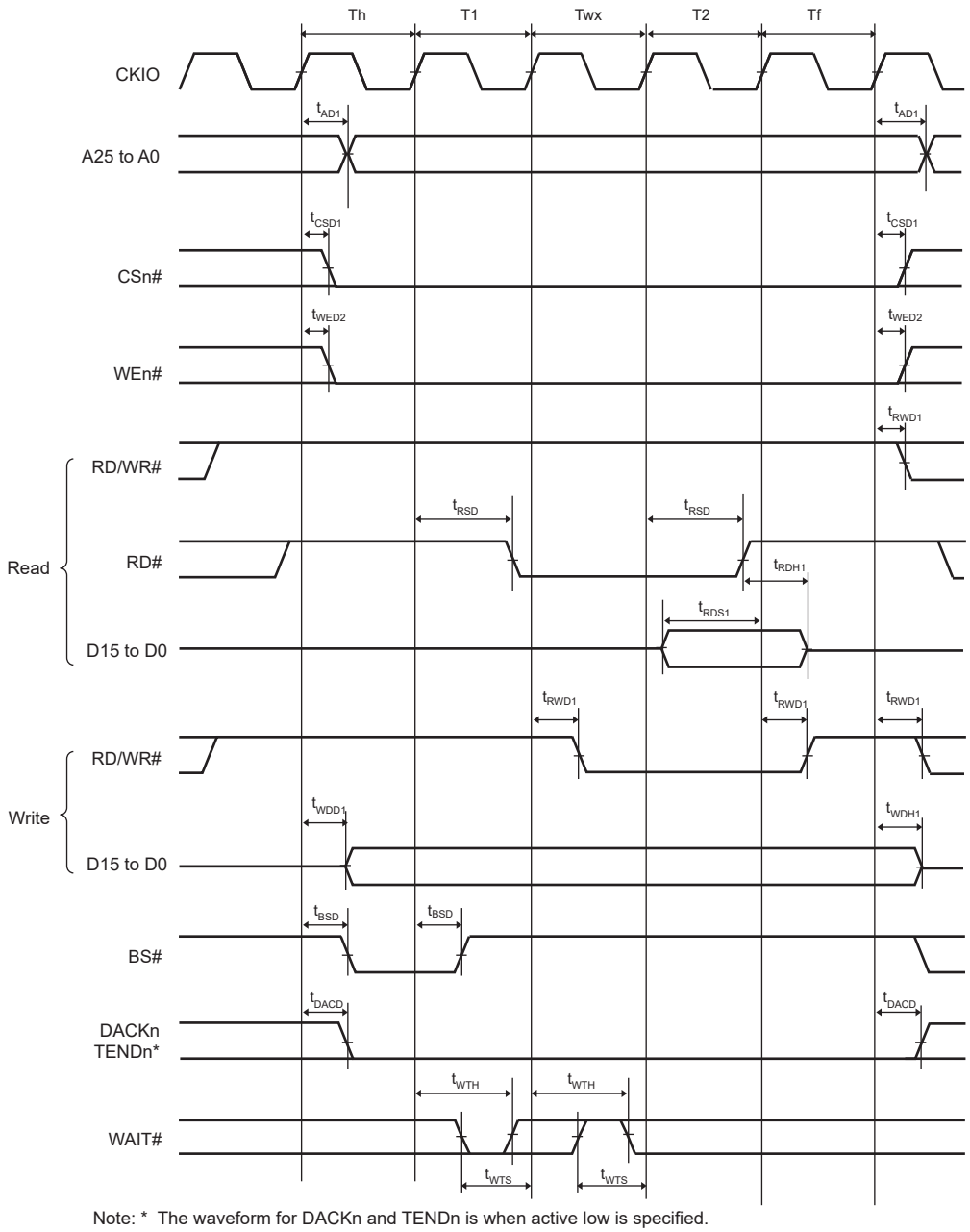
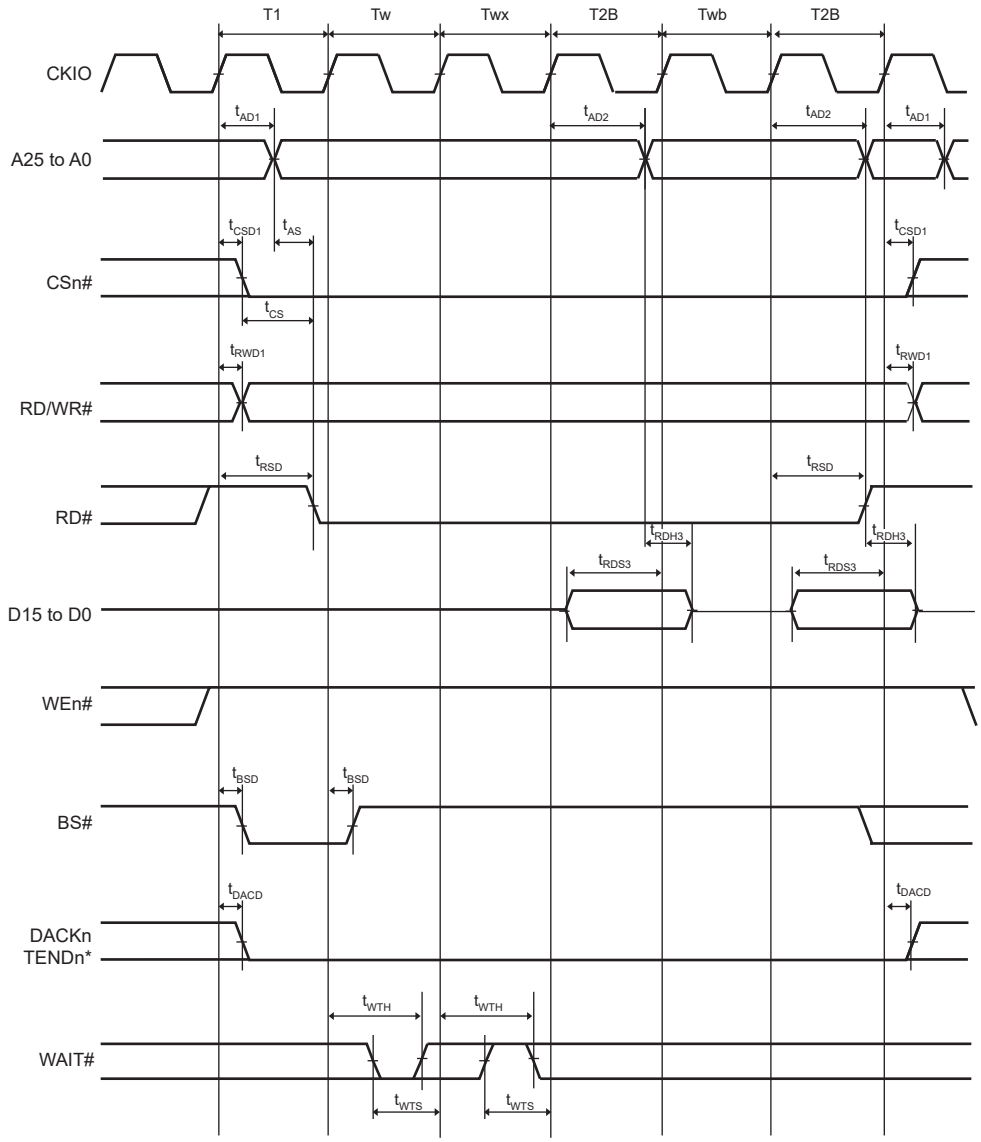
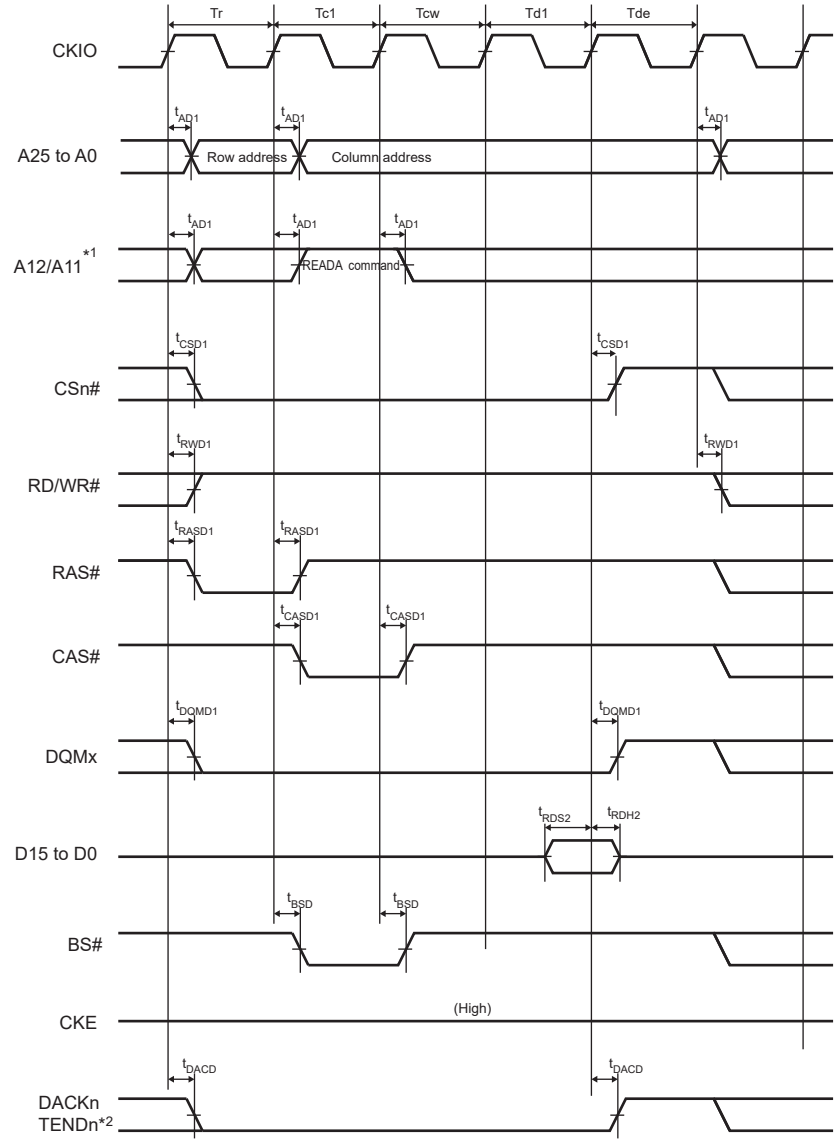


Figure 56.19 Bus Cycle of SRAM with Byte Selection (SW = 1 Cycle, HW = 1 Cycle, One Asynchronous External Wait Cycle, BAS = 1 (Write Cycle WE# Control))



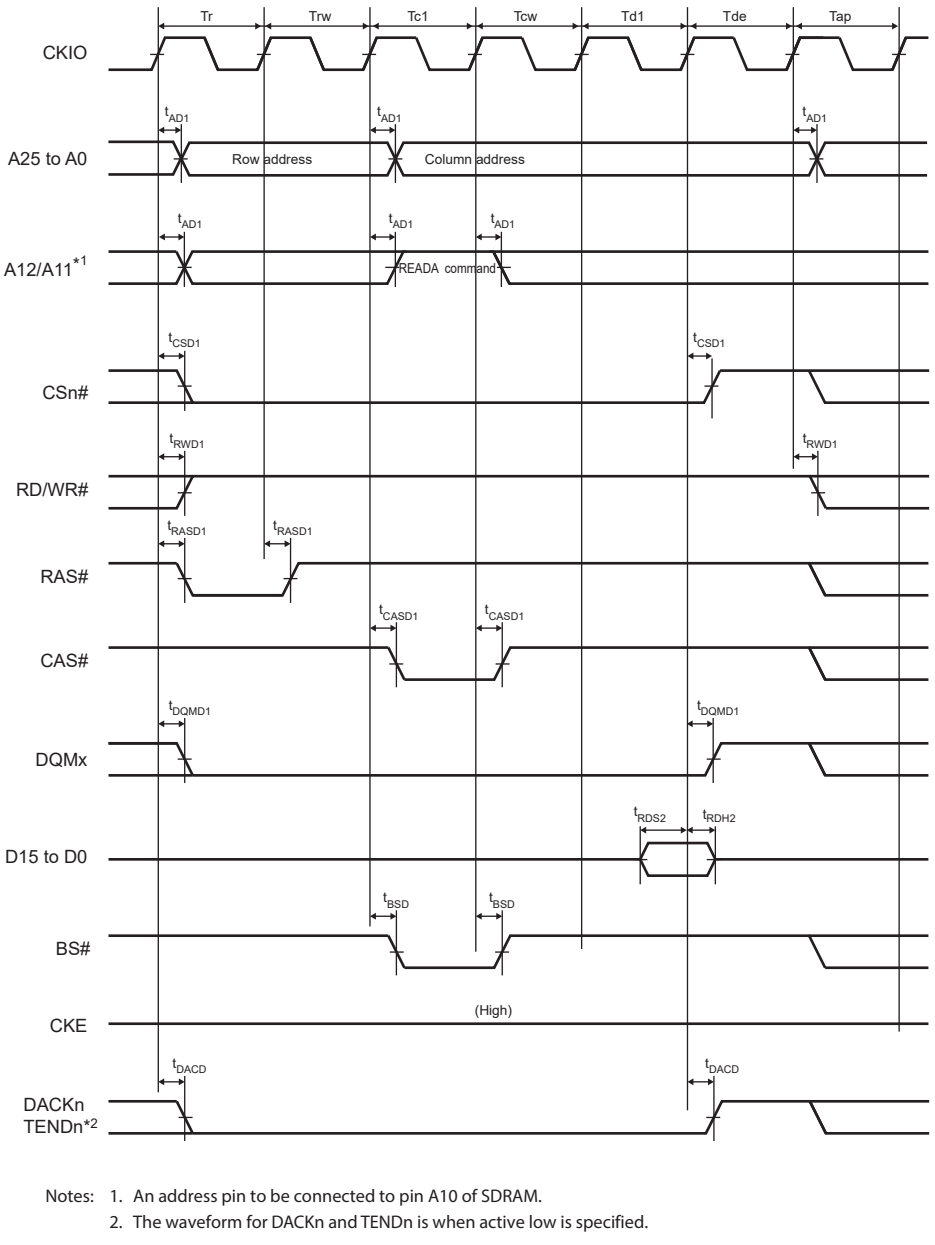
Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 56.20 Burst ROM Read Cycle (One Software Wait Cycle, One Asynchronous External Burst Wait Cycle, Two Burst)



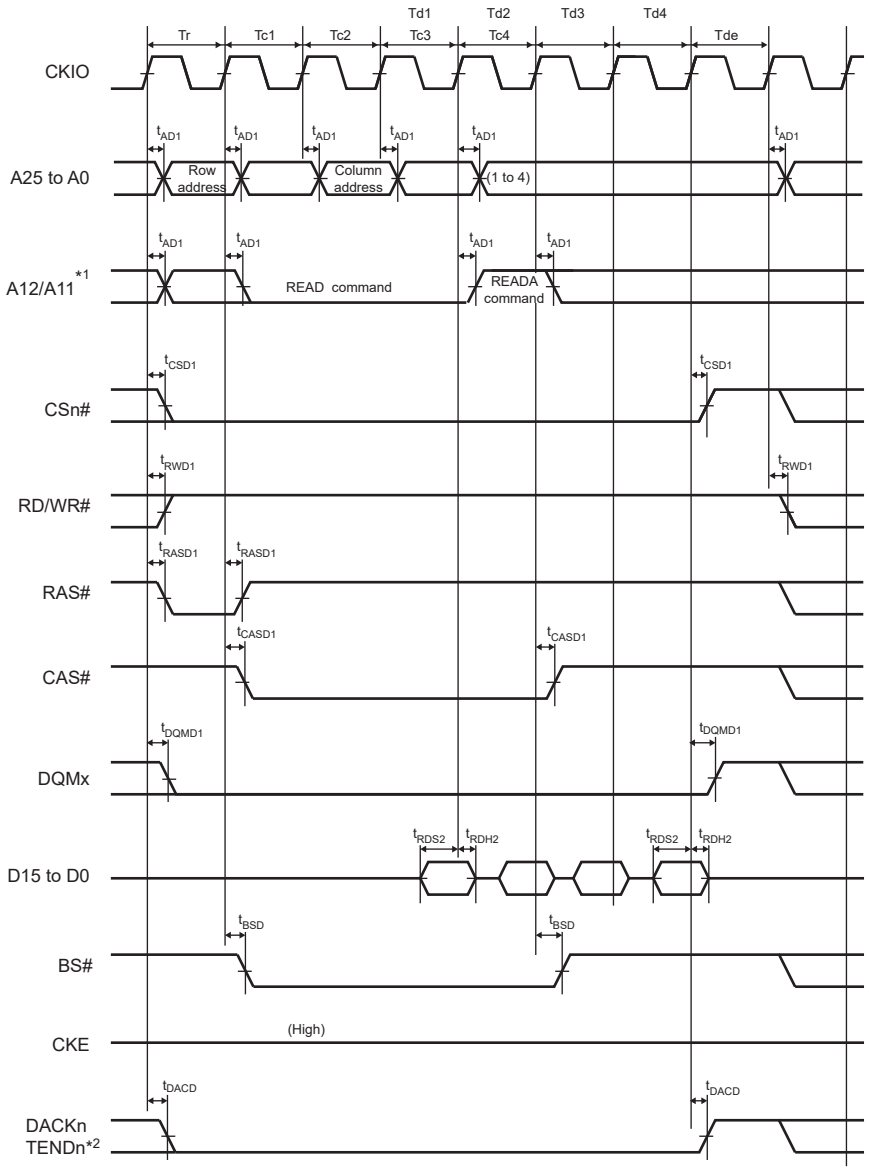
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.21 Synchronous DRAM Single Read Bus Cycle (Auto Precharge, CAS Latency 2, WTRCD = 0 Cycle, WTRP = 0 Cycle)



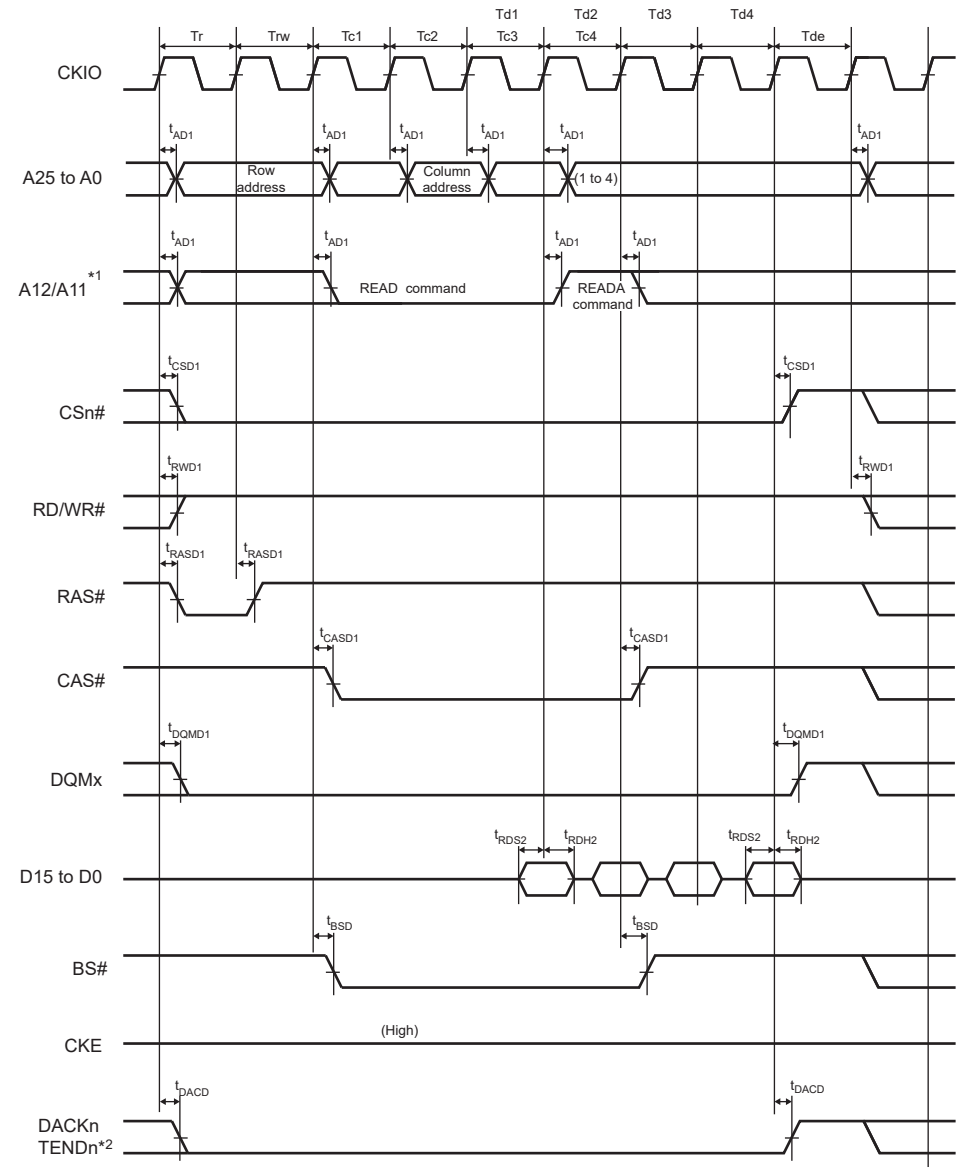
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.22 Synchronous DRAM Single Read Bus Cycle (Auto Precharge, CAS Latency 2, WTRCD = 1 Cycle, WTRP = 1 Cycle)



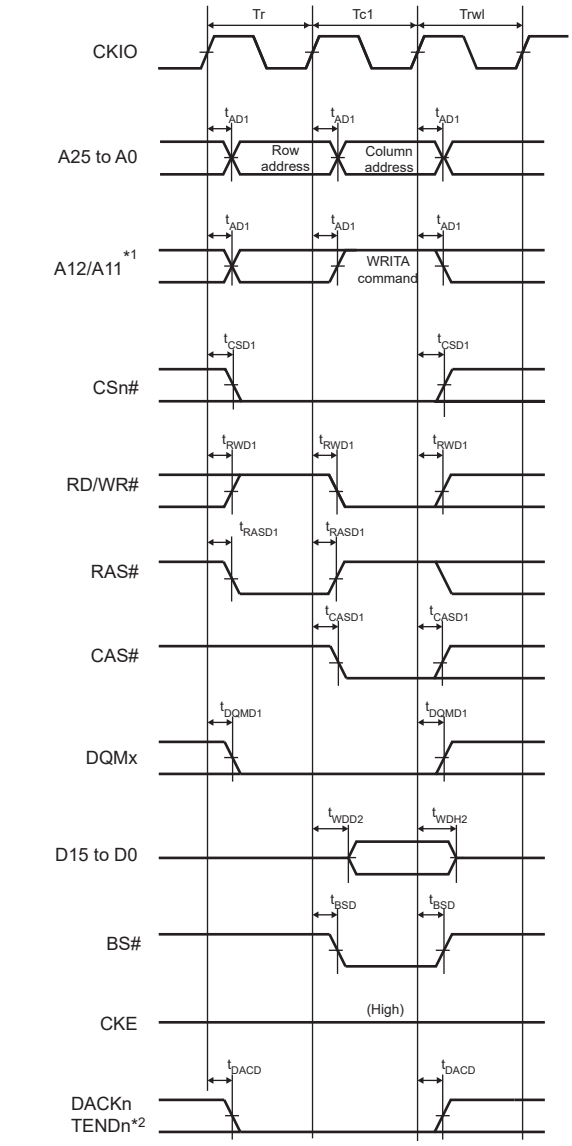
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.23 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Auto Precharge, CAS Latency 2, WTRCD = 0 Cycle, WTRP = 1 Cycle)



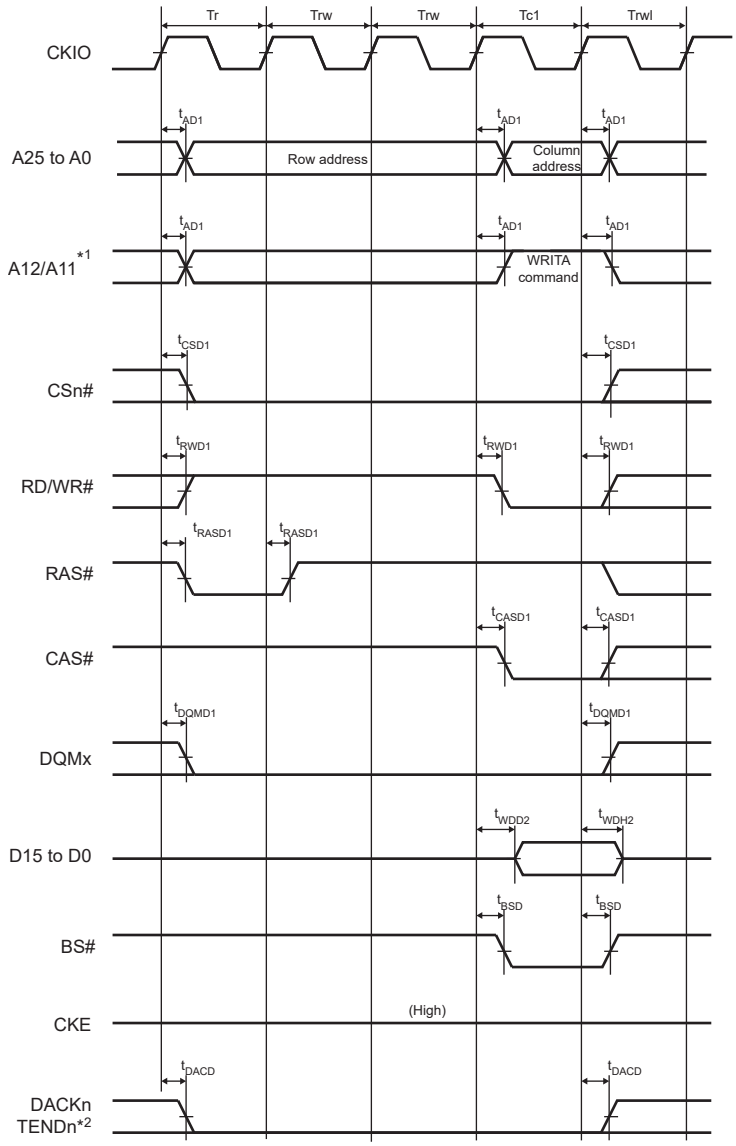
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.24 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Auto Precharge, CAS Latency 2, WTRCD = 1 Cycle, WTRP = 0 Cycle)



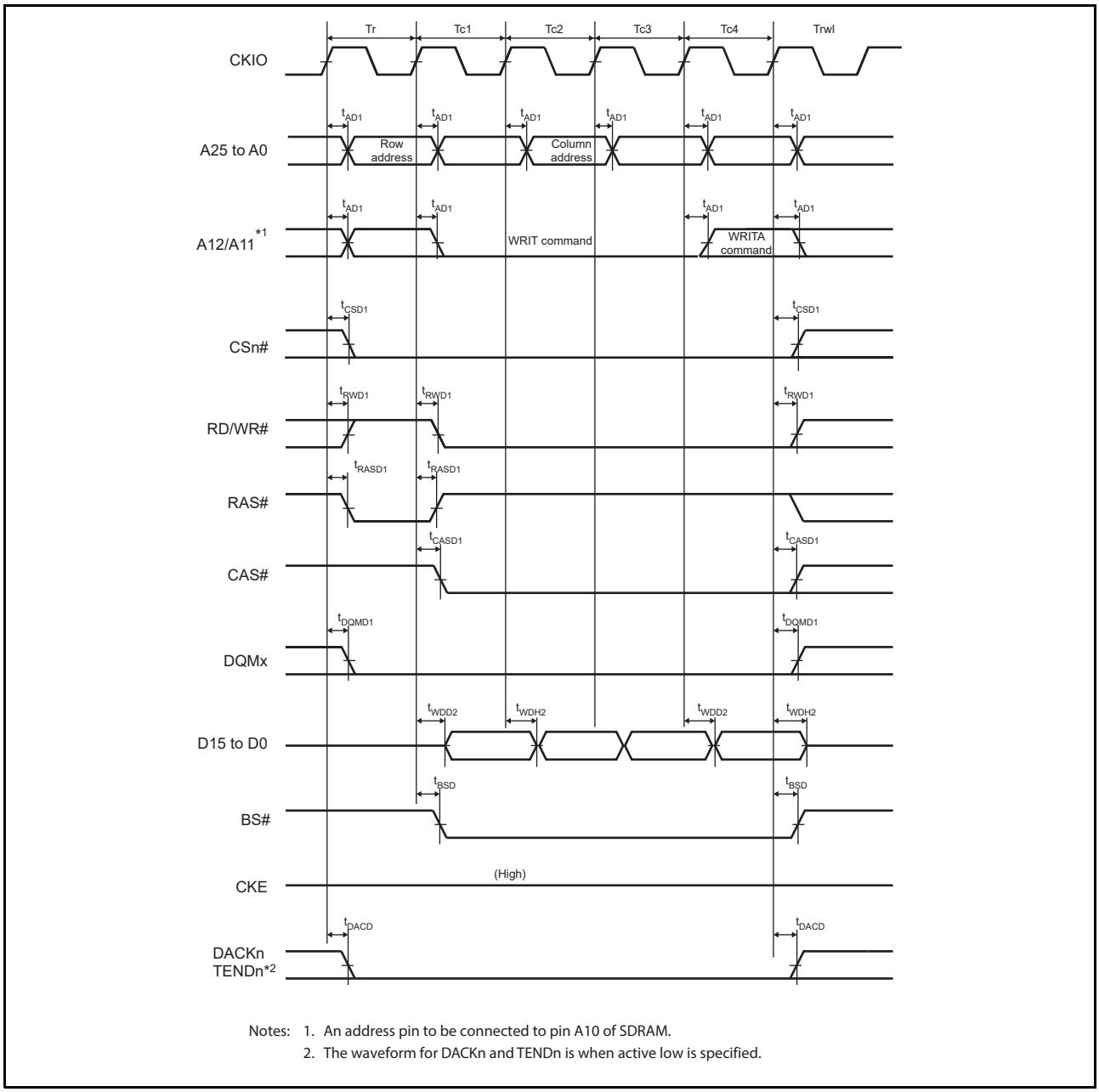
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.25 Synchronous DRAM Single Write Bus Cycle (Auto Precharge, TRWL = 1 Cycle)



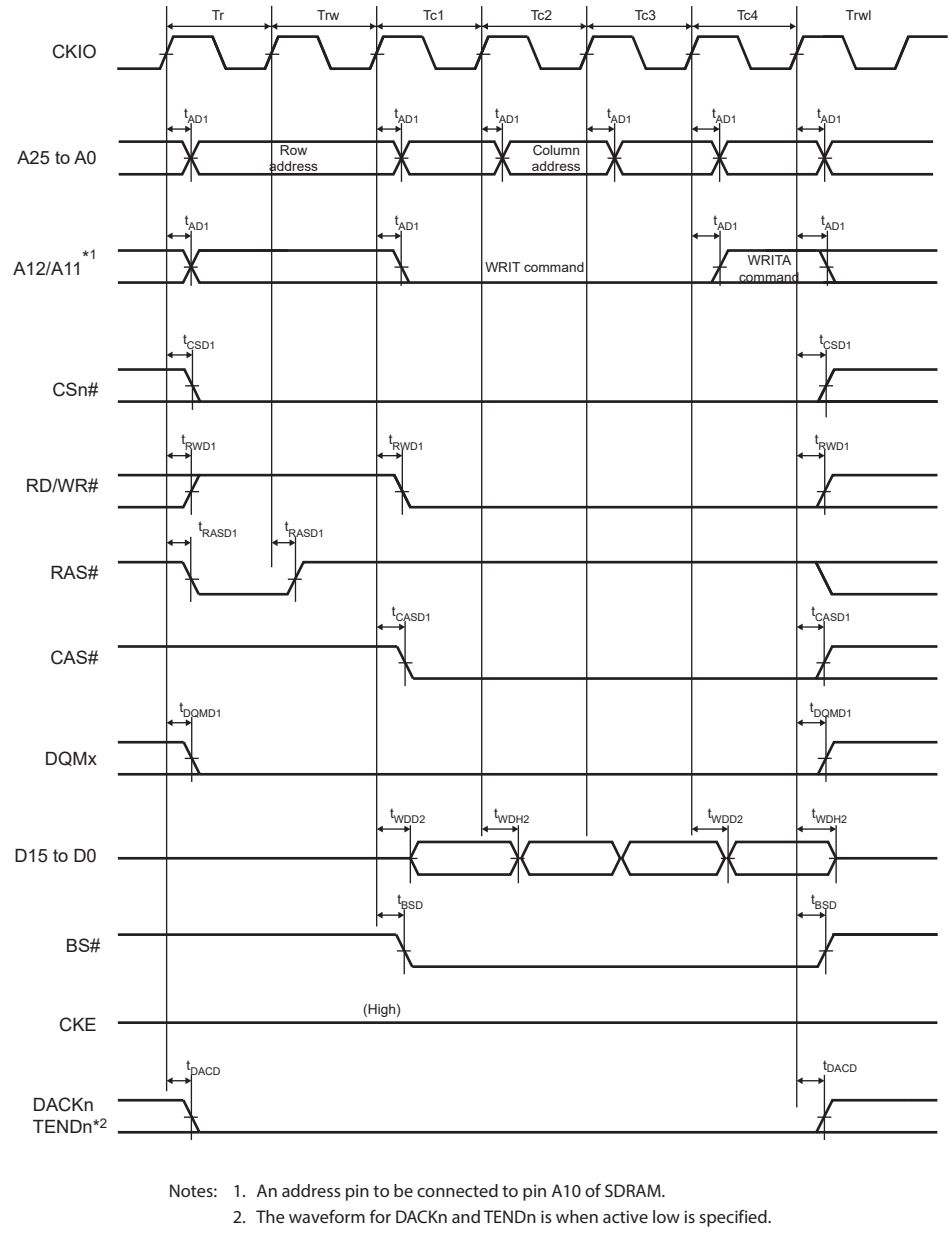
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.26 Synchronous DRAM Single Write Bus Cycle (Auto Precharge, WTRCD = 2 Cycles, TRWL = 1 Cycle)



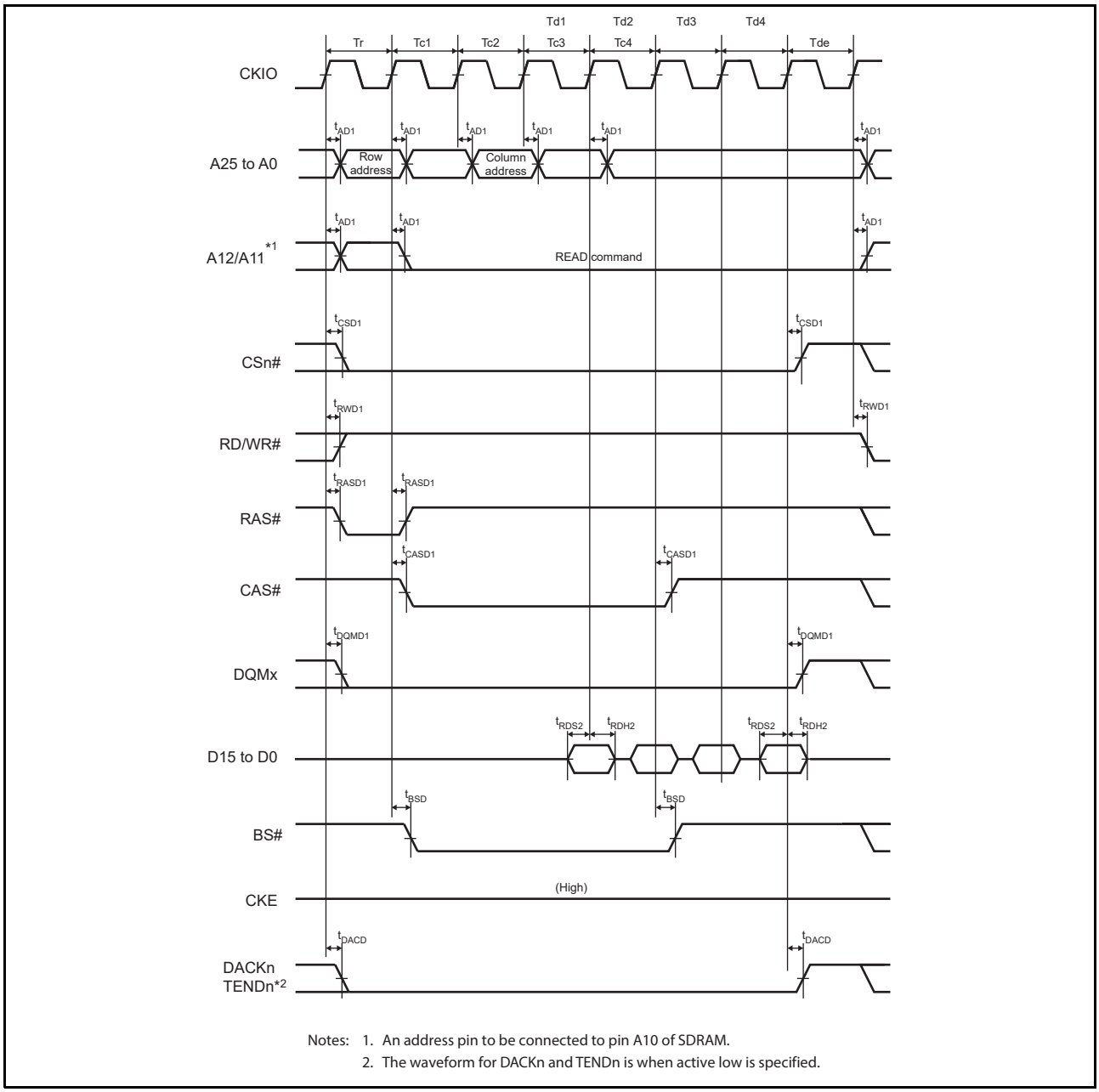
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.27 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Auto Precharge, WTRCD = 0 Cycle, TRWL = 1 Cycle)



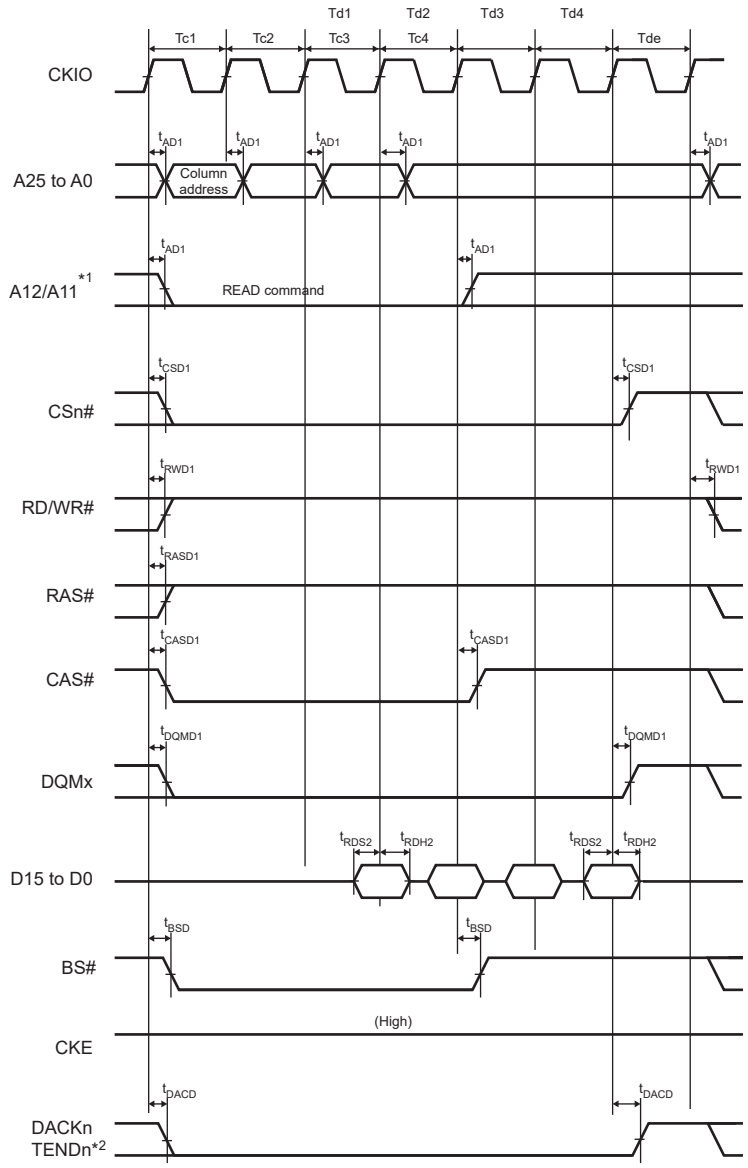
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.28 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Auto Precharge, WTRCD = 1 Cycle, TRWL = 1 Cycle)



Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.29 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Bank Active Mode: ACT + READ Commands, CAS Latency 2, WTRCD = 0 Cycle)



Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.30 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Bank Active Mode: READ Command, Same Row Address, CAS Latency 2, WTRCD = 0 Cycle)

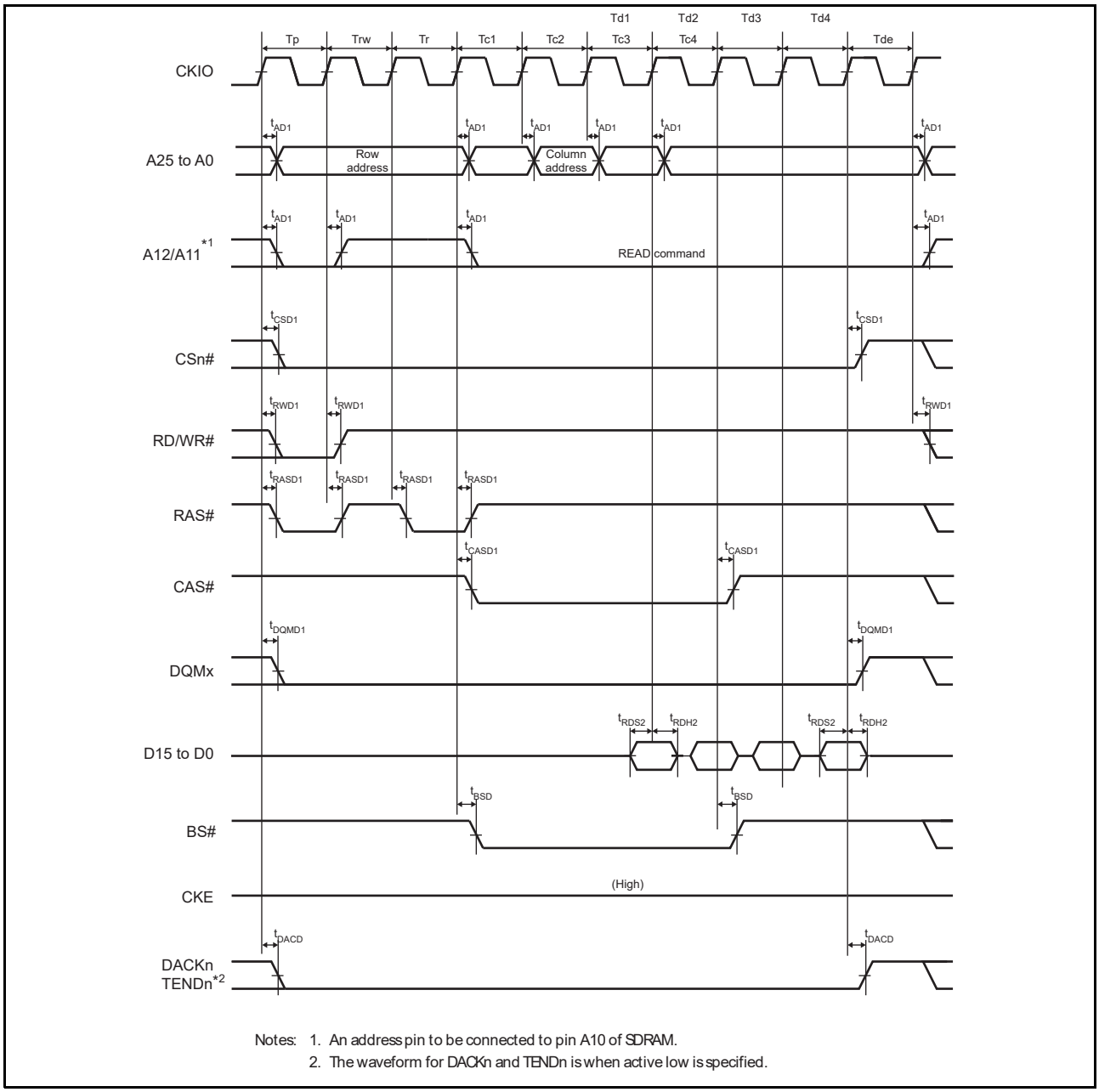
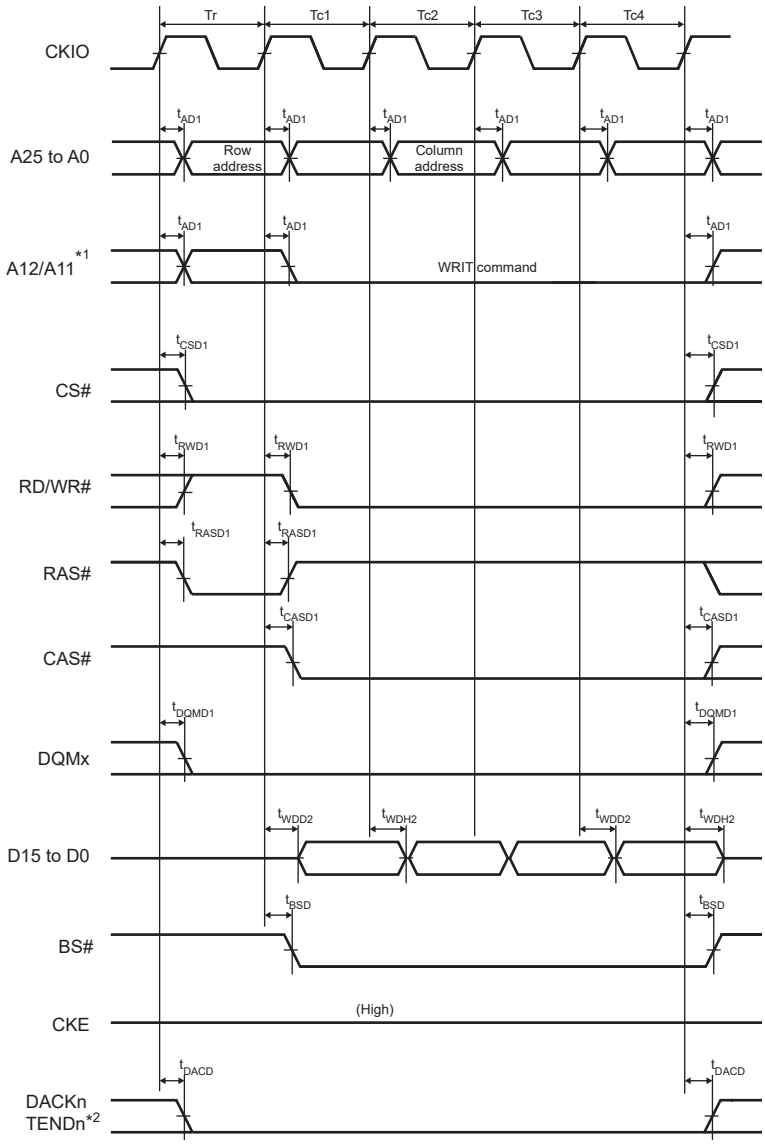
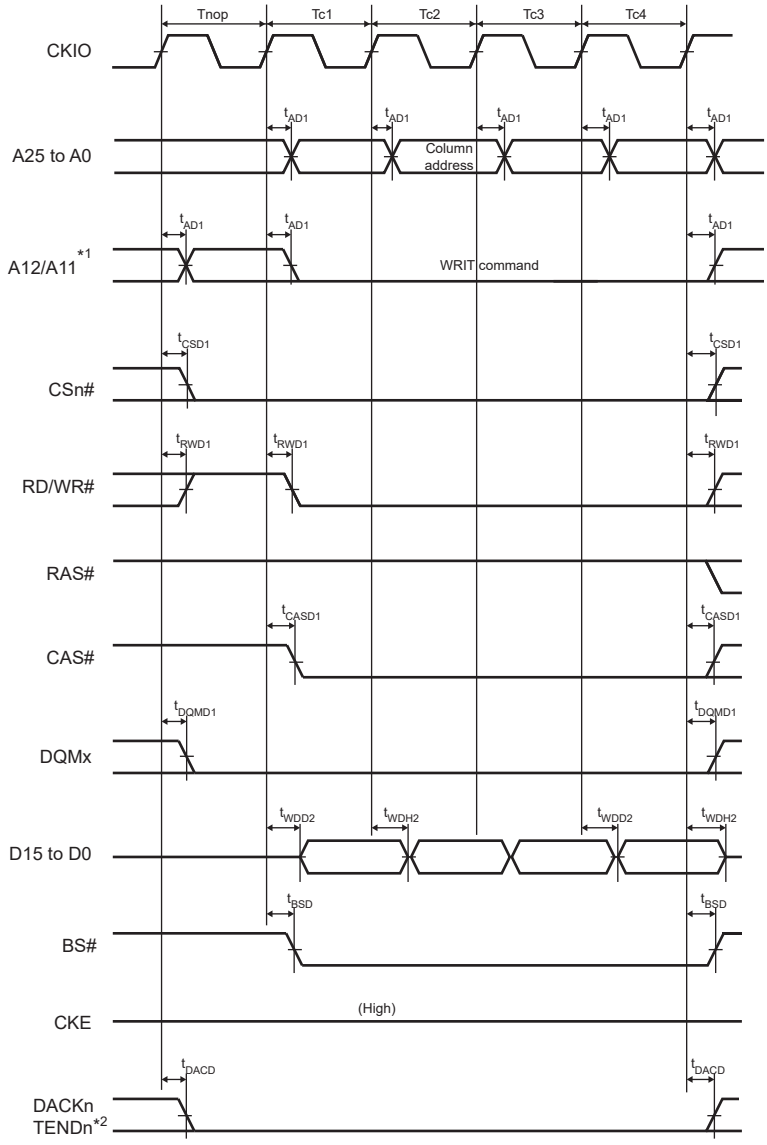


Figure 56.31 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Bank Active Mode: PRE + ACT + READ Commands, Different Row Addresses, CAS Latency 2, WTRCD = 0 Cycle)



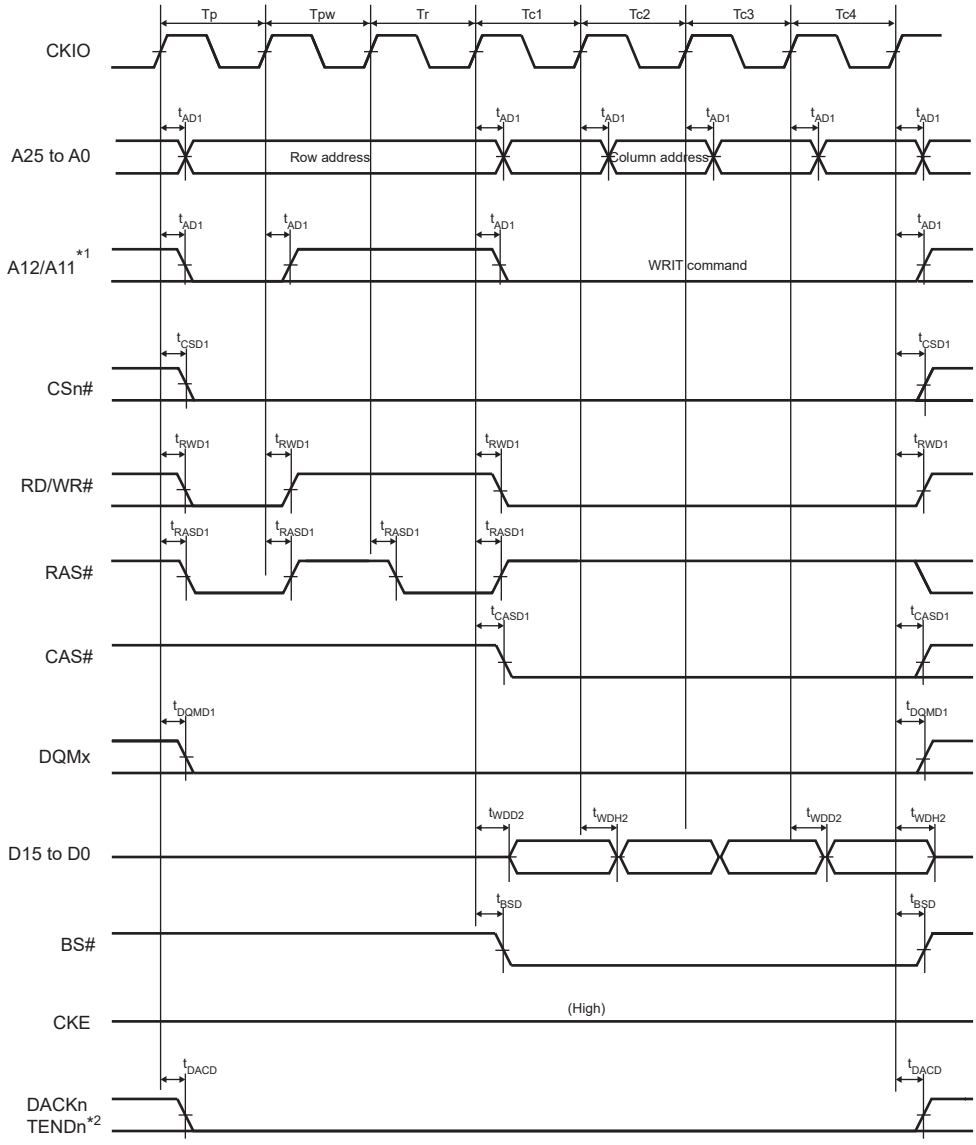
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.32 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Bank Active Mode: ACT + WRITE Commands, WTRCD = 0 Cycle, TRWL = 0 Cycle)



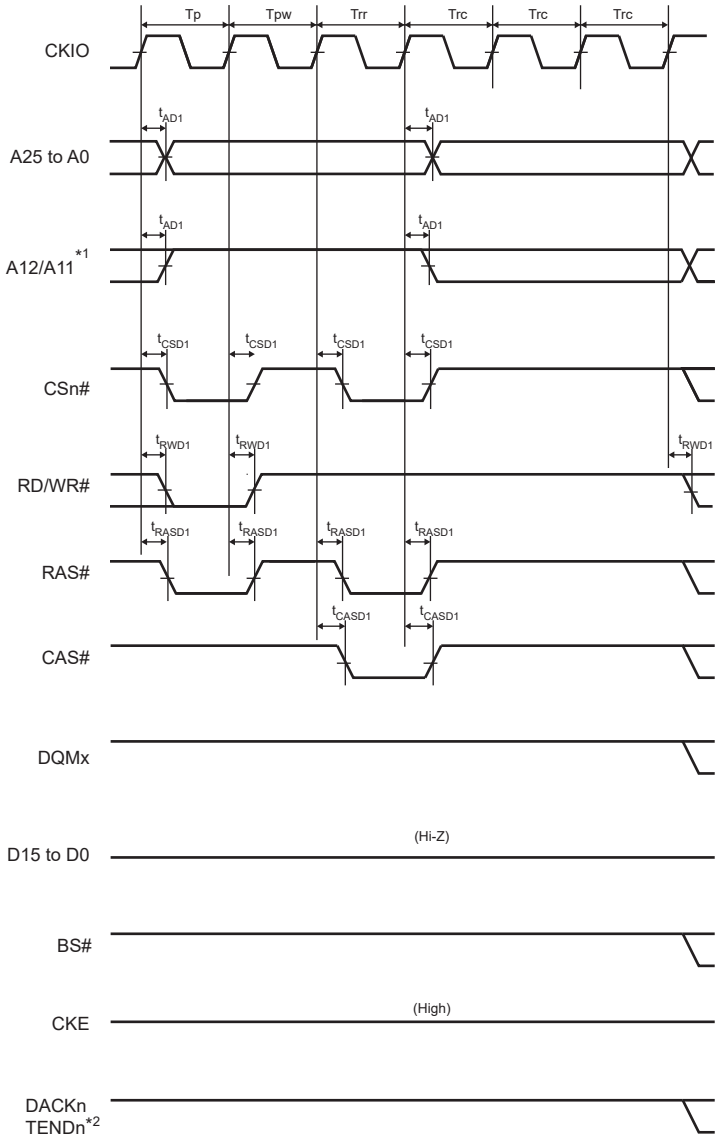
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.33 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Bank Active Mode: WRITE Command, Same Row Address, WTRCD = 0 Cycle, TRWL = 0 Cycle)



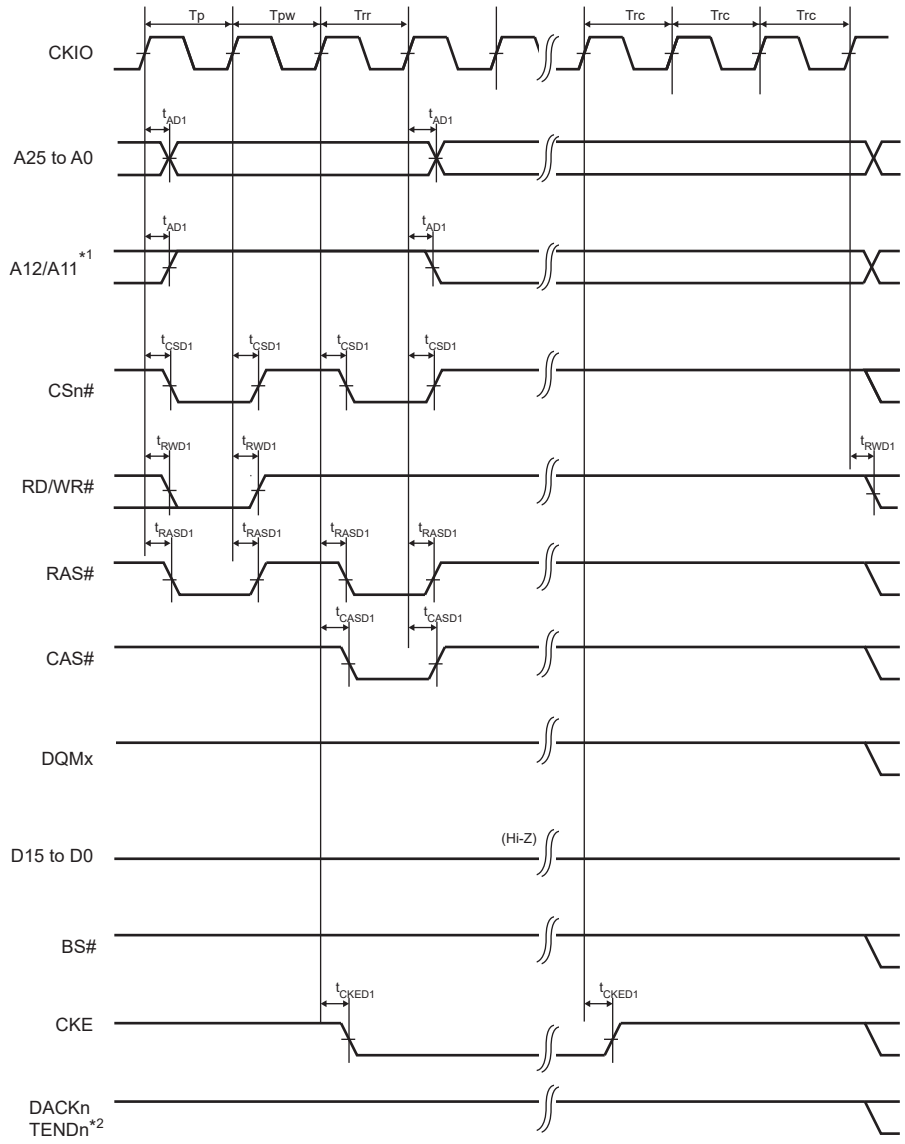
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.34 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Bank Active Mode: PRE + ACT + WRITE Commands, Different Row Addresses, WTRCD = 0 Cycle, TRWL = 0 Cycle)



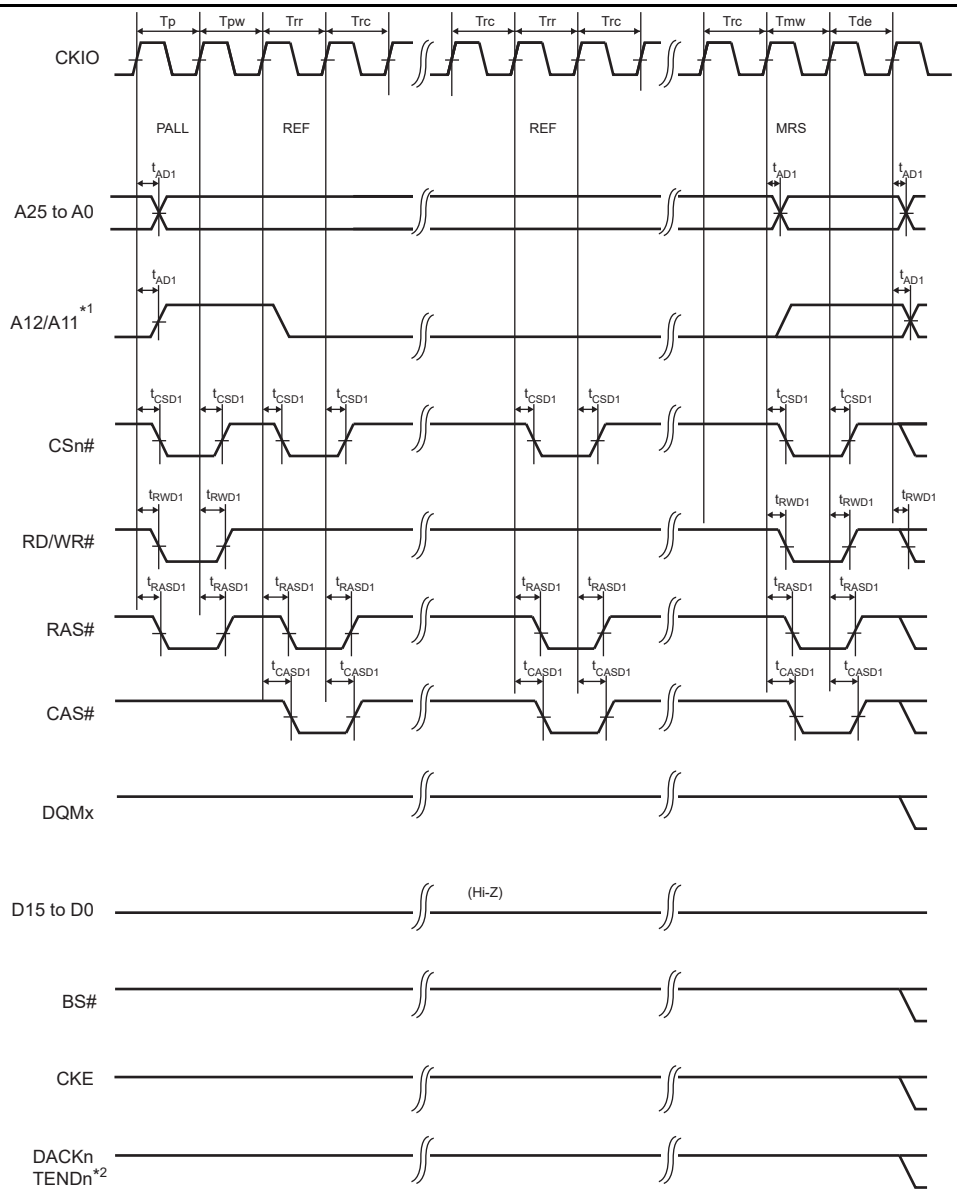
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.35 Synchronous DRAM Auto-Refreshing Timing (WTRP = 1 Cycle, WTRC = 3 Cycles)



Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.36 Synchronous DRAM Self-Refreshing Timing (WTRP = 1 Cycle)



Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 56.37 Synchronous DRAM Mode Register Write Timing (WTRP = 1 Cycle)

56.4.5 Direct Memory Access Controller Timing

Table 56.9 Direct Memory Access Controller Timing

Item	Symbol	Min.	Max.	Unit	Figure
DREQ setup time	t_{DRQS}	4.0	—	ns	Figure 56.38
DREQ hold time	t_{DRQH}	0.5	—		
DACK, TEND delay time	t_{DACD}	0	12		Figure 56.39

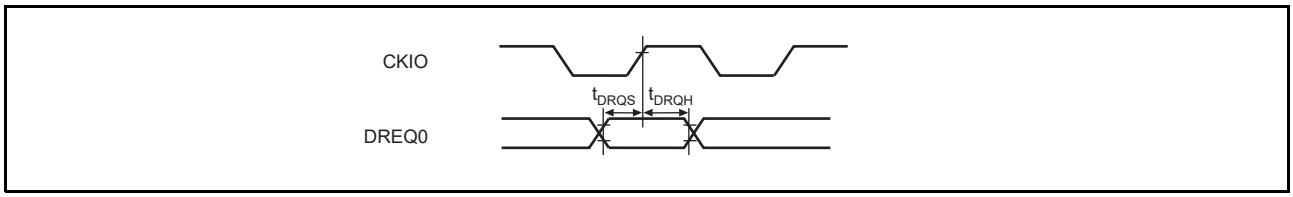


Figure 56.38 DREQ Input Timing

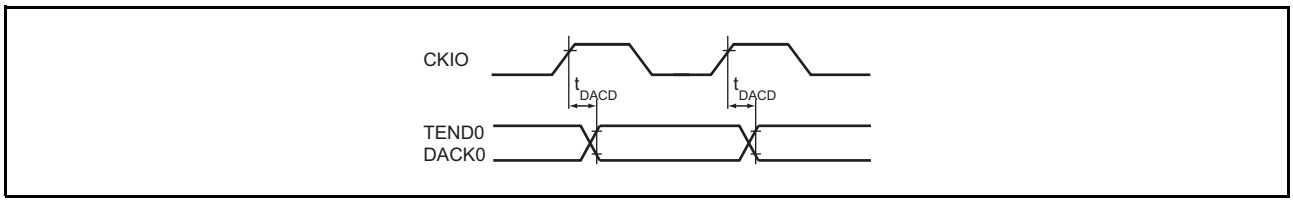


Figure 56.39 DACK, TEND Output Timing

56.4.6 Multi-Function Timer Pulse Unit 3 (MTU3a) Timing

Table 56.10 MTU3a Timing

Item			Symbol	Min.	Max.	Unit*1	Test Conditions
MTU3a	Input capture input pulse width	Single-edge setting	t_{MTICW}	1.5	—	t_{p1cyc}	Figure 56.40
		Both-edge setting		2.5	—		
	Timer clock pulse width	Single-edge setting	t_{MTCKWH}, t_{MTCKWL}	1.5	—	t_{p1cyc}	Figure 56.41
		Both-edge setting		2.5	—		
		Phase counting mode		2.5	—		

Note 1. t_{p1cyc} indicates peripheral clock 1C (P1φ) cycle.

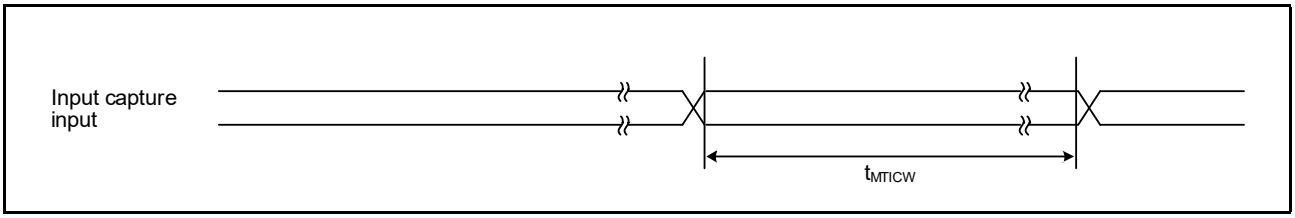


Figure 56.40 MTU3a Input Capture Input Timing

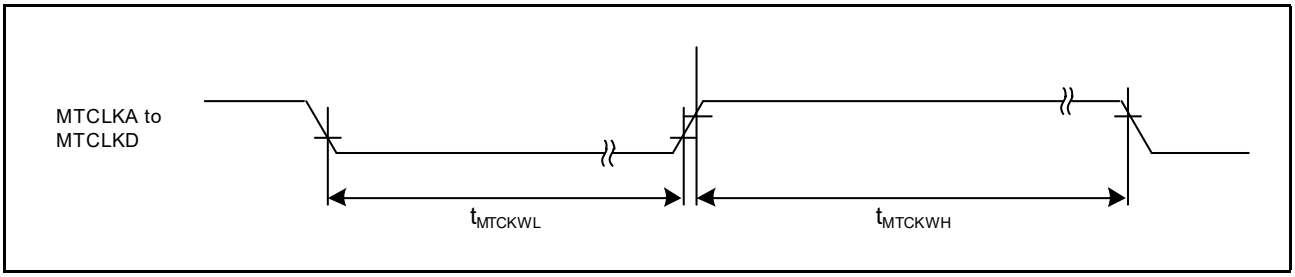


Figure 56.41 MTU3a Clock Input Timing

56.4.7 Port Output Enable 3 (POE3) Timing

Table 56.11 POE3 Timing

Item		Symbol	Min.	Max.	Unit*1	Test Conditions
POE3	POEn# input pulse width	t_{POE3W}	1.5	—	t_{p1cyc}	Figure 56.42

Note 1. t_{p1cyc} indicates peripheral clock 1C (P1φ) cycle.

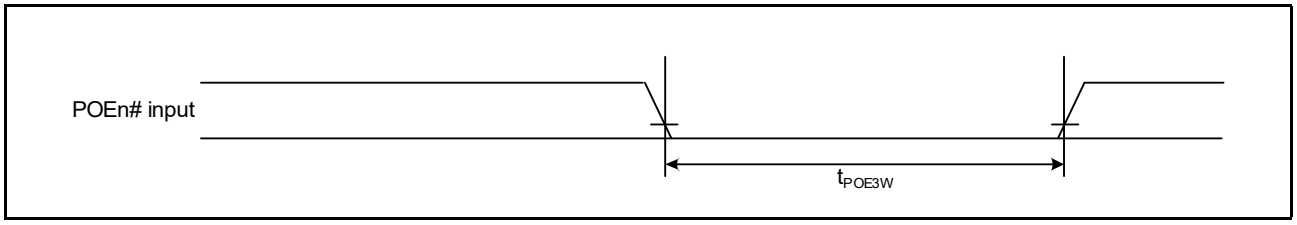


Figure 56.42 POEn# input pulse Timing

56.4.8 General PWM Timer (GPT) Timing

Table 56.12 GPT Timing

Item			Symbol	Min.	Max.	Unit*1	Test Conditions
GPT	Input capture input pulse width	Single-edge setting	t_{GTICW}	1.5	—	t_{p1cyc}	Figure 56.43
		Both-edge setting		2.5	—		

Note 1. t_{p1cyc} indicates peripheral clock 1C (P1φ) cycle.

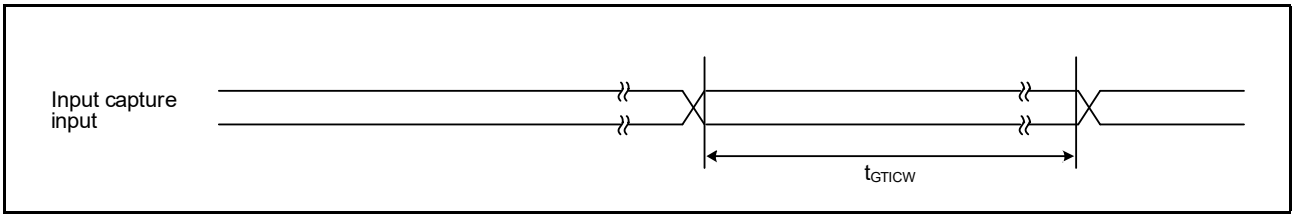


Figure 56.43 GPT Input Capture Input Timing

56.4.9 Port Output Enable for GPT (POEG) Timing

Table 56.13 POEG Timing

Item		Symbol	Min.	Max.	Unit*1	Test Conditions
POEG	POEG input trigger pulse width	t_{POEGW}	3	—	t_{p1cyc}	Figure 56.44

Note 1. t_{p1cyc} indicates peripheral clock 1C (P1φ) cycle.

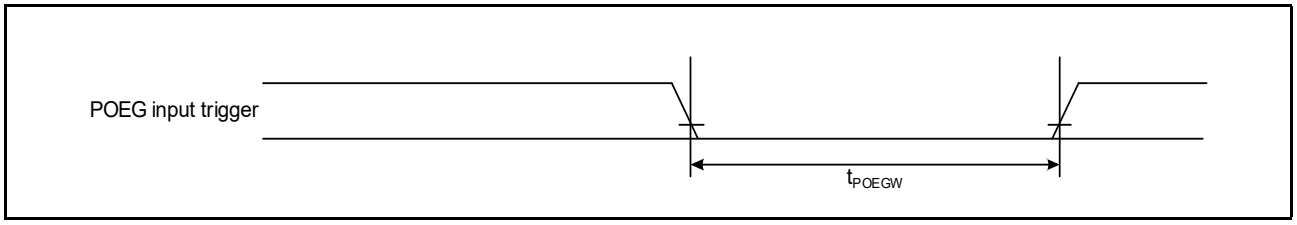


Figure 56.44 POEG Input Trigger Timing

56.4.10 Watchdog Timer Timing

Table 56.14 Watchdog Timer Timing

Item	Symbol	Min.	Max.	Unit	Figure
WDTOVF# delay time	t_{WOVD}	—	100	ns	Figure 56.45
PERROUT# delay time	t_{PEOD}	—	100	ns	

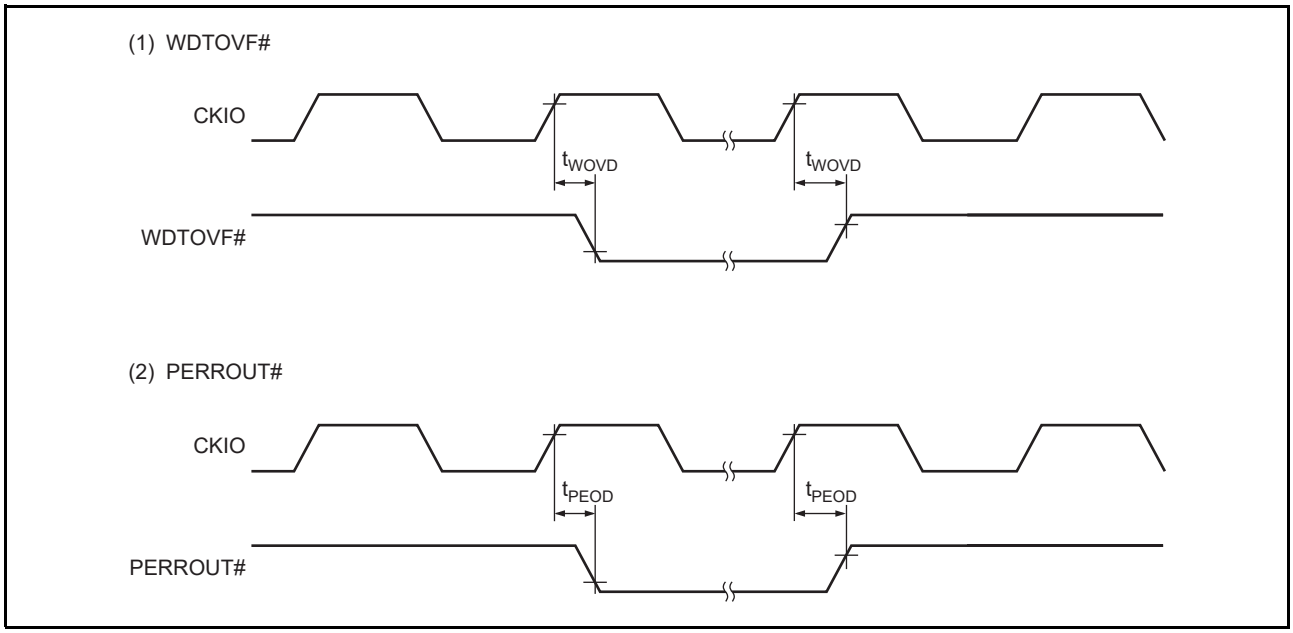


Figure 56.45 Watchdog Timer Output Timing

56.4.11 Serial Communications Interface with FIFO (SCIFA) Timing

Table 56.15 SCIFA Timing

Item			Symbol	Min.*1	Max.*1	Unit*1	Test Conditions
SCIFA	Input clock cycle	Asynchronous	t_{Scyc}	4	—	t_{p1cyc}	Figure 56.46
		Clocked synchronous		12	—		
	Input clock pulse width		t_{SCKW}	0.4	0.6	t_{Scyc}	
	Input clock rise time		t_{SCKr}	—	5	ns	
	Input clock fall time		t_{SCKf}	—	5	ns	
	Output clock cycle	Asynchronous*2	t_{Scyc}	8	—	t_{p1cyc}	
		Clocked synchronous		4	—		
	Output clock pulse width		t_{SCKW}	0.4	0.6	t_{Scyc}	
	Output clock rise time		t_{SCKr}	—	9	ns	
	Output clock fall time		t_{SCKf}	—	9	ns	
	Transmit data delay time	Internal clock	t_{TXD}	–10	10	ns	Figure 56.47
		External clock		$3 \times t_{p1cyc}$	$4 \times t_{p1cyc} + 20$		
	Receive data setup time	Internal clock	t_{RXS}	$3 \times t_{p1cyc} + 20$	—	ns	
		External clock		$t_{p1cyc} + 10$	—		
	Receive data hold time	Internal clock	t_{RXH}	$-3 \times t_{p1cyc}$	—	ns	
		External clock		$2 \times t_{p1cyc} + 10$	—		

Note 1. t_{p1cyc} indicates peripheral clock 1C (P1φ) cycle.
Note 2. When the SEMR.ABCS0 and SEMR.BGDM bits are set to 1

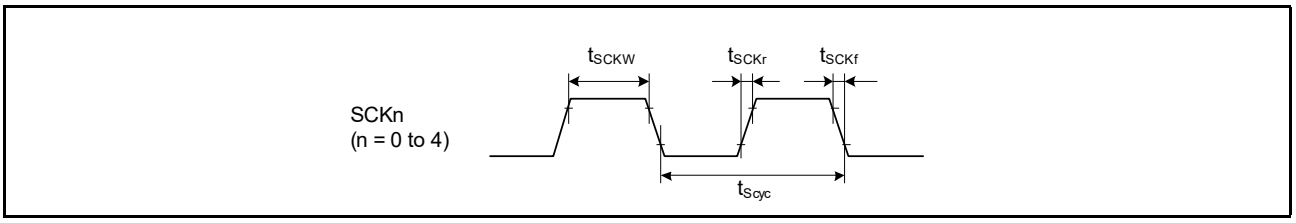


Figure 56.46 SCK Input Clock Timing

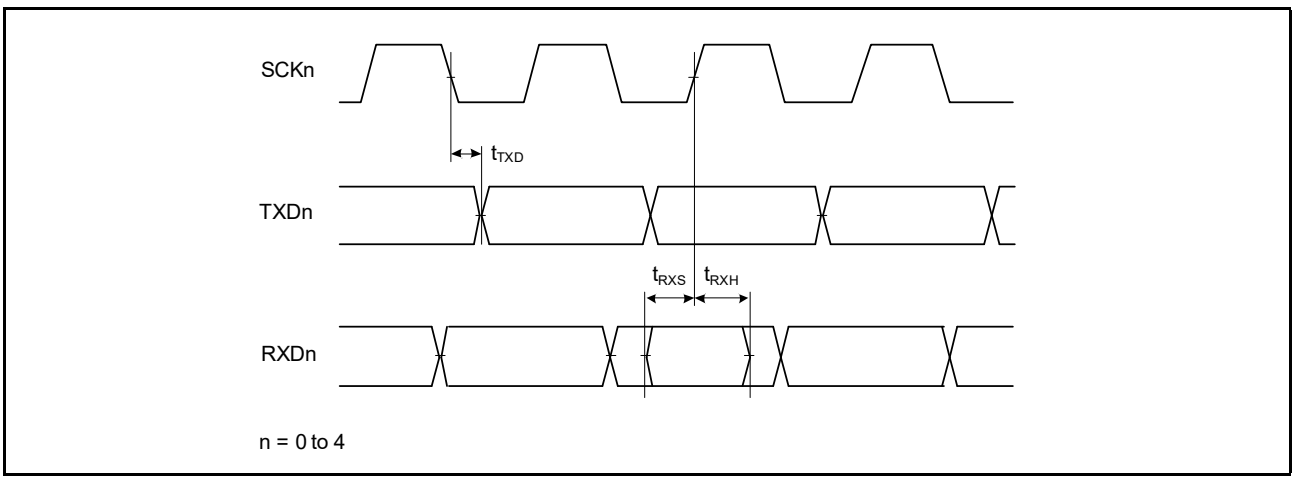


Figure 56.47 SCIFA Input/Output Timing in Clocked Synchronous Mode

56.4.12 Serial Communications Interface (SCI) Timing

Table 56.16 SCI Timing

Item			Symbol	Min.	Max.	Unit	Test Conditions
SCI	Input clock cycle	asynchronous	t_{Scyc}	4	—	t_{Pcyc} *1	Figure 56.48
		clocked synchronous		6	—		
	Input clock pulse width		t_{SCKW}	0.4	0.6	t_{Scyc}	
	Input clock rise time		t_{SCKr}	—	5	ns	
	Input clock fall time		t_{SCKf}	—	5	ns	
	Output clock cycle	asynchronous*2	t_{Scyc}	8	—	t_{Pcyc} *1	
		clocked synchronous		4	—		
	Output clock pulse width		t_{SCKW}	0.4	0.6	t_{Scyc}	
	Output clock rise time		t_{SCKr}	—	5	ns	
	Output clock fall time		t_{SCKf}	—	5	ns	
	Transmit data delay time	clocked synchronous	t_{TXD}	—	28	ns	Figure 56.49
	Receive data setup time	clocked synchronous	t_{RXS}	15	—	ns	
	Receive data hold time	clocked synchronous	t_{RXH}	5	—	ns	

Note 1. t_{Pcyc} indicates the peripheral clock 1 (P1φ) cycle.
Note 2. When the SEMR.ABCS and SEMR.BGDM bits are set to 1

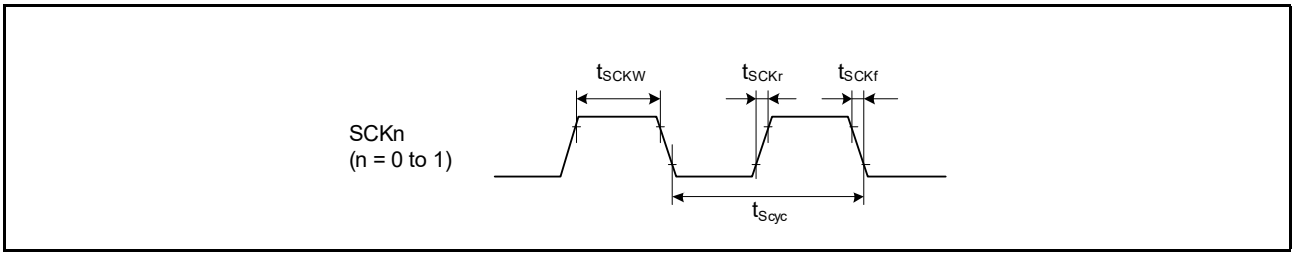


Figure 56.48 SCK Input Clock Timing

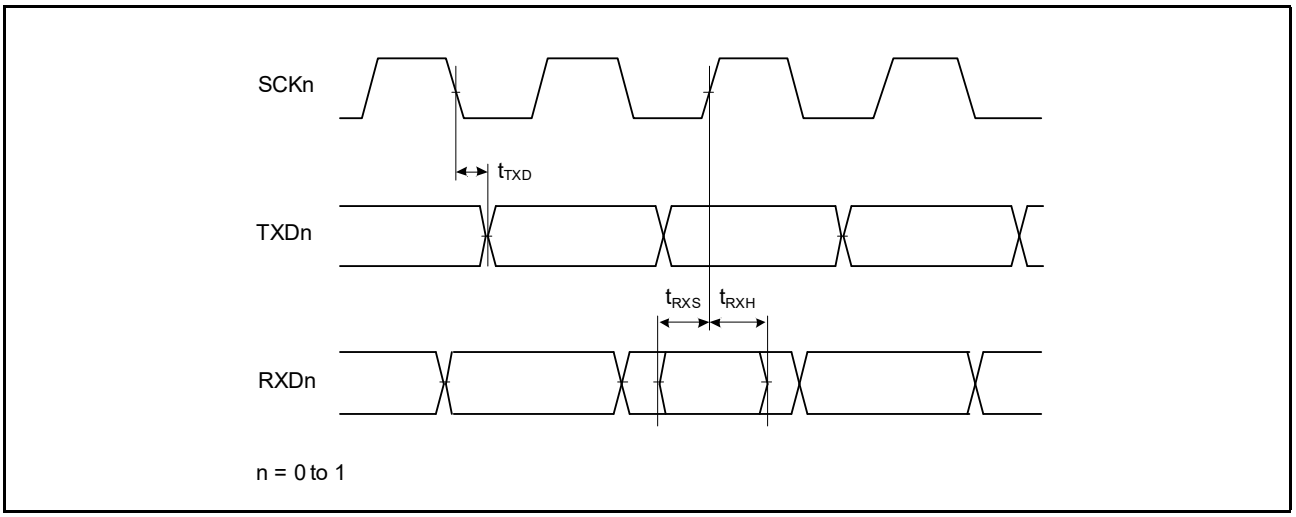


Figure 56.49 SCI Input/Output Timing in Clocked Synchronous Mode

56.4.13 Renesas Serial Peripheral Interface Timing

Table 56.17 Renesas Serial Peripheral Interface Timing

Item		Symbol	Min.	Max.	Unit	Figure
RSPCK clock cycle	Master	t_{SPcyc}	2	4096	t_{p1cyc}	Figure 56.50
	Slave		8	4096		
RSPCK clock high pulse width	Master	t_{SPCKWH}	0.4	—	t_{SPcyc}	
	Slave		0.4	—		
RSPCK clock low pulse width	Master	t_{SPCKWL}	0.4	—	t_{SPcyc}	
	Slave		0.4	—		
Data input setup time	Master	t_{SU}	15	—	ns	Figure 56.51 to Figure 56.54
	Slave		0	—	t_{p1cyc}	
Data input hold time	Master	t_H	0	—	ns	
	Slave		4	—	t_{p1cyc}	
SSL setup time	Master	t_{LEAD}	1 × t_{SPcyc} – 20	8 × t_{SPcyc}	ns	
	Slave		4	—	t_{p1cyc}	
SSL hold time	Master	t_{LAG}	1 × t_{SPcyc}	8 × t_{SPcyc} + 20	ns	
	Slave		4	—	t_{p1cyc}	
Data output delay time	Master	t_{OD}	—	21	ns	
	Slave		—	4	t_{p1cyc}	
Data output hold time	Master	t_{OH}	5	—	ns	
	Slave		3	—	t_{p1cyc}	
Continuous transmission delay time	Master	t_{TD}	1 × t_{SPcyc} + 2 × t_{cyc}	8 × t_{SPcyc} + 2 × tcyc	ns	
	Slave		4 × tcyc	—		
Slave access time		t_{SA}	—	4	t_{p1cyc}	Figure 56.53, Figure 56.54
Slave out release time		t_{REL}	—	3	t_{p1cyc}	

Note: tp1cyc indicates the period of a cycle of the peripheral clock 1 (P1φ).

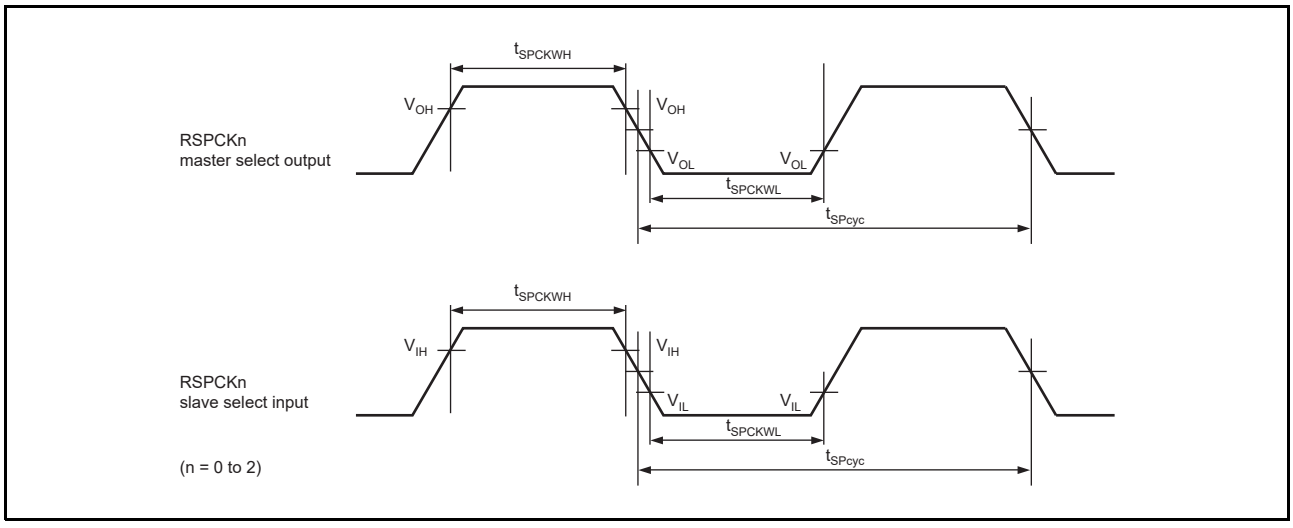


Figure 56.50 Clock Timing

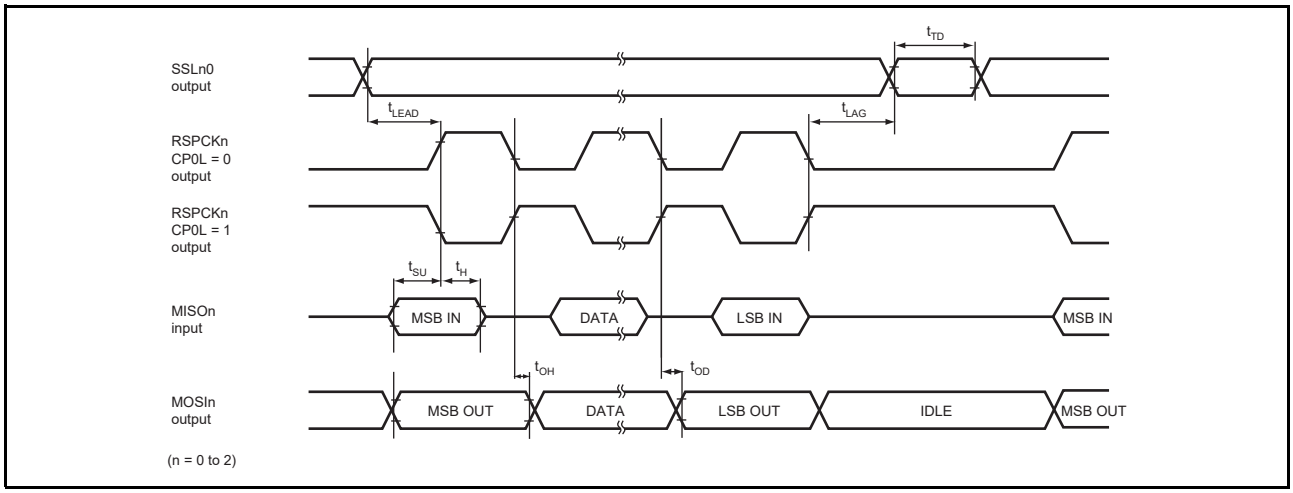


Figure 56.51 Transmission and Reception Timing (Master, CPHA = 0)

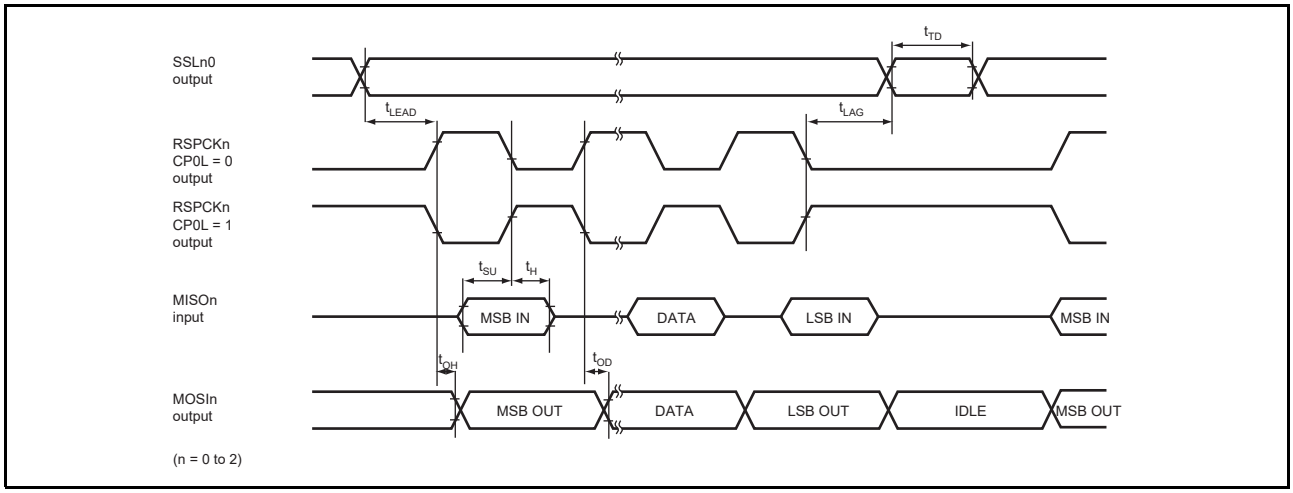


Figure 56.52 Transmission and Reception Timing (Master, CPHA = 1)

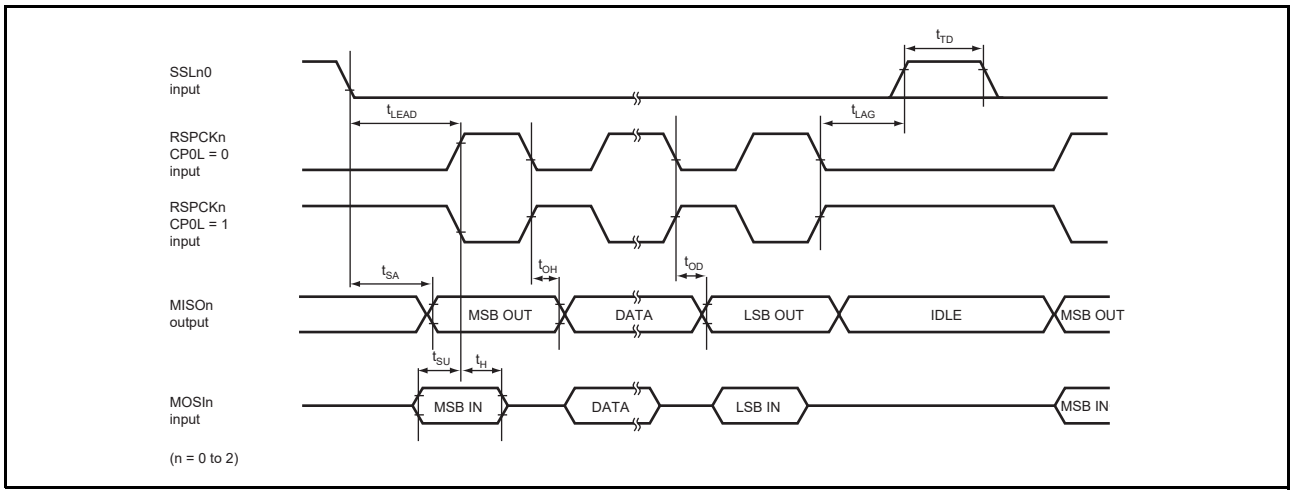


Figure 56.53 Transmission and Reception Timing (Slave, CPHA = 0)

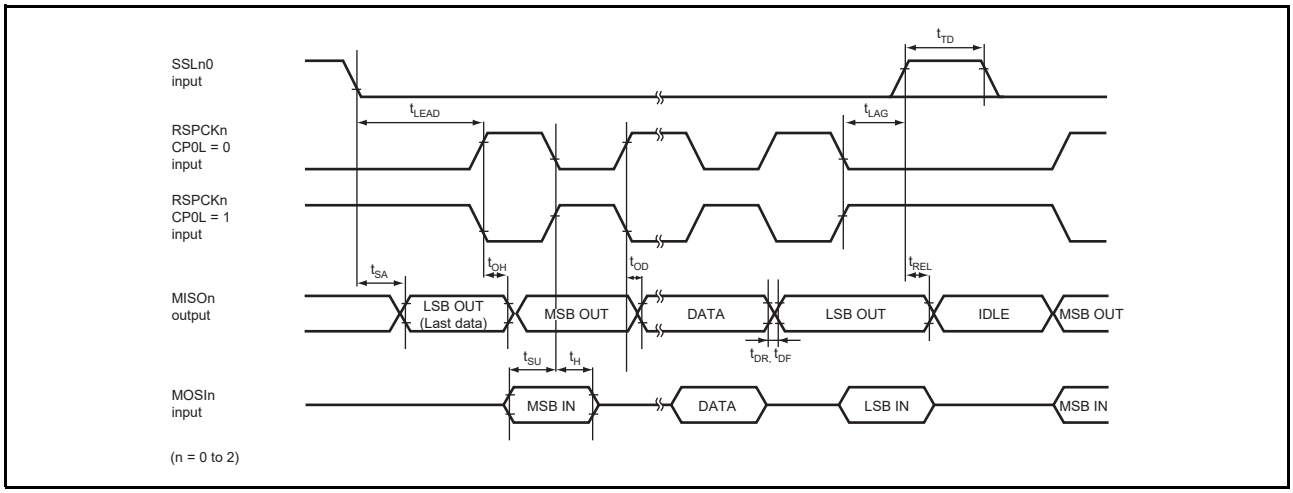


Figure 56.54 Transmission and Reception Timing (Slave, CPHA = 1)

56.4.14 SPI Multi I/O Bus Controller Timing

Table 56.18 SPI Multi I/O Bus Controller Timing *1

Item		Symbol	1.8 V (Octal-SPI flash memory/ HyperFlash connected)		3.3 V (Serial flash connected)		Unit	Figure
			Min.	Max.	Min.	Max.		
Clock cycle		t_{SPBcyc}	7.58	—	15.15	—	ns	Figure 56.55, Figure 56.59
CLK high pulse width		t_{SPBWH}	0.475	0.525	0.475	0.525	t_{SPBcyc}	Figure 56.55, Figure 56.59
CLK low pulse width		t_{SPBWL}	0.475	0.525	0.475	0.525	t_{SPBcyc}	Figure 56.55, Figure 56.59
CLK rise time		t_{SPBR}	—	1.0	—	2.0	ns	Figure 56.55, Figure 56.59
CLK fall time		t_{SPBF}	—	1.0	—	2.0	ns	Figure 56.55, Figure 56.59
Data input setup time	QSPI0_SPCLK base point (SDR mode timing adjusted)	t_{SU}	—	—	5.0	—	ns	Figure 56.56
	QSPI0_SPCLK base point (DDR mode timing adjusted)		—	—	2.0	—	ns	Figure 56.57
	QSPI1_SSL base point		– 0.9 *2	—	—	—	ns	Figure 56.60
Data input hold time	QSPI0_SPCLK base point (SDR mode timing adjusted)	t_H	—	—	0.0	—	ns	Figure 56.56
	QSPI0_SPCLK base point (DDR mode timing adjusted)		—	—	1.0	—	ns	Figure 56.57
	QSPI1_SSL base point		2.69 *2	—	—	—	ns	Figure 56.60
SSL setup time		t_{LEAD}	1.5 × tSPBcyc – 3	8.5 × t_{SPBcyc} + 3	1.5 × t_{SPBcyc} – 3	8.5 × t_{SPBcyc} + 3	ns	Figure 56.56, Figure 56.57, Figure 56.60
SSL hold time		t_{LAG}	1 × t_{SPBcyc} – 3	8 × t_{SPBcyc} + 3	1 × t_{SPBcyc} – 3	8 × t_{SPBcyc} + 3	ns	Figure 56.56, Figure 56.57, Figure 56.60
Continuous transfer delay time		t_{TD}	1 × t_{SPBcyc} – 3	8 × t_{SPBcyc} + 3	1 × t_{SPBcyc} – 3	8 × t_{SPBcyc} + 3	ns	Figure 56.56, Figure 56.57, Figure 56.60
Data output delay time	SDR	t_{OD}	—	1.4	—	5.0	ns	Figure 56.56
	DDR		—	2.69 *2	—	5.0 *3	ns	Figure 56.57, Figure 56.60
Data output hold time	SDR	t_{OH}	– 1.4	—	– 5.0	—	ns	Figure 56.56
	DDR		0.9 *2	—	2.1 *3	—	ns	Figure 56.57, Figure 56.60
Skew of Clock to Data Strobe		t_{CKDS}	—	7	—	—	ns	Figure 56.60
Data output buffer off time	SDR	t_{BOFF}	– 1.7	—	– 5.5	2	ns	Figure 56.58
	DDR		1 × t_{SPBcyc} – 1.7	—	1 × t_{SPBcyc} – 5.5	—	ns	

Item	Symbol	1.8 V (Octal-SPI flash memory/ HyperFlash connected)		3.3 V (Serial flash connected)		Unit	Figure
		Min.	Max.	Min.	Max.		
AC differential crossing voltage	V_{OX}	PVcc_SPI × 0.4	PVcc_SPI × 0.6	—	—	V	Figure 56.59

Note 1. Output load: 15 pF
Note 2. QSPI0_SPCLK frequency: 132 MHz
Note 3. QSPI0_SPCLK frequency: 66 MHz

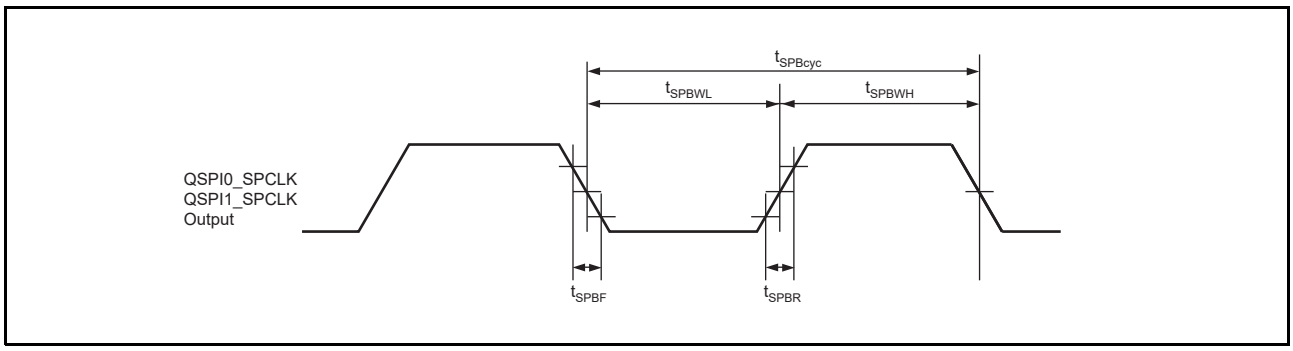


Figure 56.55 Clock Timing

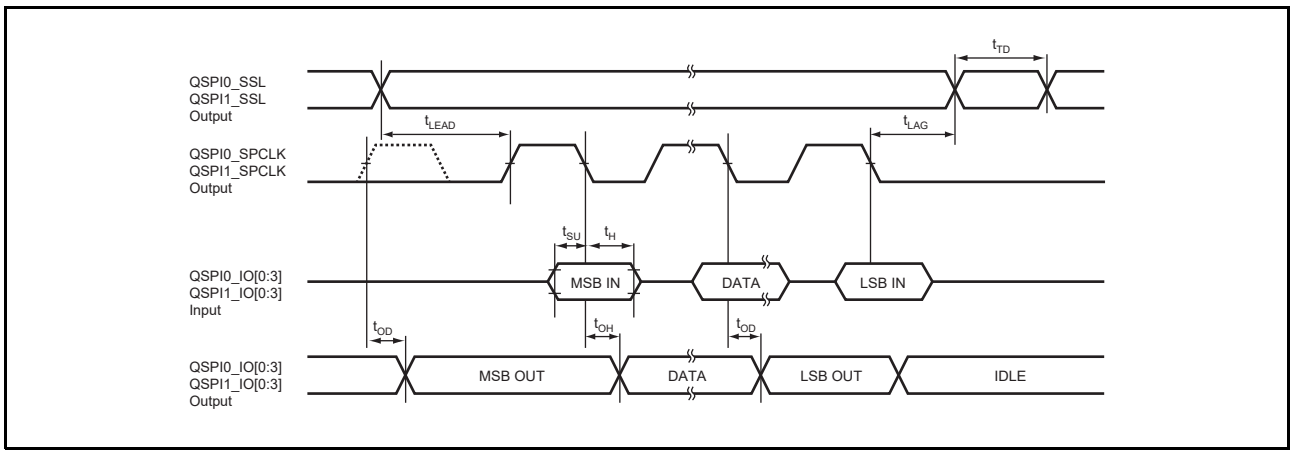


Figure 56.56 SDR Transfer Format Transmission and Reception Timing

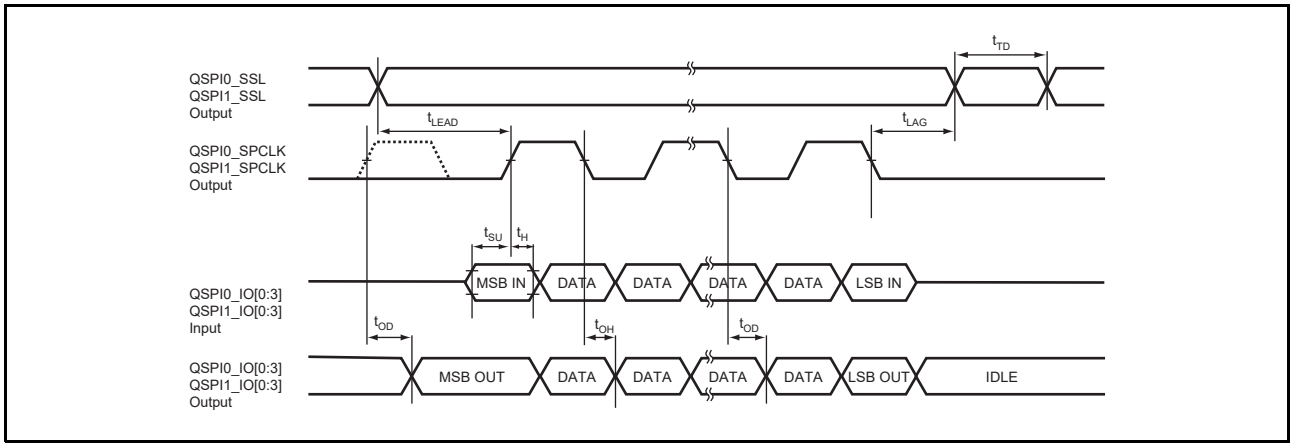


Figure 56.57 DDR Transfer Format Transmission and Reception Timing

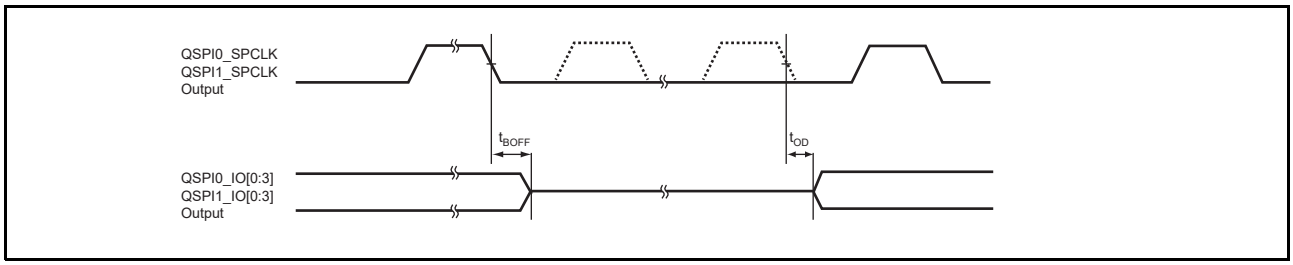


Figure 56.58 Timing for Switching the Buffers on and off

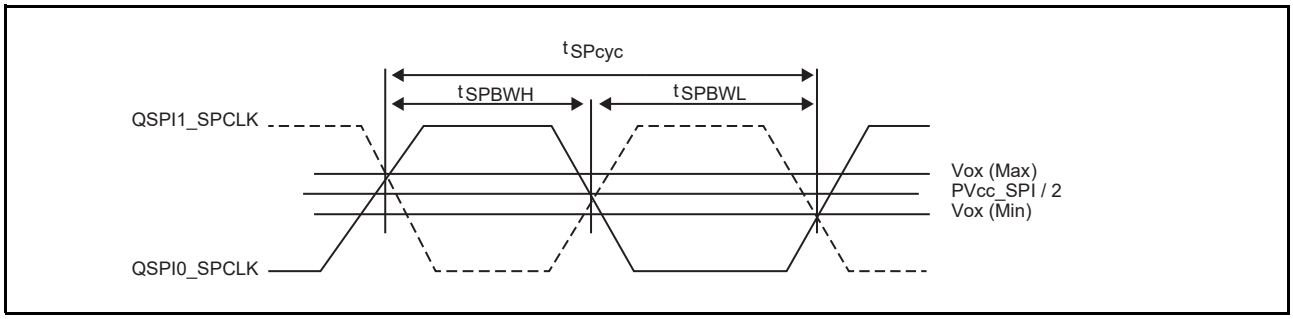


Figure 56.59 AC Differential Crossing Voltage

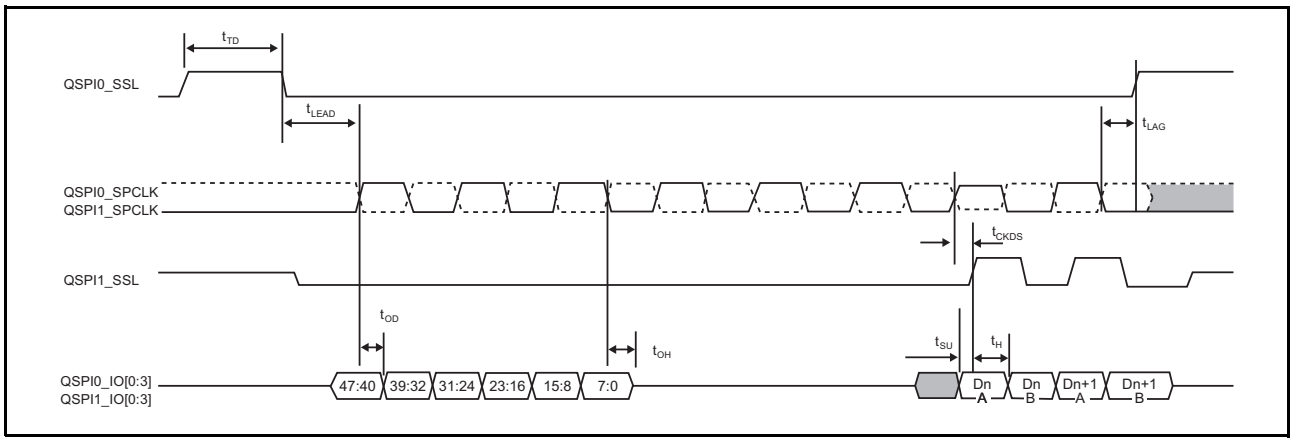


Figure 56.60 Transmit/Receive Timing with Octal-SPI flash memory or HyperFlash™ Connected

56.4.15 HyperBus™ Controller Timing

Table 56.19 HyperBus™ Controller Timing *1*2

Item	Symbol	Min.	Max.	Unit	Figure
HM_CK, HM_CK# clock frequency	f_{HY}	13.75	132	MHz	Figure 56.61, Figure 56.62
HM_CK, HM_CK# clock pulse width	t_{HYW}	0.475	0.525	t_{HYcyc}	
HM_CK, HM_CK# rise time	t_{HYr}	—	1	ns	
HM_CK, HM_CK# fall time	t_{HYf}	—	1	ns	
HM_CK, HM_CK# differential cross point voltage	V_{OX}	PVcc_HO × 0.4	PVcc_HO × 0.6	V	
HM_CS setup time	t_{CSSHY}	1 × t_{HYcyc} – 3 (Minimum register settings*3)	16 × t_{HYcyc} + 3 (Maximum register settings*3)	ns	Figure 56.63, Figure 56.64
HM_CS hold time	t_{CSHHY}	1 × t_{HYcyc} – 3 (Minimum register settings*4)	16 × t_{HYcyc} + 3 (Maximum register settings*4)	ns	
HM_CS High time (Continuous transfer)	t_{CSHIHY}	1.5 × t_{HYcyc} – 3 (Minimum register settings*5)	16.5 × t_{HYcyc} + 3 (Maximum register settings*5)	ns	
Data input setup time	t_{SU}	– 0.9 *7	—	ns	
Data input hold time	t_H	2.69 *7	—	ns	
Skew of clock to data strobe	t_{CKDS}	—	(N + 1) × t_{HYcyc} – 3 *6	ns	
Data output delay time	t_{OD}	—	2.69 *7	ns	
Data output hold time	t_{OH}	0.9 *7	—	ns	
RWDS refresh input setup time	t_{RWDSS}	12.5	—	ns	Figure 56.65
RWDS refresh input hold time	t_{RWDSH}	0.5 × t_{HYcyc}	—	ns	

Note 1. t_{HYcyc} indicates the HM_CK (HM_CK#) cycle.
Note 2. Output load: 15 pF
Note 3. The value of HM_CS0# is set by the WCSS bits and RCSS bits of the MTR0 register. The value of HM_CS1# is set by the WCSS bits and RCSS bits of the MTR1 register.
Note 4. The value of HM_CS0# is set by the WCSH bits and RCSH bits of the MTR0 register. The value of HM_CS1# is set by the WCSH bits and RCSH bits of the MTR1 register.
Note 5. The value of HM_CS0# is set by the WCSHI bits and RCSHI bits of the MTR0 register. The value of HM_CS1# is set by the WCSHI bits and RCSHI bits of the MTR1 register.
Note 6. N is RCSH bits value of the MTR0 or MTR1 register (N = 0 to 15).
Note 7. HM_CK frequency: 132 MHz

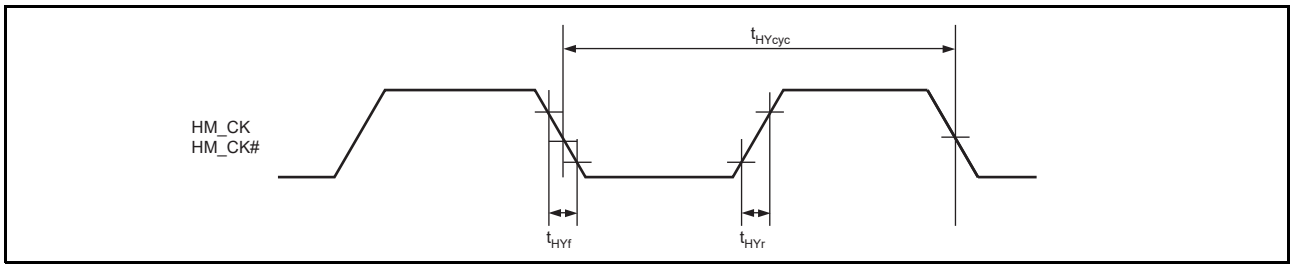


Figure 56.61 Clock Timing

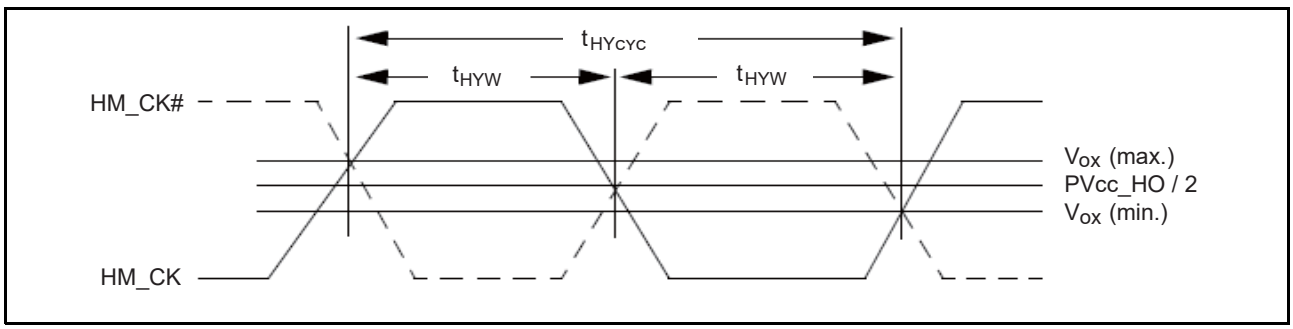


Figure 56.62 AC Differential Crossing Voltage

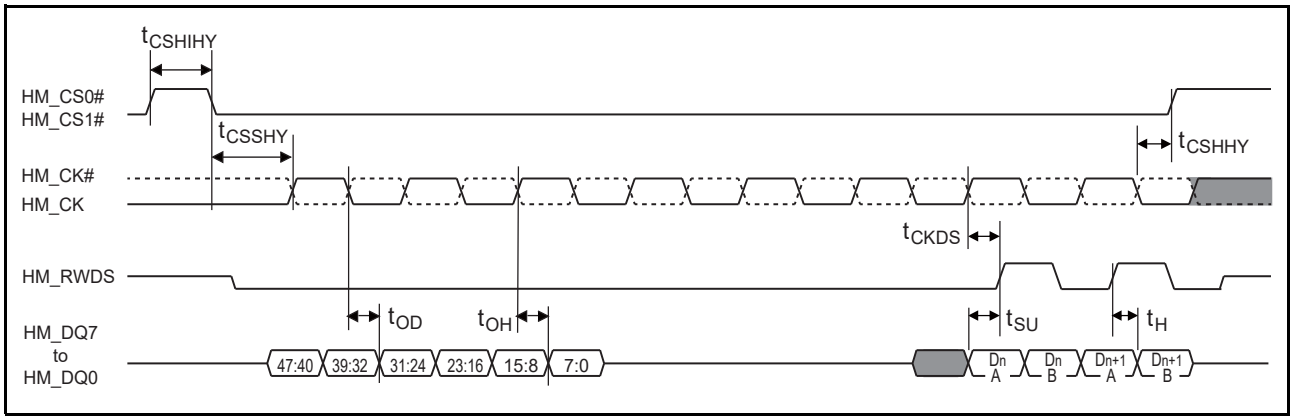


Figure 56.63 HyperBus™ Read Timing

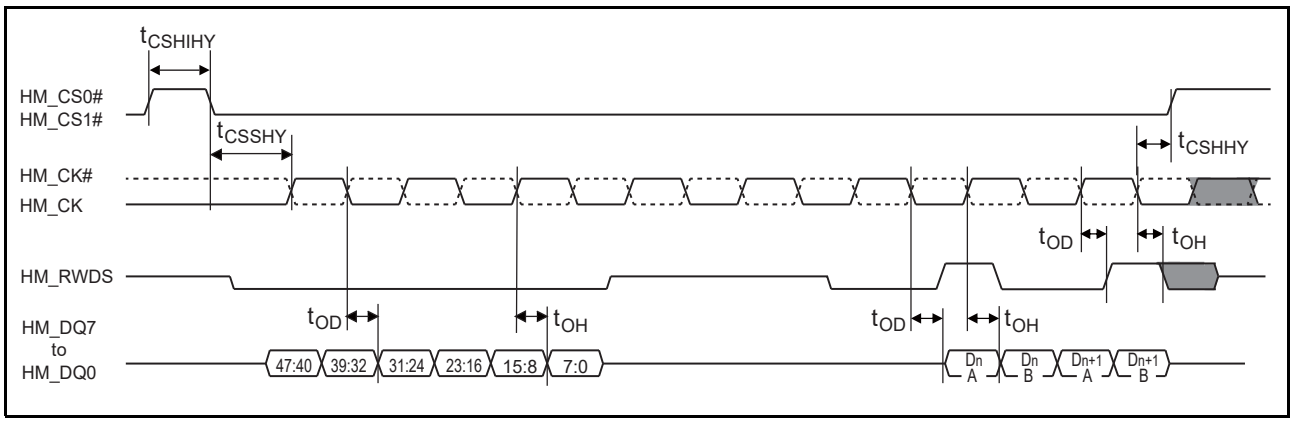


Figure 56.64 HyperBus™ Write Timing

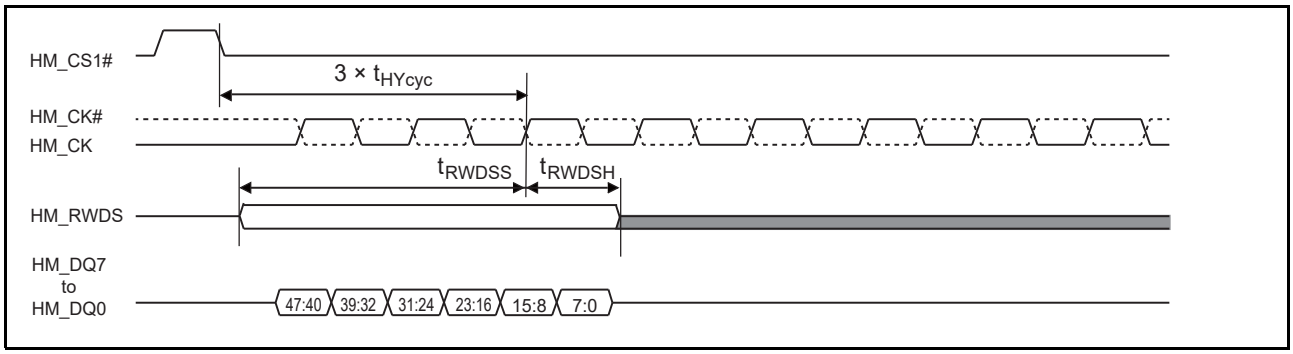


Figure 56.65 RWDS Refresh input Timing (HyperRAM™ Read/Write)

56.4.16 Octa Memory Controller Timing

Table 56.20 Octa Memory Controller Timing *1*2

Item		Symbol	Min.	Max.	Unit	Figure
OM_SCLK clockfrequency		f_{OCcyc}	—	132	MHz	Figure 56.66
OM_SCLK high pulse width		t_{OCwh}	0.475	0.525	t_{OCcyc}	
OM_SCLK low pulse width		t_{OCwl}	0.475	0.525	t_{OCcyc}	
OM_SCLK rise time		t_{OCr}	—	1	ns	
OM_SCLK fall time		t_{OCf}	—	1	ns	
OM_CS setup time	SPI/SOPI	t_{OCLEAD}	1 × t_{OCcyc} – 3 (Minimum register settings)	2.5 × t_{OCcyc} + 3 (Maximum register settings)	ns	Figure 56.67, Figure 56.68
	DOPI	t_{OCLEAD}	0.75 × t_{OCcyc} – 3 (Minimum register settings)	2.25 × t_{OCcyc} + 3 (Maximum register settings)	ns	Figure 56.69
OM_CS hold time	SPI/SOPI	t_{OCLAG}	1 × t_{OCcyc} – 3 (Minimum register settings)	4.5 × t_{OCcyc} + 3 (Maximum register settings)	ns	Figure 56.67, Figure 56.68
	DOPI read	t_{OCLAG}	3.25 × t_{OCcyc} – 3 (Minimum register settings)	4.25 × t_{OCcyc} + 3 (Maximum register settings)	ns	Figure 56.69
	DOPI write	t_{OCLAG}	0.75 × t_{OCcyc} – 3 (Minimum register settings)	4.25 × t_{OCcyc} + 3 (Maximum register settings)	ns	
Continuous transfer delay time		t_{OCTD}	1 × t_{OCcyc} – 3 (Minimum register settings)	8.5 × t_{OCcyc} + 3 (Maximum register settings)	ns	Figure 56.67, Figure 56.68, Figure 56.69
Data input setup time	SPI SCLK base point	t_{SU}	9.6	—	ns	Figure 56.67
Data input hold time		t_H	0.5	—	ns	
Data input setup time	SOPI/DOPI DQS base point *3	t_{SU}	– 0.7	—	ns	Figure 56.68, Figure 56.69
Data input hold time		t_H	2.69	—	ns	
Skew of Clock to Data Strobe		t_{CKDS}	—	20	ns	
Data output delay time	SPI/SOPI	t_{OD}	—	1.4	ns	Figure 56.67, Figure 56.68
Data output hold time		t_{OH}	– 1.4	—	ns	
Data output buffer off time	SOPI	t_{BOFF}	2	—	ns	Figure 56.68
Data output delay time	DOPI *3	t_{OD}	—	2.69	ns	Figure 56.69, Figure 56.70
Data output hold time		t_{OH}	0.9	—	ns	
Data output buffer off time	DOPI	t_{BOFF}	0.9	—	ns	Figure 56.69
DQS refresh input setup time		t_{DQSS}	12	—	ns	Figure 56.71
DQS refresh input hold time		t_{DQSH}	0.5 × t_{OCcyc}	—	ns	

Note 1. t_{OCcyc} indicates the OM_SCLK cycle.
Note 2. Maximum load capacitance: 15 pF
Note 3. OM_SCLK frequency: 132 MHz

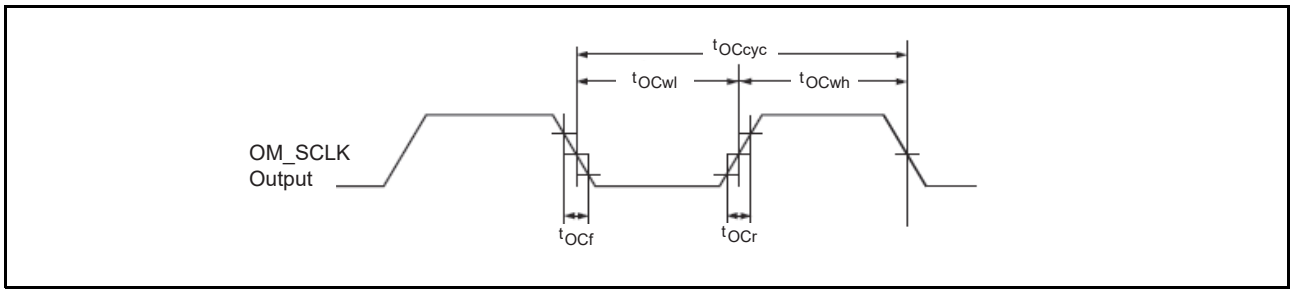


Figure 56.66 Clock Timing

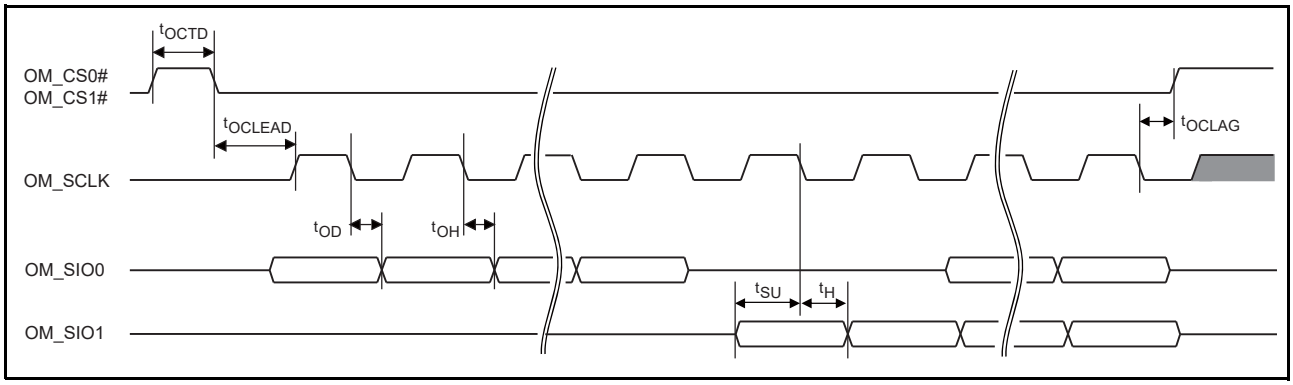


Figure 56.67 SPI Transfer Format Transmission and Reception Timing

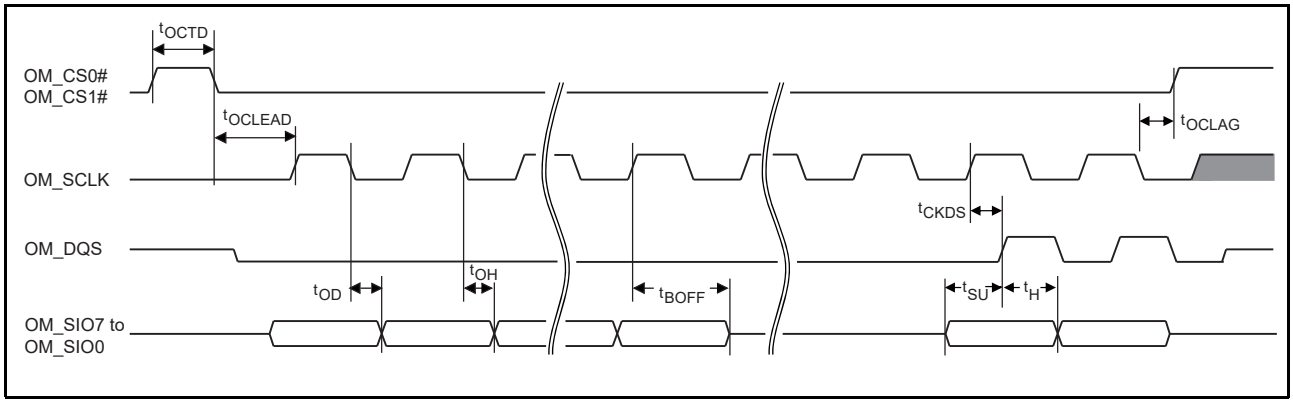


Figure 56.68 SOPI Transfer Format Transmission and Reception Timing

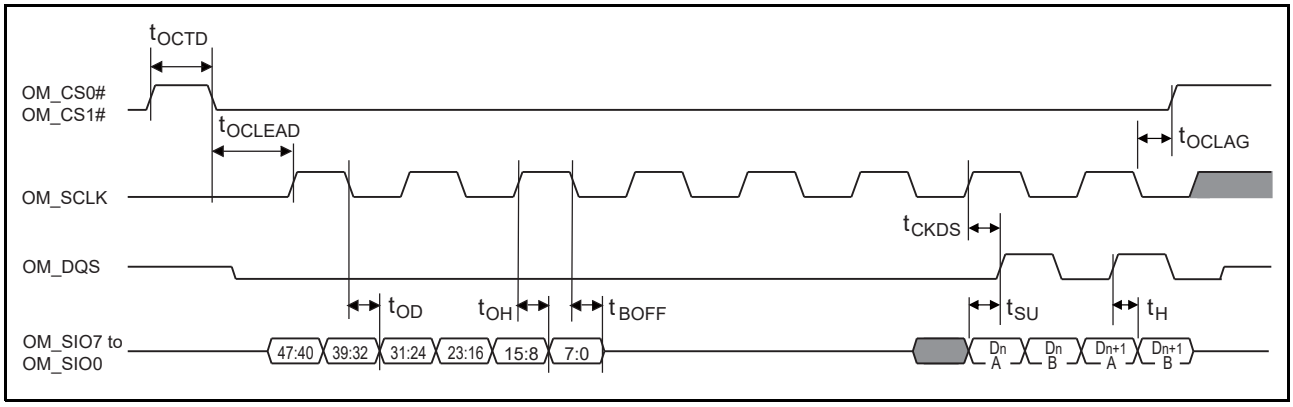


Figure 56.69 DOPI Transfer Format Transmission and Reception Timing

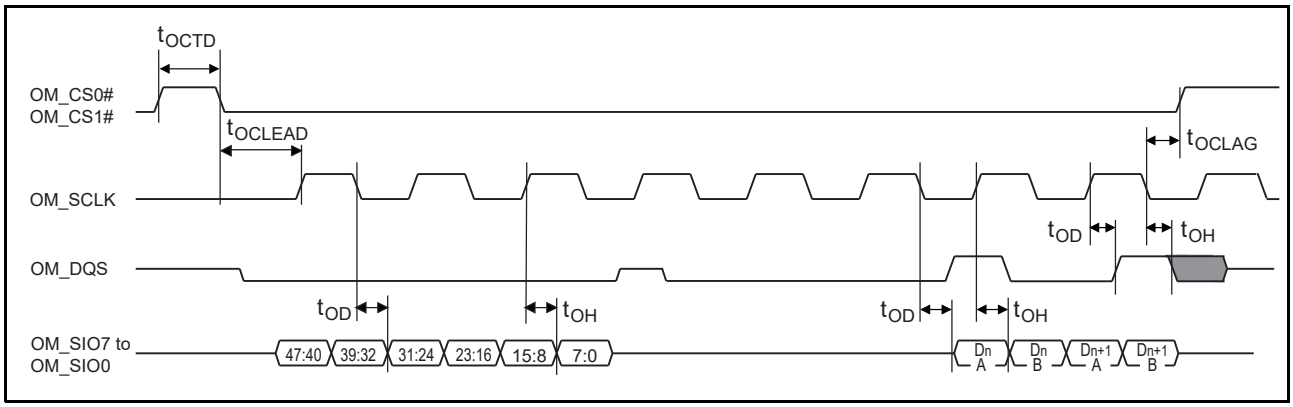


Figure 56.70 DOPI Transfer Format Transmission Timing

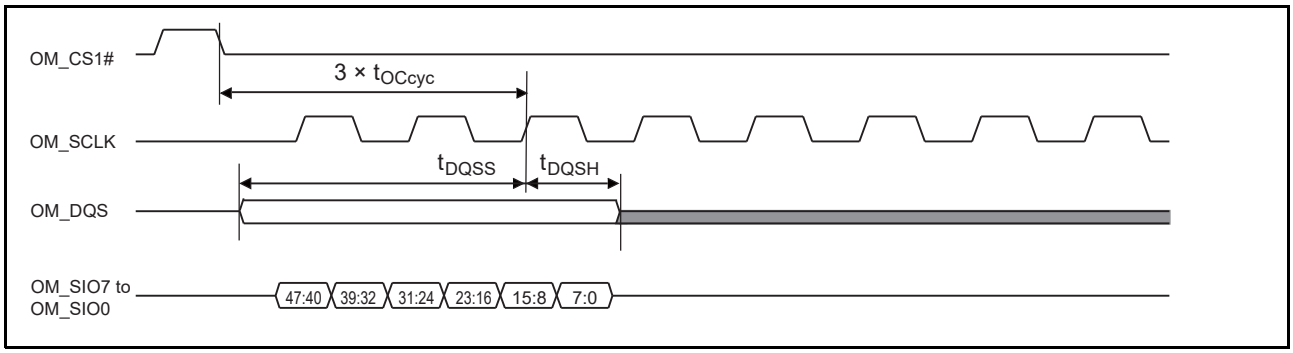


Figure 56.71 DQS Refresh input Timing (OctaRAM™ Read/Write)

56.4.17 I²C Bus Interface Timing

Table 56.21 I²C Bus Interface Timing

Item	Symbol	I/O	**Standard mode** (Sm)		**Fast mode** (Fm)		Fast mode plus (Fm+)		Unit
			Min.	Max.	Min.	Max.	Min.	Max.	
SCL clock frequency	f_{CLK}	I/O	0	100	0	400	0	1000	kHz
Bus free time (between stop and start condition)	t_{BUF}	I/O	4.7	—	1.3	—	0.5	—	µs
Hold time*1	$t_{HD:STA}$	I/O	4.0	—	0.6	—	0.26	—	µs
Low period of SCL clock	t_{LOW}	I/O	4.7	—	1.3	—	0.5	—	µs
High period of SCL clock	t_{HIGH}	I/O	4.0	—	0.6	—	0.26	—	µs
Setup time for start/restart condition	$t_{SU:STA}$	I/O	4.7	—	0.6	—	0.26	—	µs
Data hold time (I²C bus device)	$t_{HD:DAT}$	I/O	0*2	—	0*2	—	0	—	µs
Data setup time	$t_{SU:DAT}$	I/O	250	—	100*3	—	50	—	ns
SDA and SCL signal rise time	t_R	Input	—	1000	20	300	—	120	ns
SDA and SCL signal fall time*3	t_F	Input	—	300	20 × (PVcc / 5.5 V)	300	20 × (PVcc / 5.5 V)	120	ns
		Output	—	250	20 × (PVcc / 5.5 V)	250	20 × (PVcc / 5.5 V)	120	ns
Setup time for STOP condition	$t_{SU:STO}$	I/O	4.0	—	0.6	—	0.26	—	µs
Capacitive load for each bus line	C_b	—	—	400*4	—	400*4	—	550*4	pF
Pulse width of spikes that must be suppressed by the input filter	t_{SP}	Input	—	—	0	50*5	0	50*5	ns

In the above table and subsequently, SCL and SDA refer to the RIICnSCL and RIICnSDA signals, respectively.

Note 1. The first clock pulse is generated on the SCL line after the start condition has been issued and the hold time has elapsed.

Note 2. This module requires a minimum of 300 ns hold time internally for the SDA signal to handle the period over which the falling edge of SCL has not reached a defined level (time until the CnSCL signal reaches V_{IL} (max.) from V_{IH} (min.)).

Note 3. The fast-mode I²C bus device can be used in the standard mode I²C bus system. In this case, the minimum value of the data setup time ($t_{SU: DAT (min.)}$ 250 [ns]) must be satisfied.
If the system does not extend the low period of SCL clock (t_{LOW}), this condition is automatically satisfied. If the system extends the low period of SCL clock (t_{LOW}), transmit the subsequent data bit to the SDA line before the SCL line is released ($t_{R (max.)}$ + $t_{SU: DAT (min.)}$ = 1000 + 250 = 1250 [ns]: (standard mode I²C bus specification)).

Note 4. Total capacitance of one bus line. The allowable maximum bus capacitance may differ from this specification, depending on the actual operating voltage and frequency of an application. For techniques to cope with a large bus capacitance, see the I²C bus specification provided by NXP Semiconductors.

Note 5. Noise is removed by the analog and digital input filters. The level of noise reduction of the digital input filter is determined by the period of internal reference clock (IICφ) and the NF[1:0] bits in RIICnMR3. For details, refer to section 23, I²C Bus Interface.

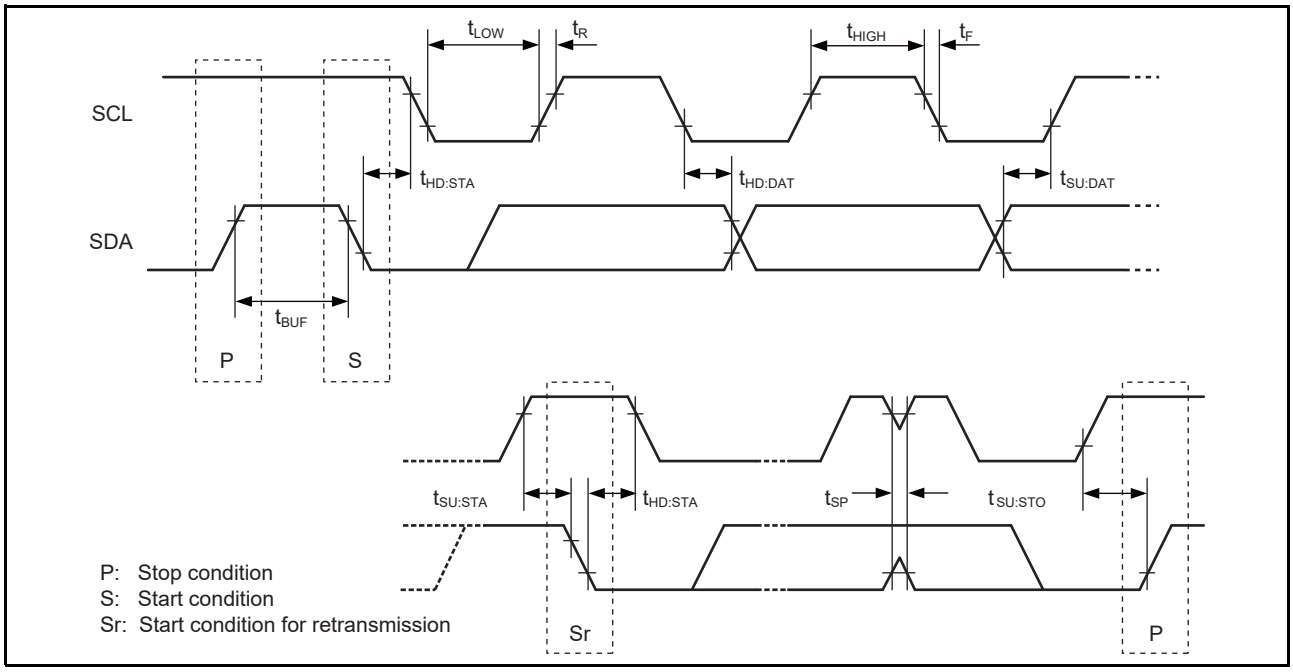


Figure 56.72 Input/Output Timing

56.4.18 Serial Sound Interface (SSIF-2) Timing

Table 56.22 SSIF-2 Timing

Item	Remarks	Symbol	Target Specification Min.	Target Specification Max.	Unit	Figure
Output clock cycle	Output	t_O	80	64000	ns	Figure 56.73
Input clock cycle	Input	t_I	80	64000	ns	
Clock high	Bidirectional	t_{HC}	32	—	ns	
Clock low		t_{LC}	32	—	ns	
Clock rise time/Clock fall time	Output	t_{RC}/t_{FC}	—	25	ns	
Setup time		t_{SR}	25	—	ns	Figure 56.74, Figure 56.75, Figure 56.76
Hold time		t_{HR}	5	—	ns	
SSILRCK output delay time		t_{DTR}	–5	15	ns	
Data output delay time (Noise canceler not in use)		t_{DTR}	–5	25	ns	
Data output delay time (Noise canceler in use)		t_{DTR}	10	45	ns	

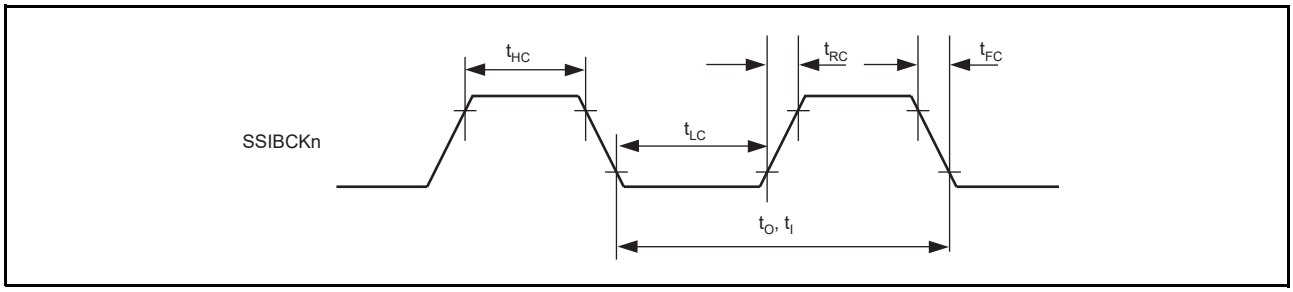


Figure 56.73 Bit Clock Input/Output Timing

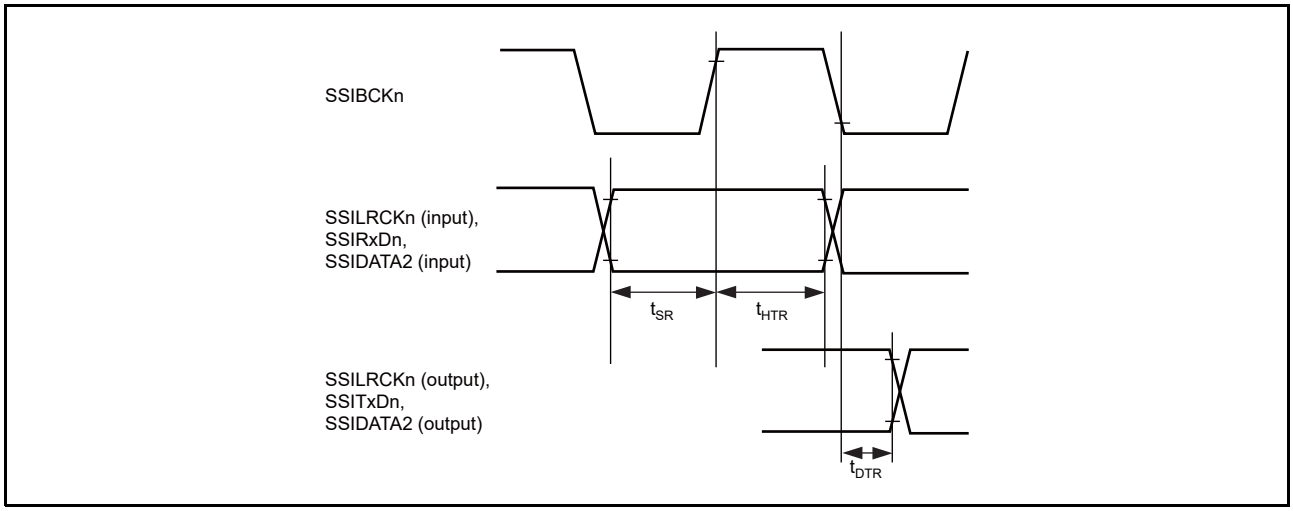


Figure 56.74 Transmission and Reception Timing (SSIBCK Falling Output)

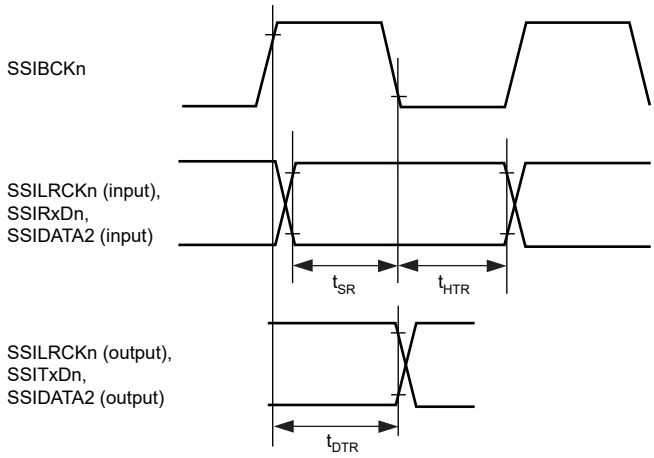
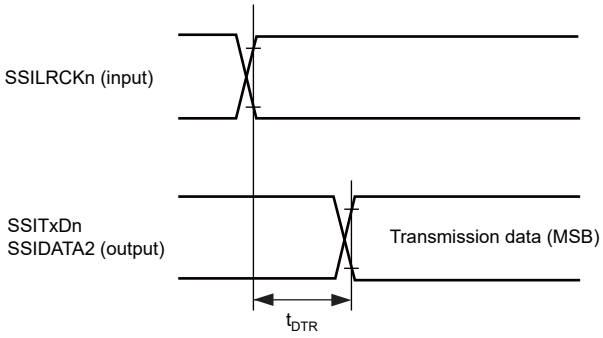


Figure 56.75 Transmission and Reception Timing (SSIBCK Rising Output)



Note. Output timing of the MSB when slave transmission satisfies the following condition a) or b)
a) SSICR.DEL=1, SSICR.SDTA=0
b) SSICR.DEL=1, SSICR.SDTA=1, without Padding

Figure 56.76 Transmission Timing (Slave, in Synchronization with SSILRCK)

56.4.19 CANFD Interface Timing

Table 56.23 CANFD Interface Timing

Item	Symbol	CAN		CANFD		Unit	Conditions
		Min.	Max.	Min.	Max.		
Internal delay time	t_{node}	—	100	—	75	ns	Figure 56.77
Transmission rate		—	1	—	4.125	Mbps	

Internal delay time (t_{node}) = Internal transfer delay time (t_{output}) + Internal receive delay time (t_{input})

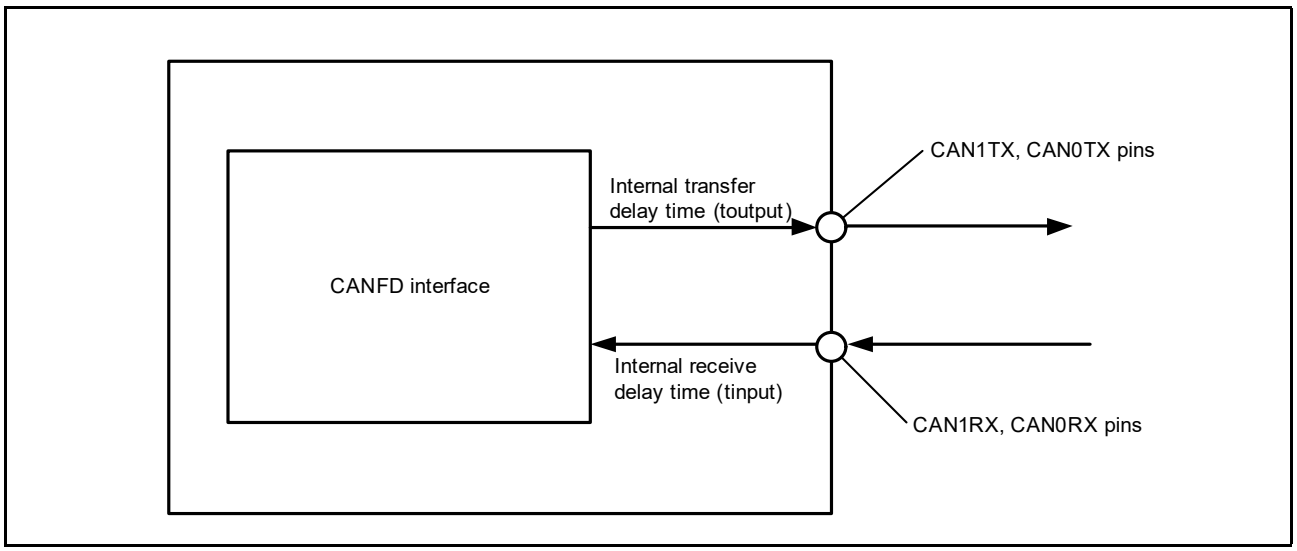


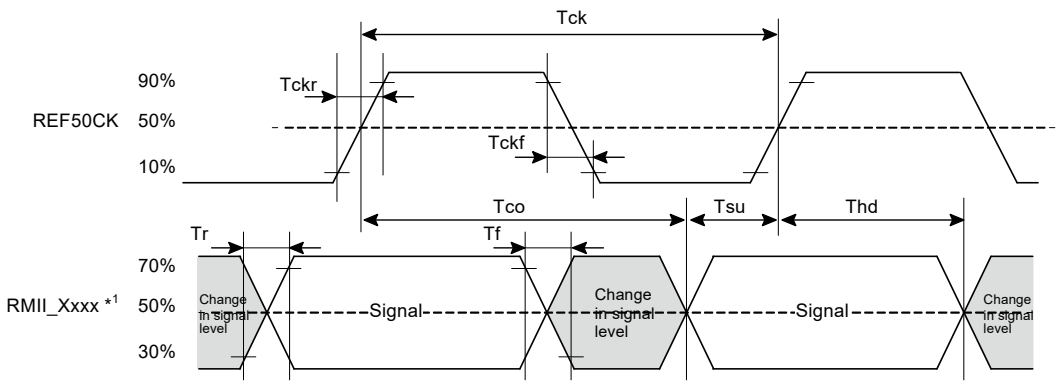
Figure 56.77 CANFD Interface Condition

56.4.20 Ethernet Controller (ETHERC) Timing

Table 56.24 ETHERC Timing

Item		Symbol	Min.	Max.	Unit	Test Conditions
ETHERC (RMII)	REF50CK cycle time	T_{ck}	20	—	ns	Figure 56.78 to Figure 56.81
	REF50CK frequency Typ. 50 MHz	—	—	50 + 100ppm	MHz	
	REF50CK duty	—	35	65	%	
	REF50CK rise/fall time	$T_{ckr/ckf}$	0.5	3.5	ns	
	RMII_xxxx*1 output delay time	T_{co}	2.5	15.0	ns	
	RMII_xxxx*2 setup time	T_{su}	3	—	ns	
	RMII_xxxx*2 hold time	T_{hd}	1	—	ns	
	RMII_xxxx*1*2 rise/fall time	T_r/T_f	0.5	6	ns	
ETHERC (MII)	ET_TX_CLK cycle time	t_{Tcyc}	40	—	ns	—
	ET_TX_EN output delay time	t_{TENd}	1	20	ns	Figure 56.82
	ET_ETXD0 to ET_ETXD3 output delay time	t_{MTDd}	1	20	ns	
	ET_CRS setup time	t_{CRSs}	10	—	ns	
	ET_CRS hold time	t_{CRSh}	10	—	ns	
	ET_COL setup time	t_{COLs}	10	—	ns	Figure 56.83
	ET_COL hold time	t_{COLh}	10	—	ns	
	ET_RX_CLK cycle time	t_{TRcyc}	40	—	ns	—
	ET_RX_DV setup time	t_{RDVs}	10	—	ns	Figure 56.84
	ET_RX_DV hold time	t_{RDVh}	10	—	ns	
	ET_ERXD0 to ET_ERXD3 setup time	t_{MRDs}	10	—	ns	
	ET_ERXD0 to ET_ERXD3 hold time	t_{MRDh}	10	—	ns	
	ET_RX_ER setup time	t_{RERs}	10	—	ns	Figure 56.85
	ET_RX_ER hold time	t_{RERh}	10	—	ns	

Note 1. RMII_TXD_EN, RMII_TXD1, RMII_TXD0
Note 2. RMII_CRS_DV, RMII_RXD1, RMII_RXD0, RMII_RX_ER



Note 1. RMII_TXD_EN, RMII_TXD1, RMII_TXD0, RMII_CRS_DV, RMII_RXD1, RMII_RXD0, RMII_RX_ER

Figure 56.78 Timing with the REF50CK and RMII Signals

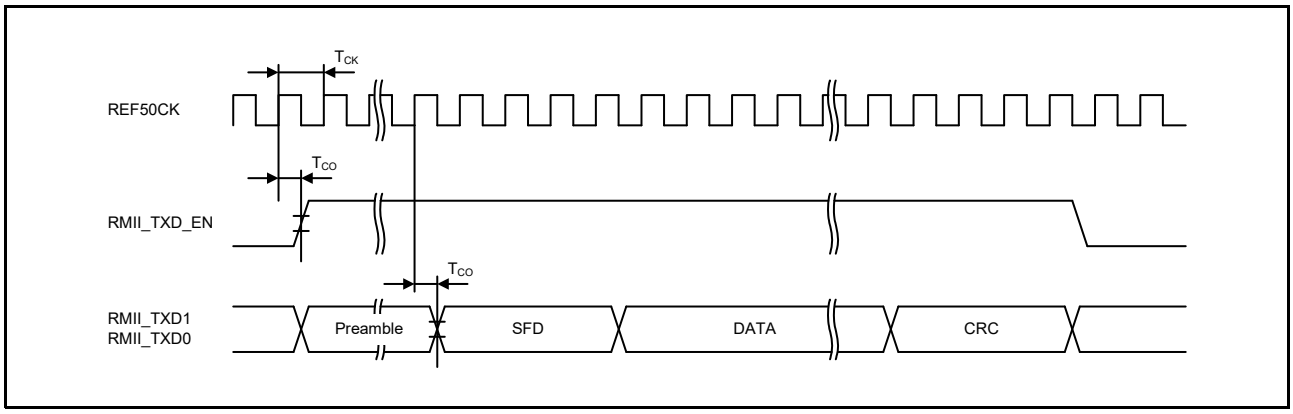


Figure 56.79 RMII Transmission Timing

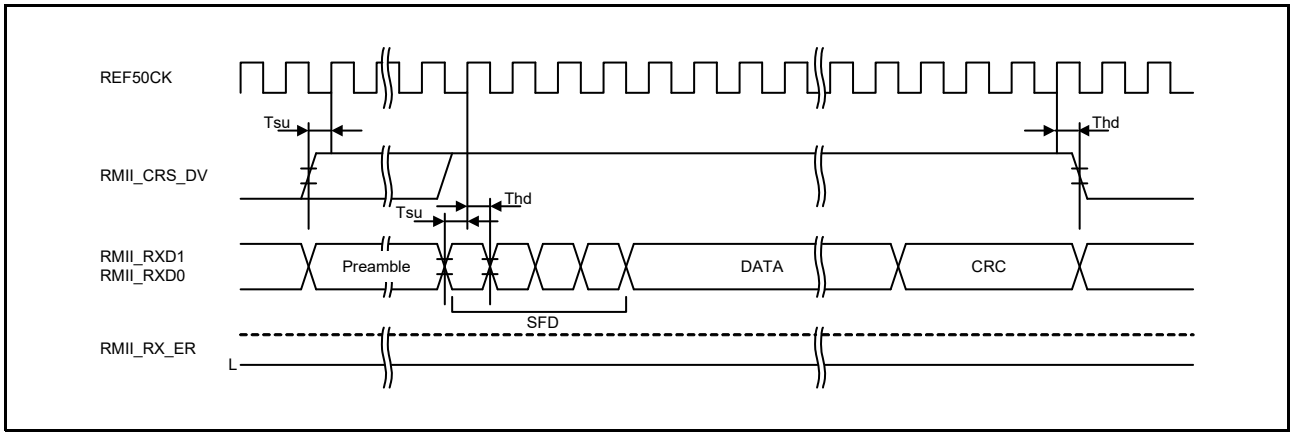


Figure 56.80 RMII Reception Timing (Normal Operation)

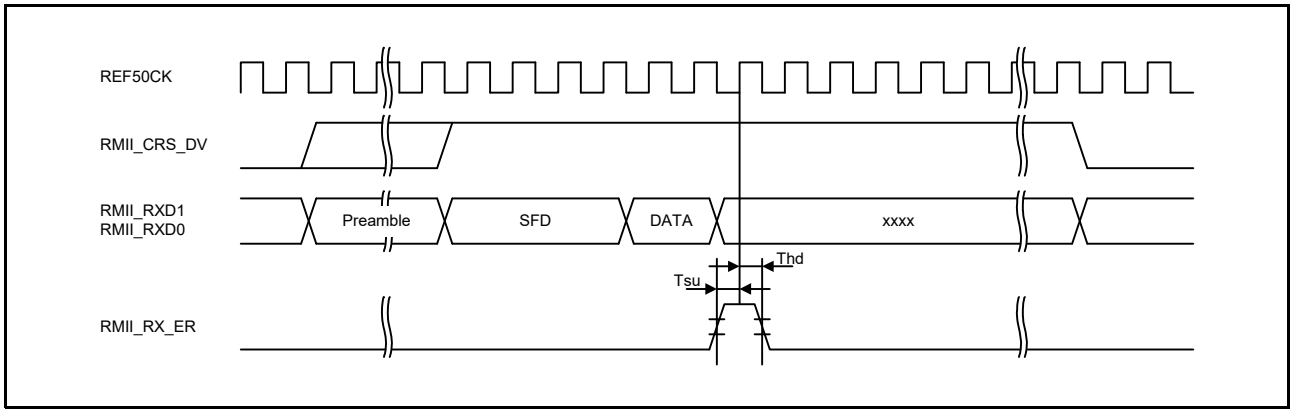


Figure 56.81 RMII Reception Timing (Error Occurrence)

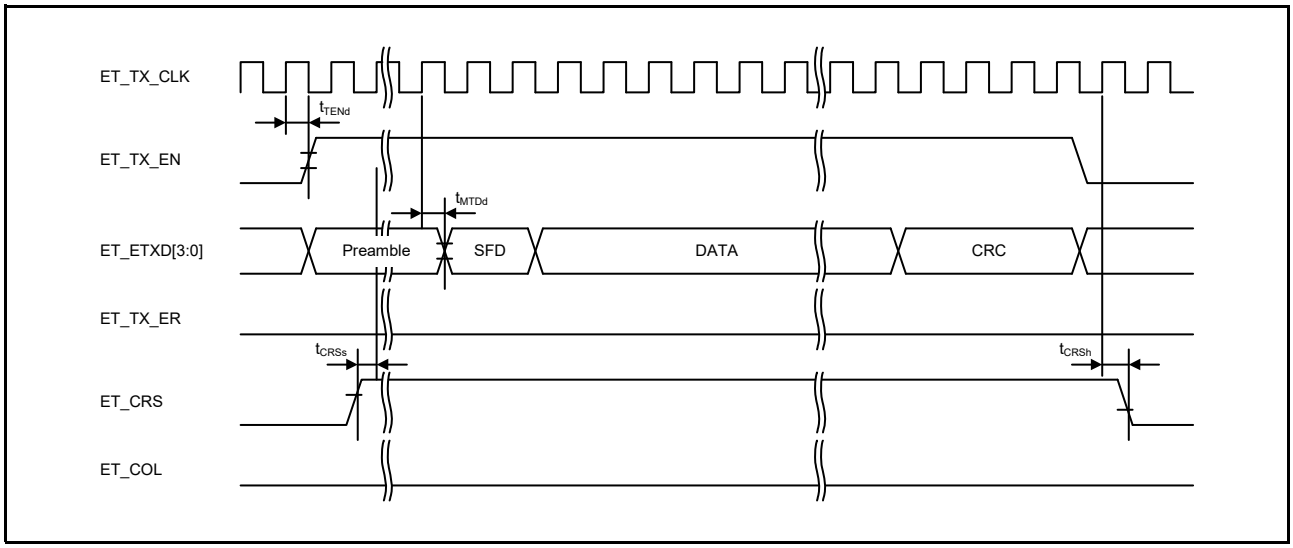


Figure 56.82 MII Transmission Timing (Normal Operation)

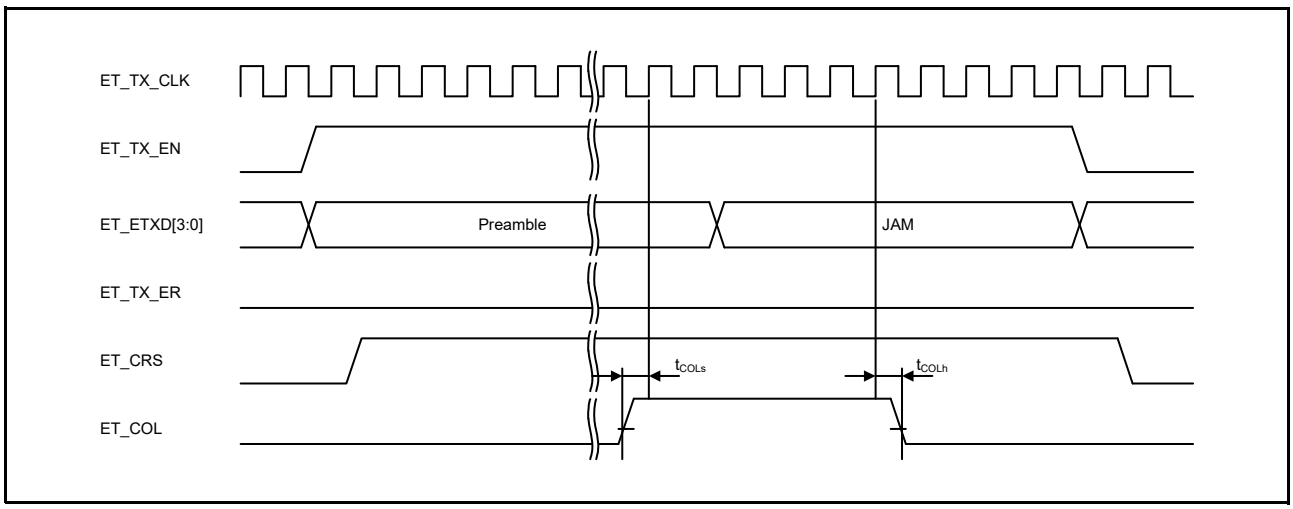


Figure 56.83 MII Transmission Timing (Conflict Occurrence)

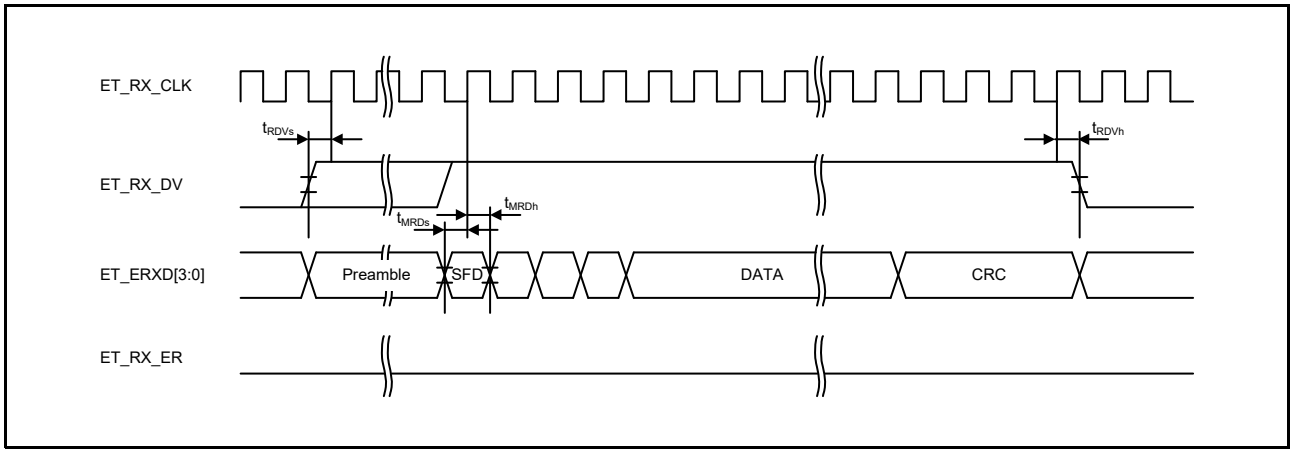


Figure 56.84 MII Reception Timing (Normal Operation)

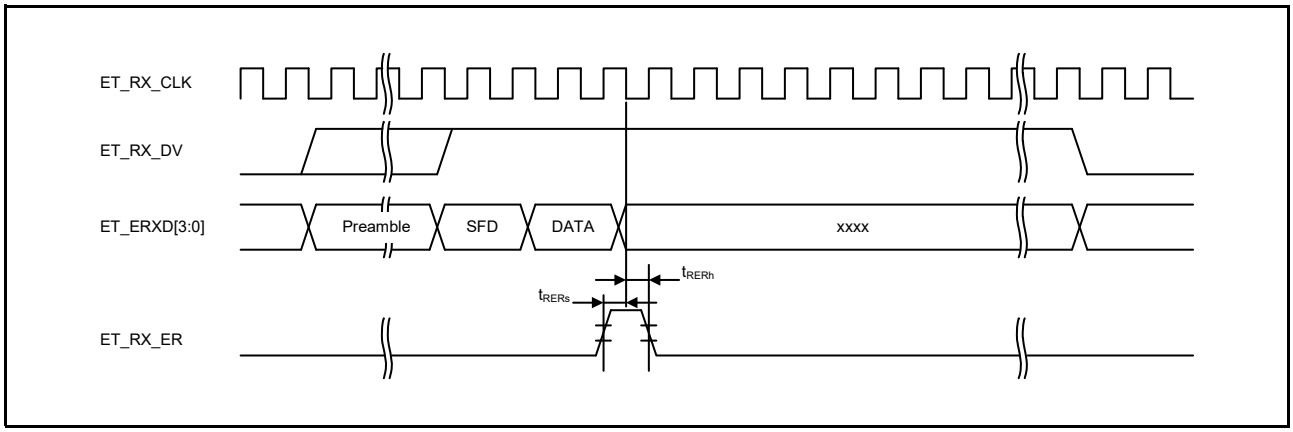


Figure 56.85 MII Reception Timing (Error Occurrence)

56.4.21 A/D Converter Timing

Table 56.25 A/D Converter Timing

Item		Symbol	Min.	Max.	Unit	Test Conditions
ADC12	ADC12 Trigger Input Pulse Width	t_{TRGW}	1.5	—	t_{Pcyc}	Figure 56.86

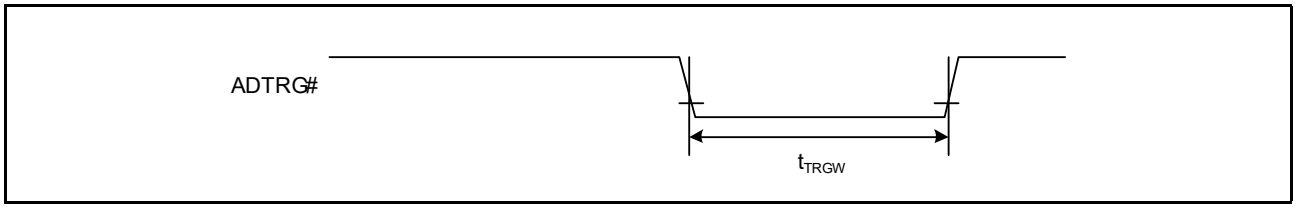


Figure 56.86 ADC12 Trigger Input Timing

56.4.22 NAND Flash Controller Timing

Table 56.26 NAND Flash Controller Timing

Item	Symbol	Min.	Max.	Unit	Test Conditions
Delay time between output signals (t_{OD} delay difference of each output signals)	t_{RFD}	—	5	ns	
Read data setup time (based on the rising edge of NFRE#)	t_{SD}	12	—	ns	Figure 56.87
Read data hold time (based on the rising edge of NFRE#)	t_{HD}	0	—	ns	
Time of NFWE# rising edge to NFRB# falling edge (NFRB# Capture start time)	t_{WB}	N × t_{p1cyc} *1 *2	—	ns	
Time of Ready to NFRE# falling edge	t_{RR}	M × t_{p1cyc} *1 *3	—	ns	

Note 1. t_{p1cyc} indicates the peripheral clock 1 (P1φ) cycle.
Note 2. N is TWB bits value of the TIME_SEQ_1 register.
Note 3. M is TRR bits value of the TIME_SEQ_1 register.

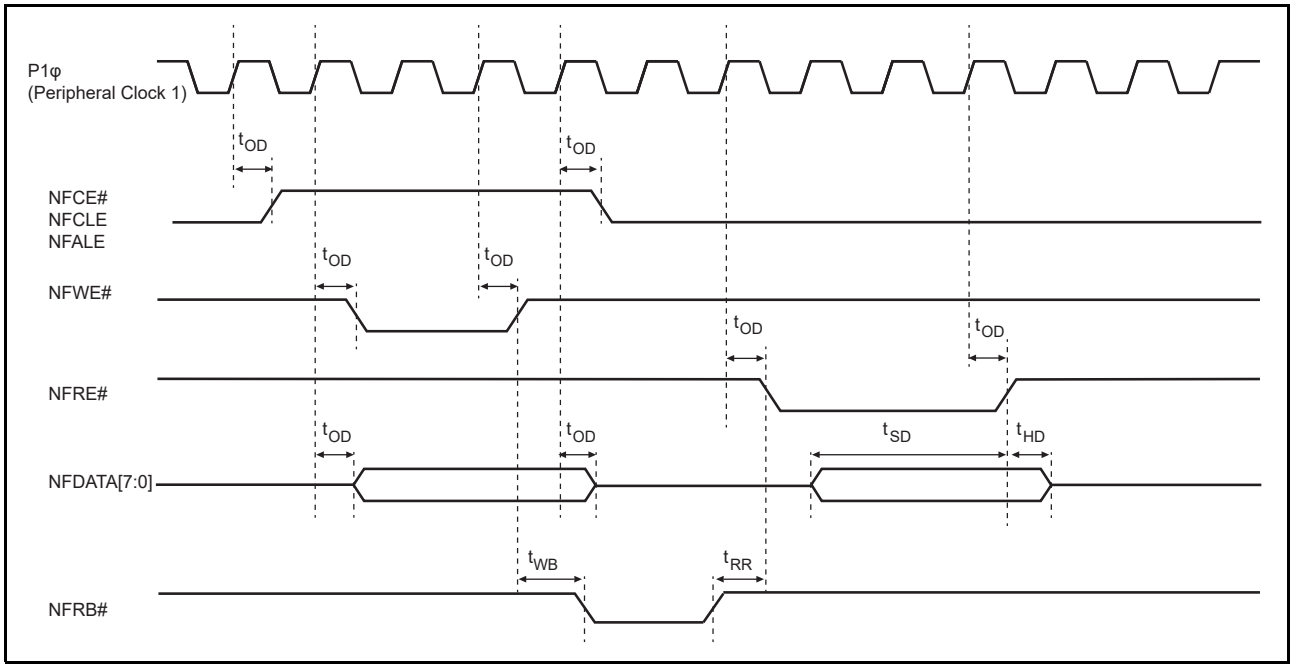


Figure 56.87 NAND Flash Controller Timing

56.4.23 USB 2.0 Host/Function Module Timing

Table 56.27 USB Transceiver Timing (Low-Speed)

Item	Symbol	Min.	Max.	Unit	Figure
Rise time	t_{LR}	75	300	ns	Figure 56.88
Fall time	t_{LF}	75	300	ns	
Rise/fall time lag	t_{LR}/t_{LF}	80	125	%	

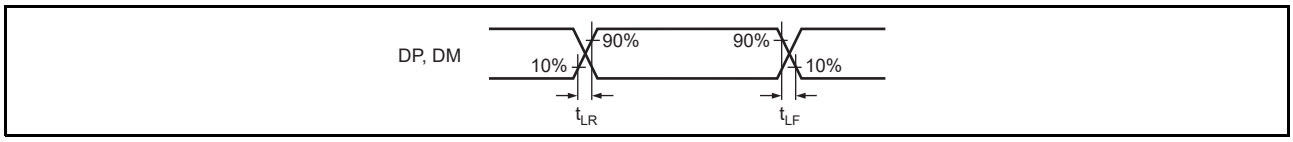


Figure 56.88 DP1, DP0, DM1, and DM0 Output Timing (Low-Speed)

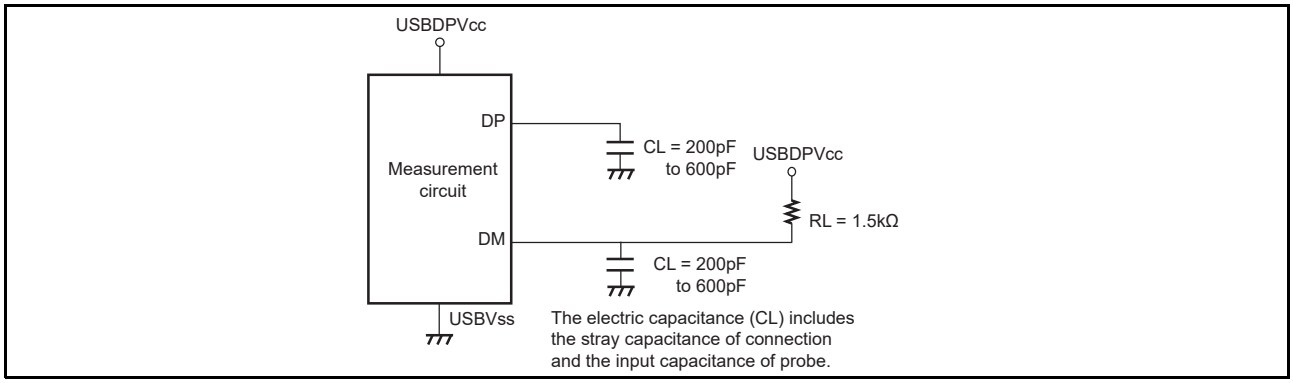


Figure 56.89 Measurement Circuit (Low-Speed)

Table 56.28 USB Transceiver Timing (Full-Speed)

Item	Symbol	Min.	Max.	Unit	Figure
Rise time	t_{FR}	4	20	ns	Figure 56.90
Fall time	t_{FF}	4	20	ns	
Rise/fall time lag	t_{FR}/t_{FF}	90	111.11	%	

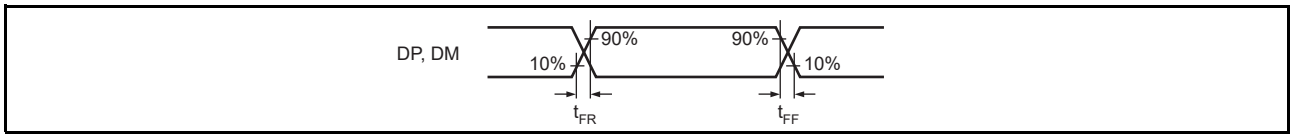


Figure 56.90 DP1, DP0, DM1, and DM0 Output Timing (Full-Speed)

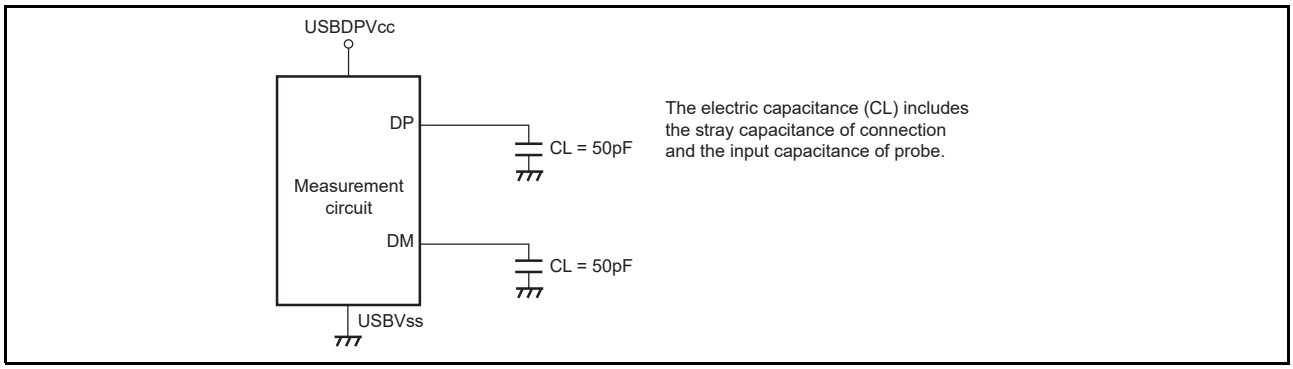


Figure 56.91 Measurement Circuit (Full-Speed)

Table 56.29 USB Transceiver Timing (Hi-Speed)

Item	Symbol	Min.	Max.	Unit	Figure
Rise time	t_{HSR}	500	—	ps	Figure 56.92
Fall time	t_{HSF}	500	—	ps	
Output driver resistance	Z_{HSDRV}	40.5	49.5	Ω	

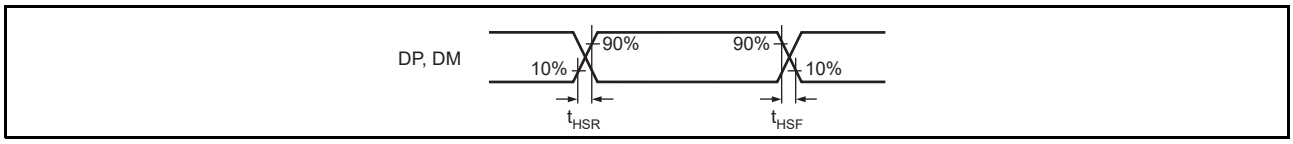


Figure 56.92 DP1, DP0, DM1, and DM0 Output Timing (Hi-Speed)

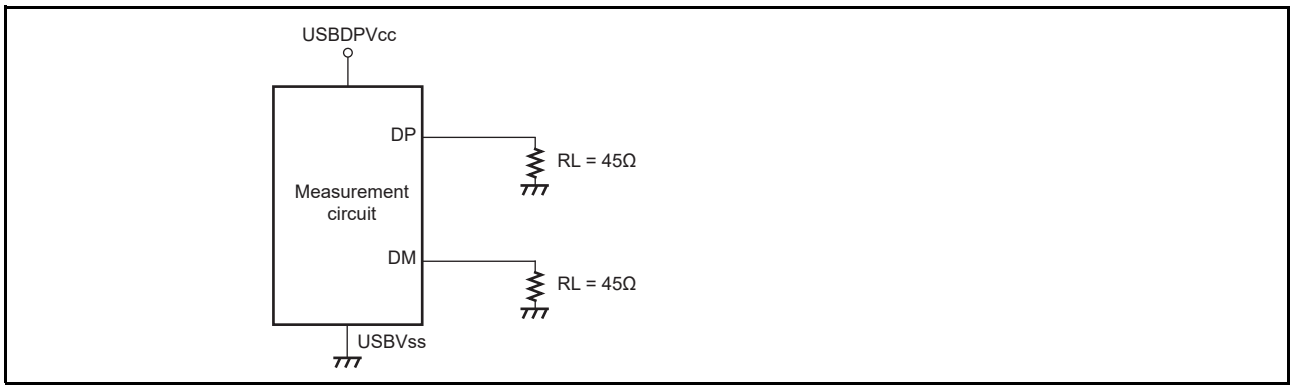


Figure 56.93 Measurement Circuit (Hi-Speed)

56.4.24 Video Display Controller 6 Timing

Table 56.30 Video Display Controller 6 Timing

Item	Symbol	Min.	Max.	Unit	Figure
DV0_CLK input clock cycle	t_{Dcyc}	11.50	—	ns	Figure 56.94
DV0_CLK input clock low pulse width	t_{WIL}	0.4	—	t_{Dcyc}	
DV0_CLK input clock high pulse width	t_{WIH}	0.4	—		
LCD0_EXTCLK input clock cycle	t_{Ecyc}	11.50	—	ns	
LCD0_EXTCLK input clock low pulse width	t_{WIL}	0.4	—	t_{Ecyc}	
LCD0_EXTCLK input clock high pulse width	t_{WIH}	0.4	—		
LCD0_CLK output clock cycle	t_{Lcyc}	11.50	—	ns	Figure 56.95
LCD0_CLK clock output low pulse width*1	t_{LOL}	t_{WIL} – 0.95	t_{WIL} + 0.95	ns	
LCD0_CLK clock output high pulse width*1	t_{LOH}	t_{WIH} – 0.95	t_{WIH} + 0.95	ns	
LCD0_CLK clock output low pulse width*2	t_{LOL}	t_{Lcyc} / 2 – 1.06	t_{Lcyc} / 2 + 1.06	ns	
LCD0_CLK clock output high pulse width*2	t_{LOH}	t_{Lcyc} / 2 – 1.06	t_{Lcyc} / 2 + 1.06	ns	
LCD0_CLK clock output rise time	t_{LOR}	—	3	ns	
LCD0_CLK clock output fall time	t_{LOF}	—	3	ns	
Input data setup time	t_{VS}	2	—	ns	Figure 56.96
Input data hold time	t_{VH}	4	—	ns	
Output data delay time	t_{DD}	–3	3*3	ns	Figure 56.97
			4*4		

Note 1. This is the case when the video image clock or an external clock is selected as the clock for frequency division and the division ratio is set to 1/1.
Note 2. This is for cases other than when the video image clock or an external clock is selected as the clock for frequency division and the division ratio is set to 1/1.
Note 3. This is the case when normal driving ability is set with LCD0_CLK pin.
Note 4. This is the case when high driving ability is set with LCD0_CLK pin.

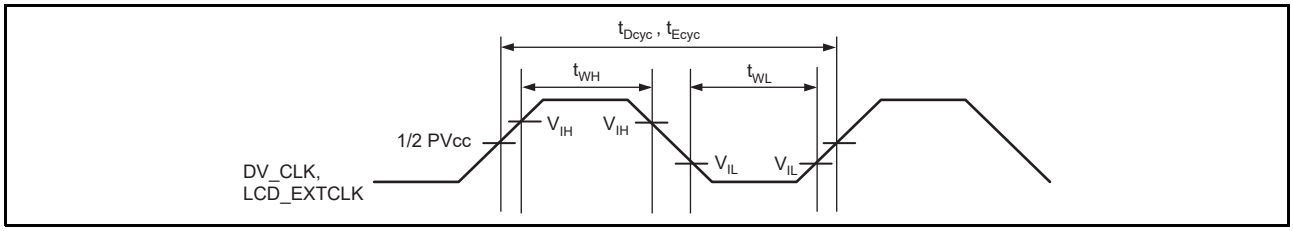


Figure 56.94 DV0_CLK and LCD0_EXTCLK Clock Input Timing

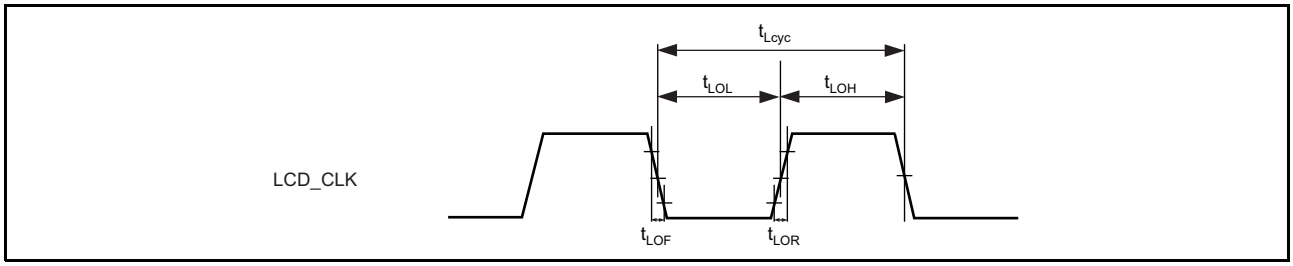


Figure 56.95 LCD0_CLK Clock Output Timing

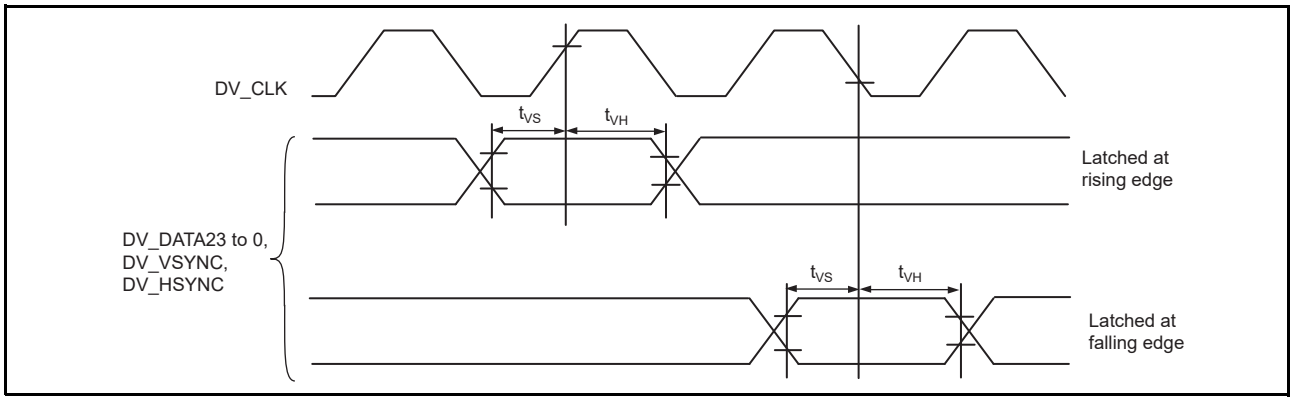


Figure 56.96 Video Input Timing

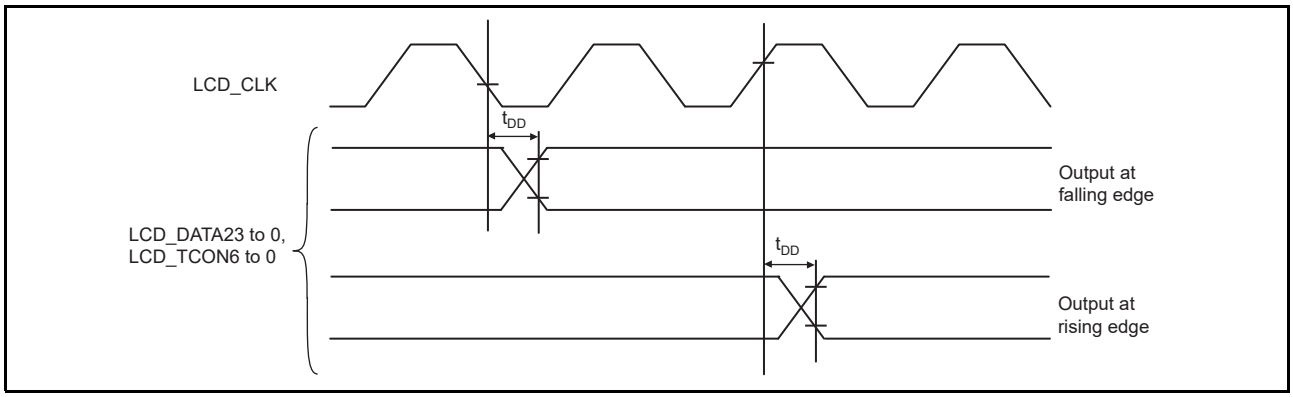


Figure 56.97 Display Output Timing

56.4.25 LVDS Output Interface Timing

Table 56.31 LVDS Timing

Item	Symbol	Min.	Typ.	Max.	Unit	Figure
Panel clock for LVDS output (LSI internal signal)	T	11.49 (87 MHz)	—	24.1 (41.43 MHz)	ns	Figure 56.98
Transmitter Output Pulse Position for Bit1	TPPos1	–0.20	0	0.20	ns	
Transmitter Output Pulse Position for Bit0	TPPos0	T / 7 – 0.20	T / 7	T / 7 + 0.20	ns	
Transmitter Output Pulse Position for Bit6	TPPos6	T × 2 / 7 – 0.20	T × 2 / 7	T × 2 / 7 + 0.20	ns	
Transmitter Output Pulse Position for Bit5	TPPos5	T × 3 / 7 – 0.20	T × 3 / 7	T × 3 / 7 + 0.20	ns	
Transmitter Output Pulse Position for Bit4	TPPos4	T × 4 / 7 – 0.20	T × 4 / 7	T × 4 / 7 + 0.20	ns	
Transmitter Output Pulse Position for Bit3	TPPos3	T × 5 / 7 – 0.20	T × 5 / 7	T × 5 / 7 + 0.20	ns	
Transmitter Output Pulse Position for Bit2	TPPos2	T × 6 / 7 – 0.20	T × 6 / 7	T × 6 / 7 + 0.20	ns	

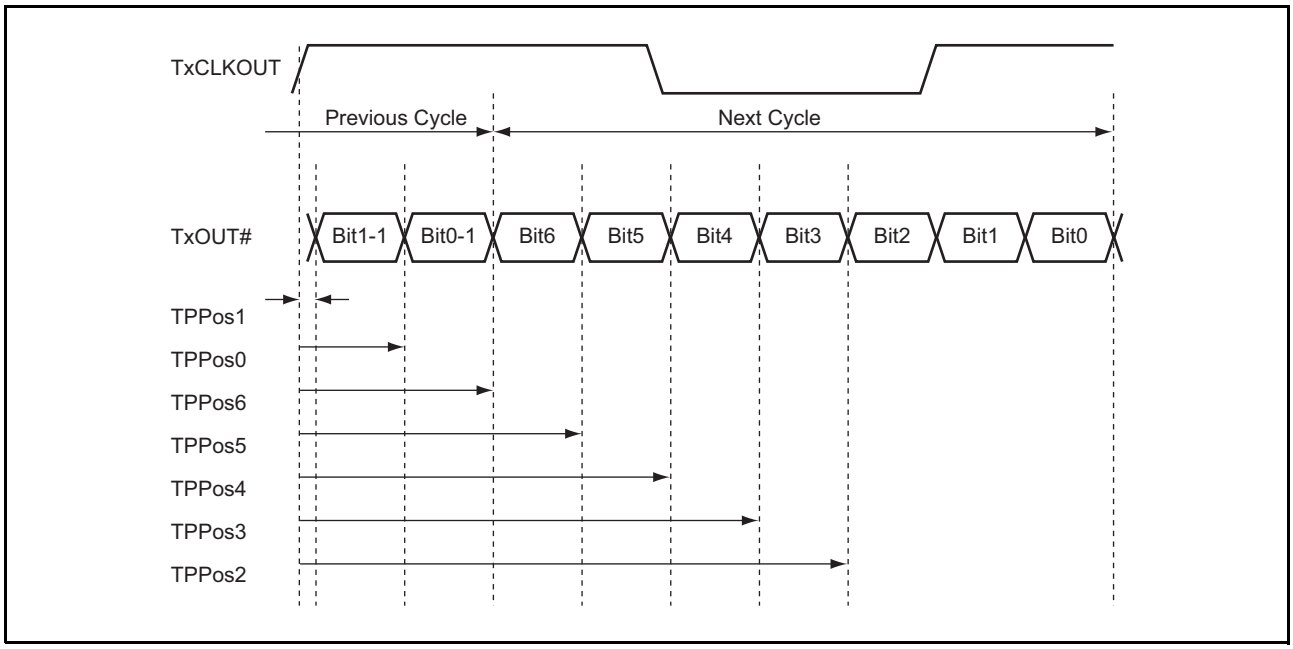


Figure 56.98 Transmitter LVDS Output Pulse Position Measurement

56.4.26 Capture Engine Unit Timing

Table 56.32 Capture Engine Unit Signal Timing

Item	Symbol	Min.	Max.	Unit	Figure
Vertical sync (VIO_VD) setup time (Camera clock rising)	t_{VVDS}	2	—	ns	Figure 56.99 (1) , Figure 56.99 (2)
Vertical sync (VIO_VD) setup time (Camera clock falling)	t_{VVDS}	2.5	—	ns	
Vertical sync (VIO_VD) hold time	t_{VVDH}	3.5	—	ns	
Horizontal sync (VIO_HD) setup time (Camera clock rising)	t_{VHDS}	2	—	ns	
Horizontal sync (VIO_HD) setup time (Camera clock falling)	t_{VHDS}	2.5	—	ns	
Horizontal sync (VIO_HD) hold time	t_{VHDH}	3.5	—	ns	
Capture image data (VIO_D) setup time (Camera clock rising)	t_{VDTS}	2	—	ns	
Capture image data (VIO_D) setup time (Camera clock falling)	t_{VDTS}	2.5	—	ns	
Capture image data (VIO_D) hold time	t_{VDTH}	3.5	—	ns	
Camera clock cycle	t_{VCYC}	11.50	—	ns	
Camera clock high level width	t_{VHW}	0.4 × t_{VCYC}	—	ns	
Camera clock low level width	t_{VLW}	0.4 × t_{VCYC}	—	ns	
Field identification signal (VIO_FLD) setup time (Camera clock rising)	t_{VFDS}	2	—	ns	
Field identification signal (VIO_FLD) setup time (Camera clock falling)	t_{VFDS}	2.5	—	ns	
Field identification signal (VIO_FLD) hold time	t_{VFDH}	3.5	—	ns	

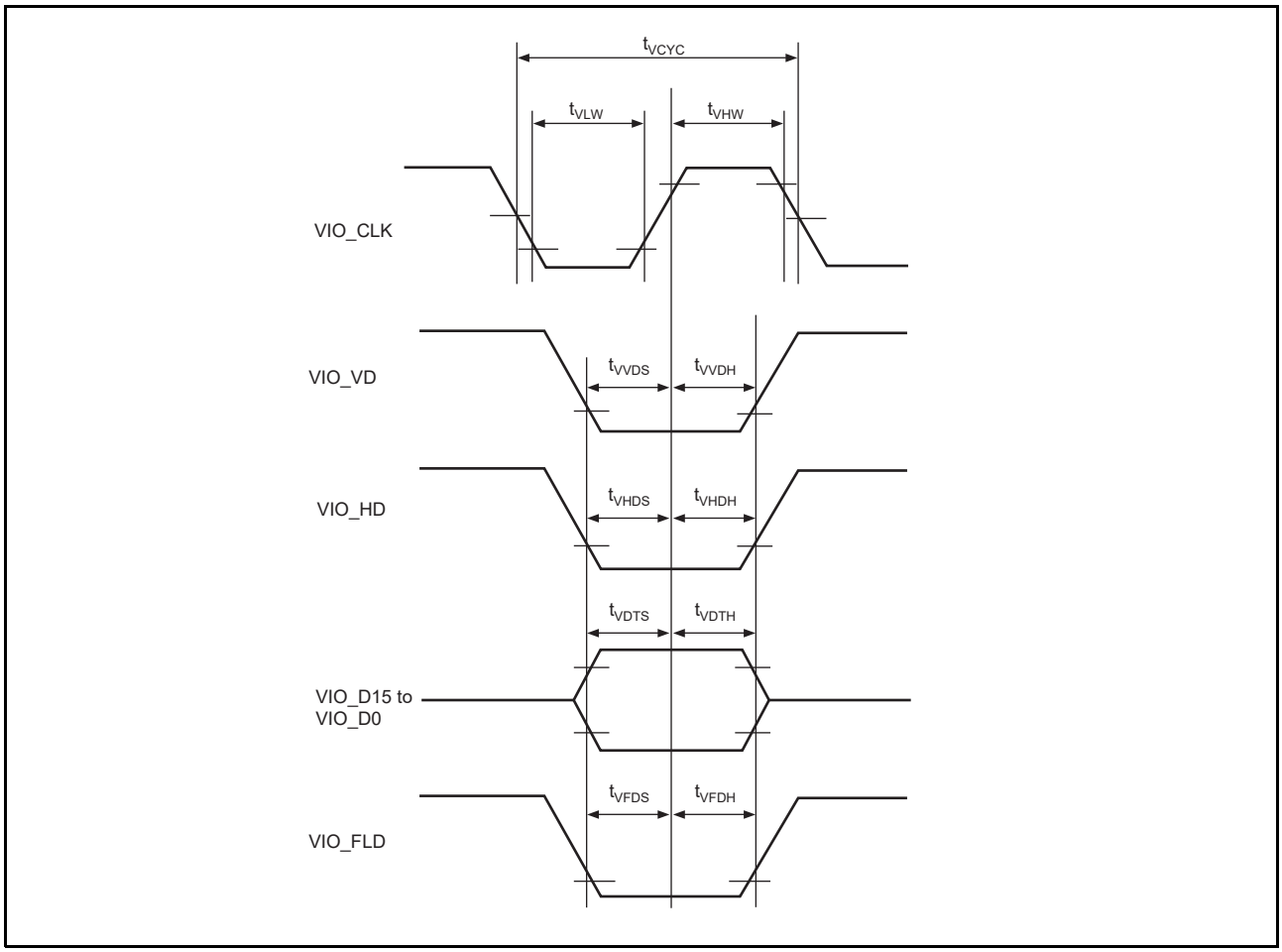


Figure 56.99 (1) Capture Engine Unit Module Signal Timing of data capturing on the rising edge of VIO_CLK

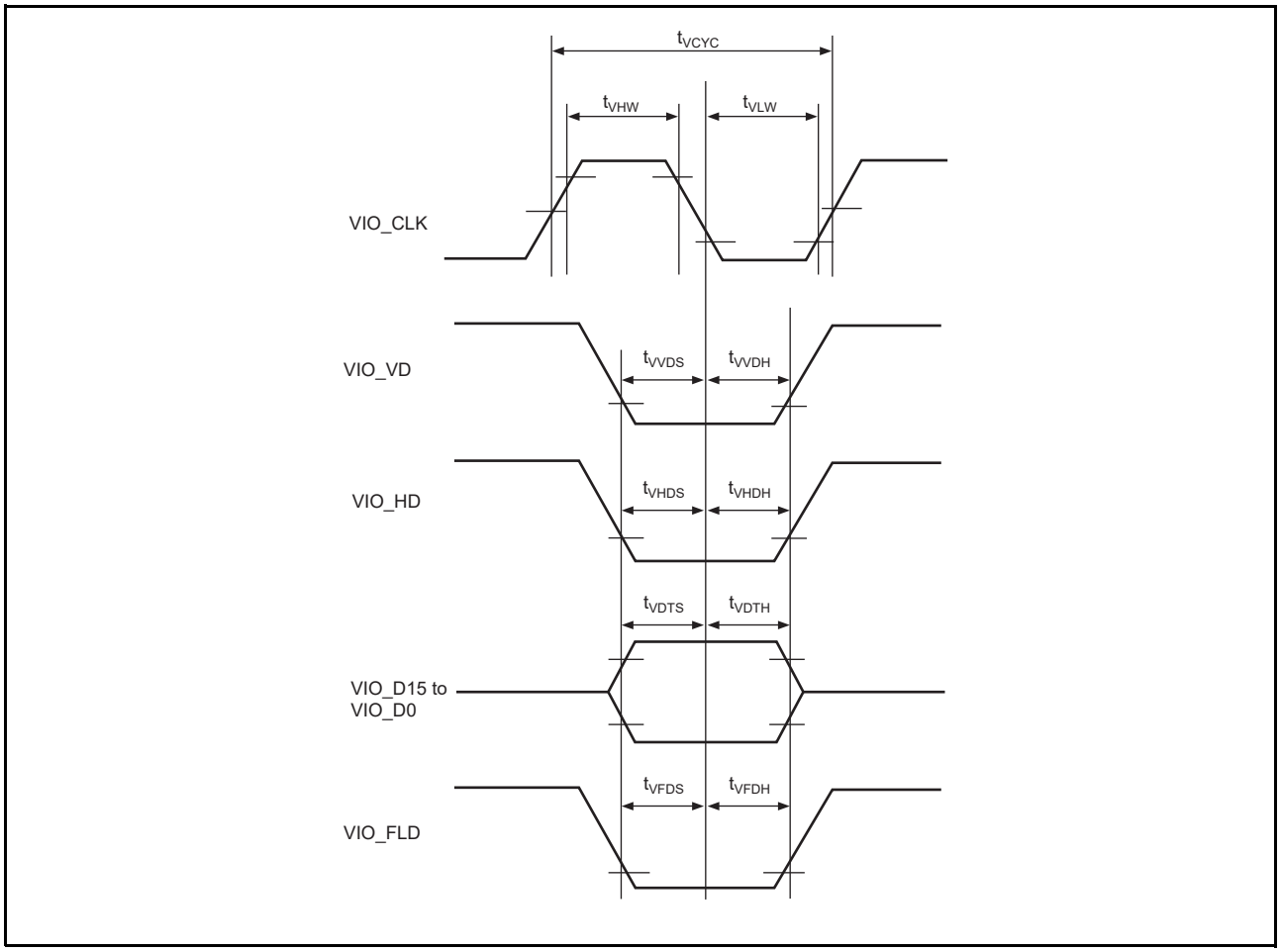


Figure 56.99 (2) Capture Engine Unit Module Signal Timing of data capturing on the falling edge of VIO_CLK

56.4.27 MIPI CSI-2 Interface Timing

Table 56.33 MIPI CSI-2 Interface Timing

Item	Symbol	Min.	Max.	Unit	Remarks	Figure
UI instantaneous	UI_{INST}	1	12.5	ns		Figure 56.100
Common–mode interference beyond 450 MHz	$\Delta V_{CMRX(HF)}$	—	100	mV	HS Receiver	
Common–mode interference 50MHz – 450MHz	$\Delta V_{CMRX(LF)}$	–50	50	mV	HS Receiver	
Setup time	T_{CDS}	0.15	—	UI	HS Receiver	Figure 56.100
Hold time	T_{CDH}	0.15	—	UI	HS Receiver	Figure 56.100

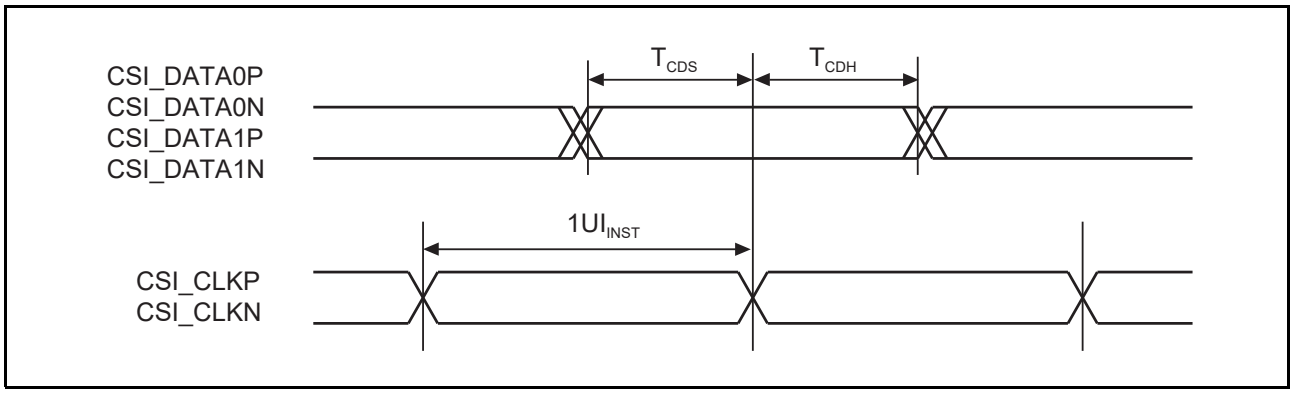


Figure 56.100 MIPI CSI-2 Interface Timing

56.4.28 SD/MMC Host Interface Timing

56.4.28.1 SD Interface

Table 56.34 SD/MMC Host Interface Timing (SD Default/High Speed mode 3.3-V power supply selection)

Item	Symbol	Min.	Max.	Unit	Figure
SD_CLK clock cycle	t_{SDPP}	4 × t_{bcyc}	—	ns	Figure 56.101
SD_CLK clock high level width	t_{SDWH}	0.4 × t_{SDPP}	—	ns	
SD_CLK clock low level width	t_{SDWL}	0.4 × t_{SDPP}	—	ns	
SD_CLK clock rise time	t_{SDLH}	—	3	ns	
SD_CLK clock fall time	t_{SDHL}	—	3	ns	
SD_CMD, SD_D7 to SD_D0 output data delay time	t_{SDODLY}	–8	4	ns	
SD_CMD, SD_D7 to SD_D0 input data setup time	t_{SDISU}	4.5	—	ns	
SD_CMD, SD_D7 to SD_D0 input data hold time	t_{SDIH}	2	—	ns	

Note: t_{bcyc} indicates internal bus clock (Bφ) cycle.

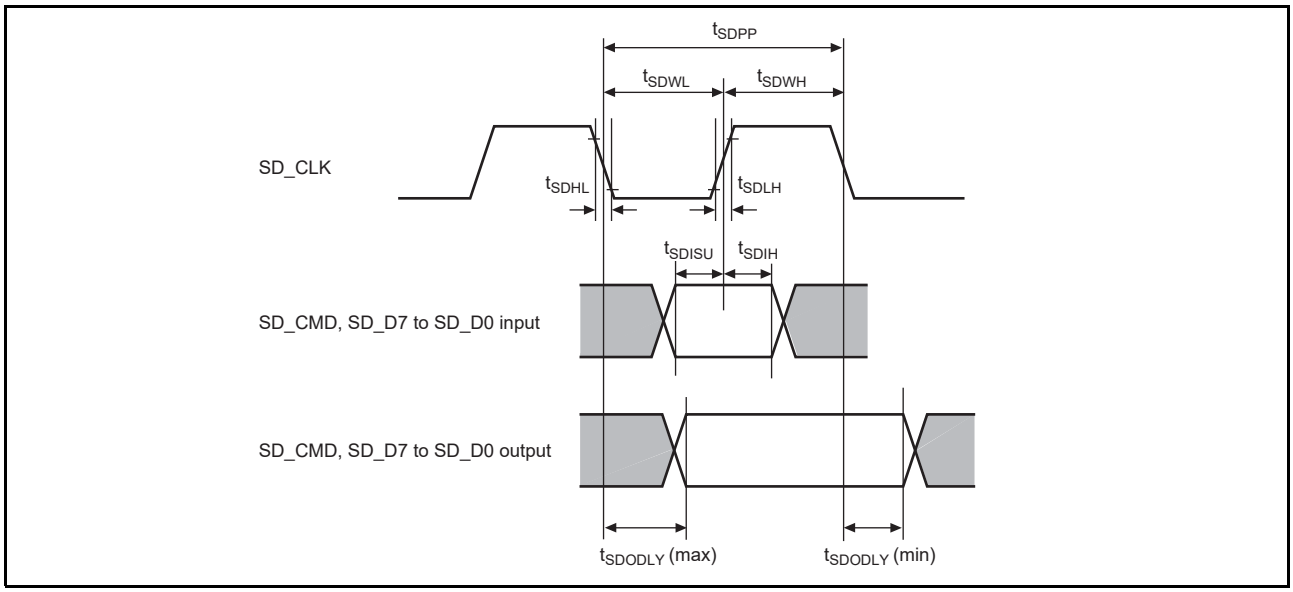


Figure 56.101 SD/MMC Host Interface (SD Interface)

Other characteristics of the SD interface are available upon non-disclosure agreement.
For details, contact your local sales representatives.

56.4.28.2 MMC Interface Timing

Table 56.35 SD/MMC Host Interface Timing (MMC Default/High Speed mode 3.3-V power supply selection)

Item	Symbol	Min.	Max.	Unit	Figure
SD_CLK clock cycle	t_{MMCPP}	4 × t_{bcyc}	—	ns	Figure 56.102
SD_CLK clock high level width	t_{MMCWH}	0.4 × t_{MMCPP}	—	ns	
SD_CLK clock low level width	t_{MMCWL}	0.4 × t_{MMCPP}	—	ns	
SD_CLK clock rise time	t_{MMCLH}	—	3	ns	
SD_CLK clock fall time	t_{MMCHL}	—	3	ns	
SD_CMD, SD_D7 to SD_D0 output data delay time	$t_{MMCODLY}$	–7	4	ns	
SD_CMD, SD_D7 to SD_D0 input data setup time	t_{MMCISU}	4.5	—	ns	
SD_CMD, SD_D7 to SD_D0 input data hold time	t_{MMCIH}	2	—	ns	

Note: t_{bcyc} indicates internal bus clock (Bφ) cycle.

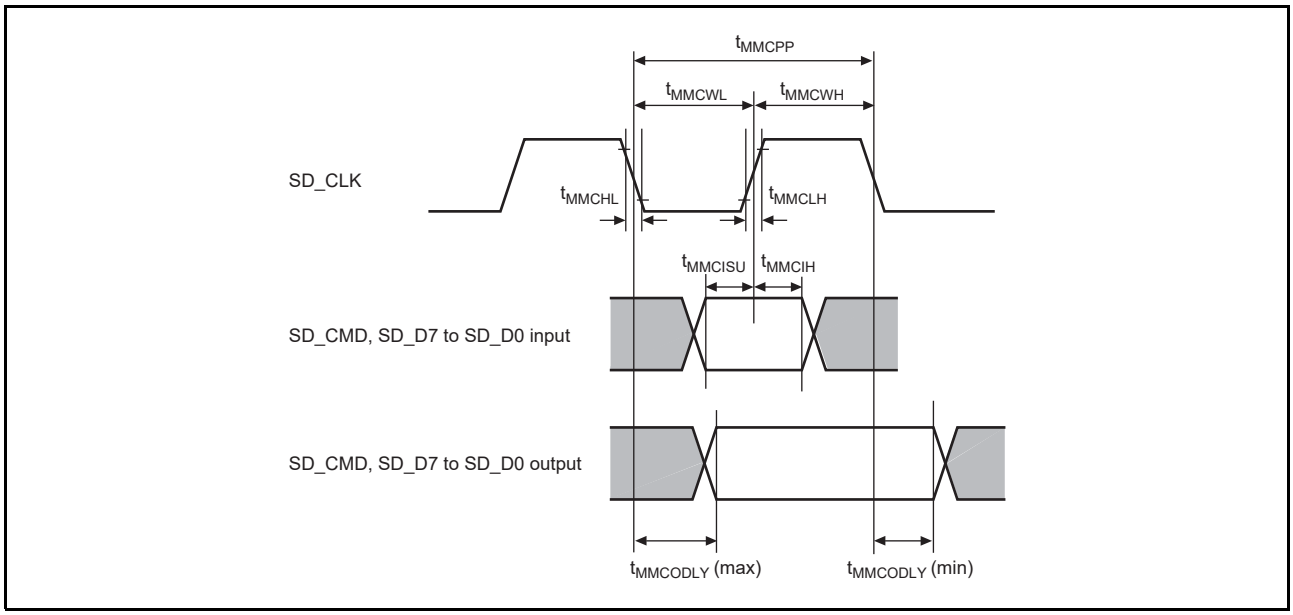


Figure 56.102 SD/MMC Host Interface (MMC Interface Default/High Speed mode 3.3-V power supply selection)

Table 56.36 SD/MMC Host Interface Timing (MMC HS-DDR mode 3.3-/1.8-V power supply selection)

Item	Symbol	Min.	Max.	Unit	Figure
SD_CLK clock cycle	t_{MMCPP}	4 × t_{bcyc}	—	ns	Figure 56.103, Figure 56.104
SD_CLK clock high level width	t_{MMCWH}	0.45 × t_{MMCPP}	—	ns	
SD_CLK clock low level width	t_{MMCWL}	0.45 × t_{MMCPP}	—	ns	
SD_CLK clock rise time	t_{MMCLH}	—	3	ns	
SD_CLK clock fall time	t_{MMCHL}	—	3	ns	
SD_CMD output data delay time	$t_{MMCODLY}$	–6.6	6.6	ns	Figure 56.103
SD_CMD input data setup time	t_{MMCISU}	5.5	—	ns	
SD_CMD input data hold time	t_{MMCIH}	2.5	—	ns	
SD_D7 to SD_D0 output data delay time	$t_{MMCODLYddr}$	2.5	0.5 × t_{bcyc} + 3	ns	Figure 56.104
SD_D7 to SD_D0 input data setup time	$t_{MMCISUddr}$	2.6	—	ns	
SD_D7 to SD_D0 input data hold time	$t_{MMCIHddr}$	1.5	—	ns	

Note: t_{bcyc} indicates internal bus clock (Bφ) cycle.

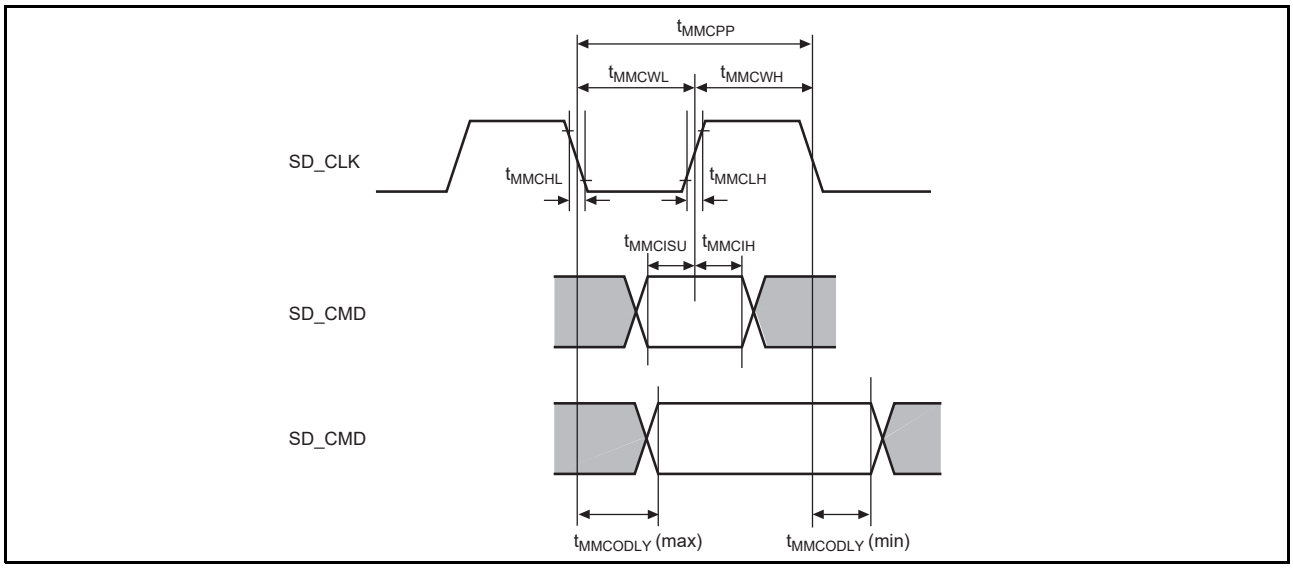


Figure 56.103 SD_CMD, SD/MMC Host Interface (MMC Interface HS-DDR mode 3.3-/1.8-V power supply selection)

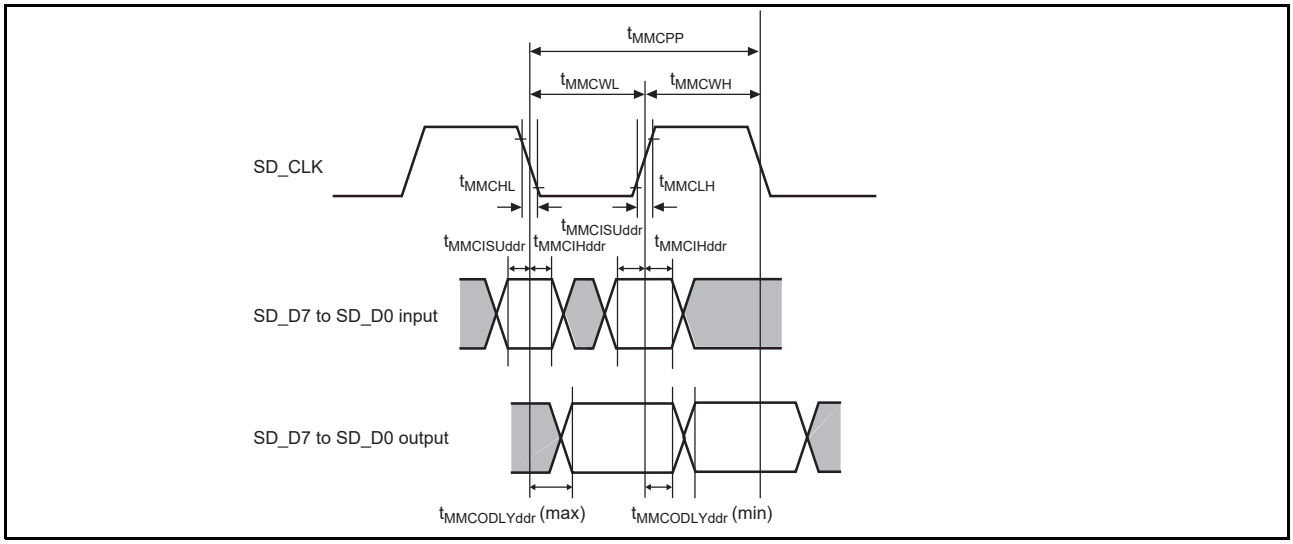


Figure 56.104 SD_D7 to SD_D0, SD/MMC Host Interface (MMC Interface HS-DDR mode 3.3-/1.8-V power supply selection)

Table 56.37 SD/MMC Host Interface Timing (MMC HS200 mode 1.8-V power supply selection, Output load: 15pF)

Item	Symbol	Min.	Max.	Unit	Figure
SD_CLK clock cycle	t_{MMCPP}	1 × t_{bcyc}*1	—	ns	Figure 56.105
SD_CLK clock high level width	t_{MMCWH}	0.35 × t_{MMCPP}	—	ns	
SD_CLK clock low level width	t_{MMCWL}	0.35 × t_{MMCPP}	—	ns	
SD_CLK clock rise time	t_{MMCLH}	—	1.5	ns	
SD_CLK clock fall time	t_{MMCHL}	—	1.5	ns	
SD_CMD, SD_D7 to SD_D0 output data delay time	$t_{MMCODLY}$	–1.7	1.1	ns	
SD_CMD, SD_D7 to SD_D0 input data setup time*2	t_{MMCISU}	—	—	ns	
SD_CMD, SD_D7 to SD_D0 input data hold time*2	t_{MMCIH}	—	—	ns	
SD_CMD, SD_D7 to SD_D0 input data width*2	t_{MMCIDW}	3.6	—	ns	

Note 1. t_{bcyc} indicates internal bus (Bφ) cycle.
Note 2. In HS200 mode, tuning by SCC is required. For details, see section 49.6, Sampling Clock Controller (SCC).

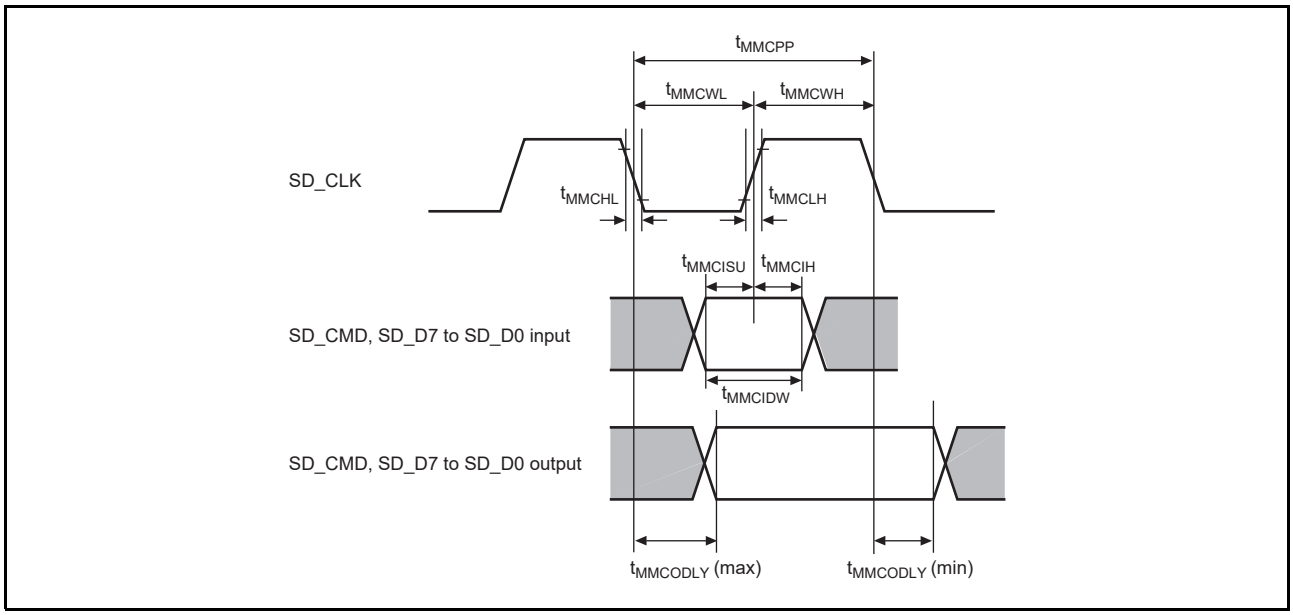


Figure 56.105 SD/MMC Host Interface (MMC Interface HS200 mode 1.8-V power supply selection)

56.4.29 General Purpose I/O Ports Timing

Table 56.38 General Purpose I/O Ports Timing

Item	Symbol	Min.	Max.	Unit	Figure
Output data delay time	t_{PORTD}	—	100	ns	Figure 56.106
Input data setup time	t_{PORTS}	100	—		
Input data hold time	t_{PORTH}	100	—		

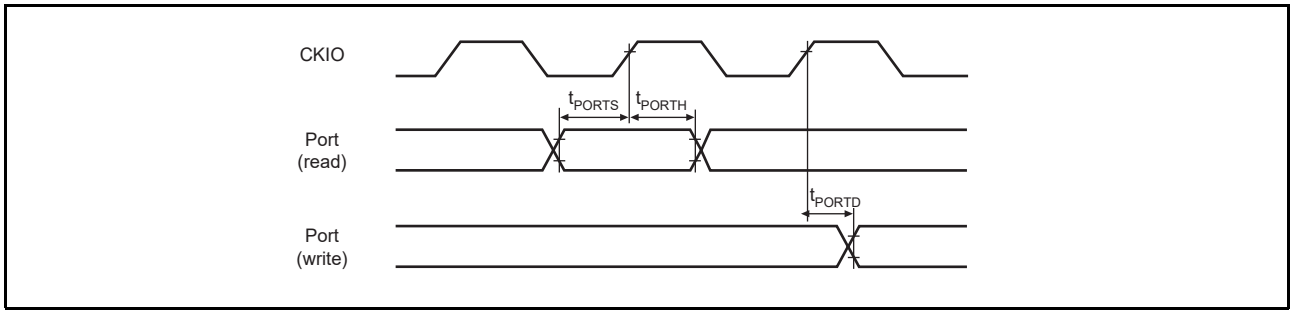


Figure 56.106 General I/O Ports Timing

56.4.30 Debugger Interface Timing

Table 56.39 Debugger Interface Timing

Item	Symbol	Min.	Max.	Unit	Figure
TCK cycle time	t_{TCKcyc}	50*1	—	ns	Figure 56.107
TCK high pulse width	t_{TCKH}	0.4	0.6	t_{TCKcyc}	
TCK low pulse width	t_{TCKL}	0.4	0.6	t_{TCKcyc}	
TDI setup time	t_{TDIS}	10	—	ns	Figure 56.108
TDI hold time	t_{TDIH}	10	—	ns	
TMS/SWDIO setup time	t_{TMSS}	10	—	ns	
TMS/SWDIO hold time	t_{TMSH}	10	—	ns	
SWDIO delay time	t_{SWDO}	—	16	ns	
TDO delay time	t_{TDOD}	—	16	ns	
Capture register setup time	t_{CAPTS}	10	—	ns	Figure 56.109
Capture register hold time	t_{CAPTH}	10	—	ns	
Update register delay time	$t_{UPDATED}$	—	20	ns	
Trace clock cycle	t_{TCYC}	7.57	—	ns	Figure 56.110 Output load: 15 pF
Trace clock high level	t_{THC}	2.5	—	ns	
Trace clock low level	t_{TLC}	2.5	—	ns	
Trace data delay time	t_{TDT}	0.1 × t_{TCYC} – 0.1	0.4 × t_{TCYC} – 0.2	ns	

Note 1. Should be greater than the peripheral clock 0 (P0φ) cycle time.

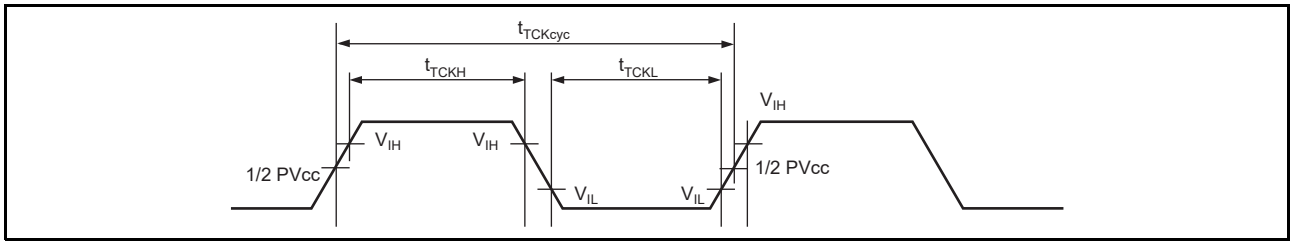


Figure 56.107 TCK Input Timing

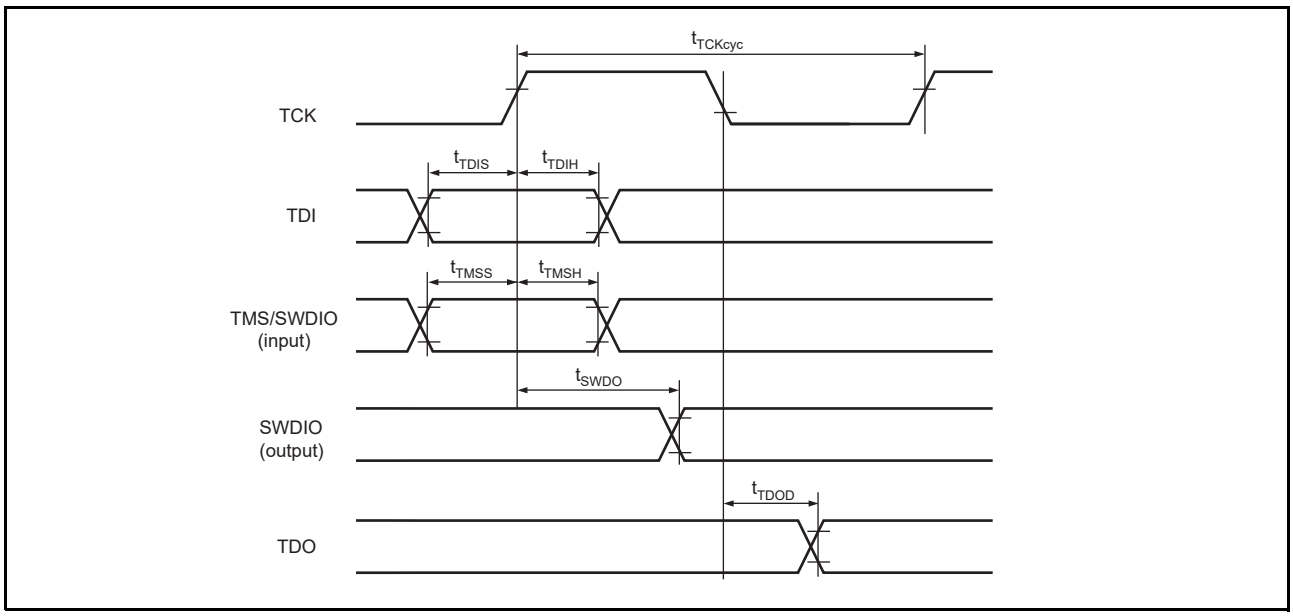


Figure 56.108 Data Transfer Timing

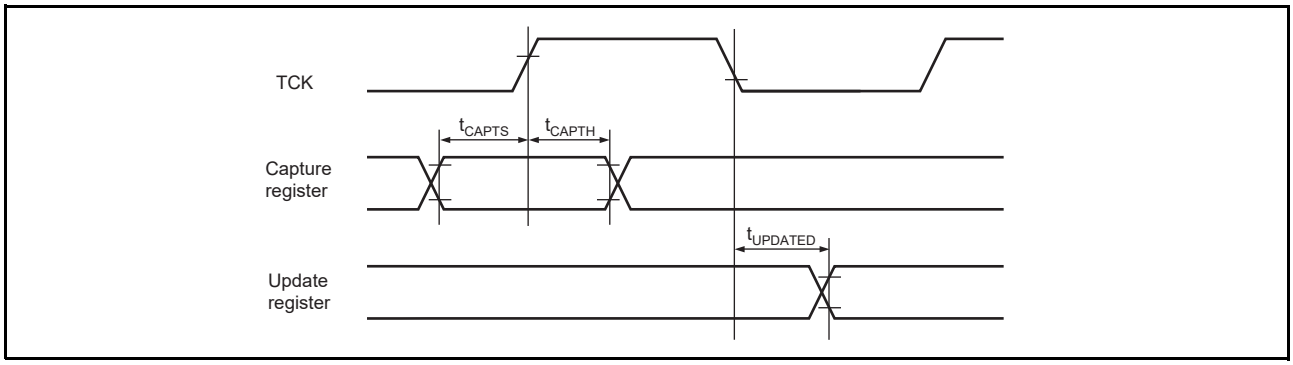


Figure 56.109 Boundary Scan Input/Output I/O Timing

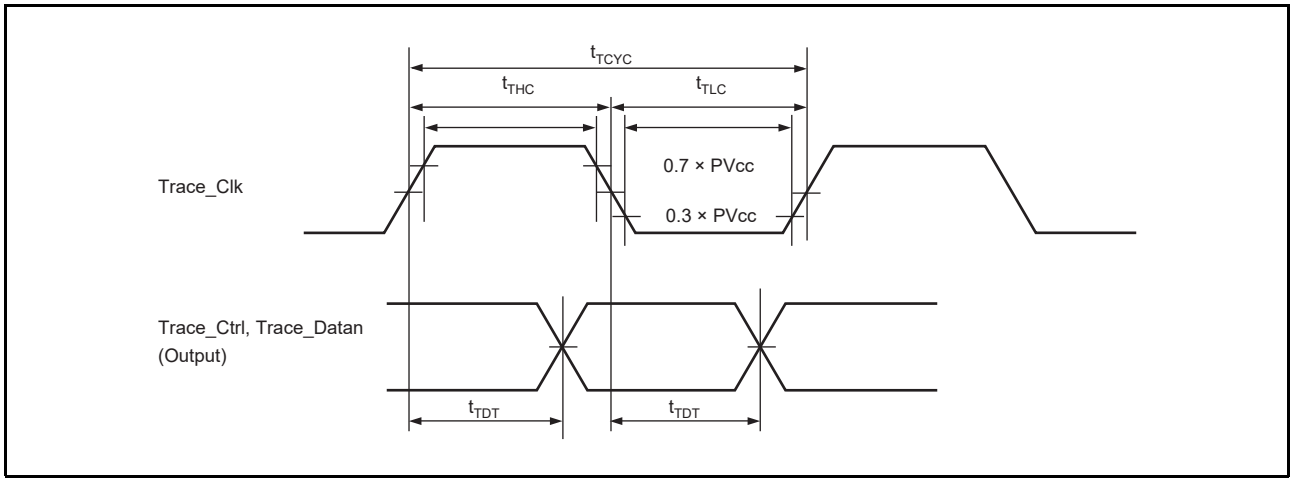


Figure 56.110 Trace Interface Timing

56.4.31 AC Characteristics Measurement Conditions

- I/O signal reference level: PVcc/2, PVcc_SPI/2, PVcc_HO/2, PVcc_SD0/2, PVcc_SD1/2, the minimum values of V_{IH}, V_{T+}, and V_{OH}, and the maximum values of V_{IL}, V_{T-}, and V_{OL} (refer to the individual timing chart)
- Input pulse level: PVcc, PVcc_SPI, PVcc_HO, PVcc_SD0, and PVcc_SD1
- Input rise and fall times: 1 ns

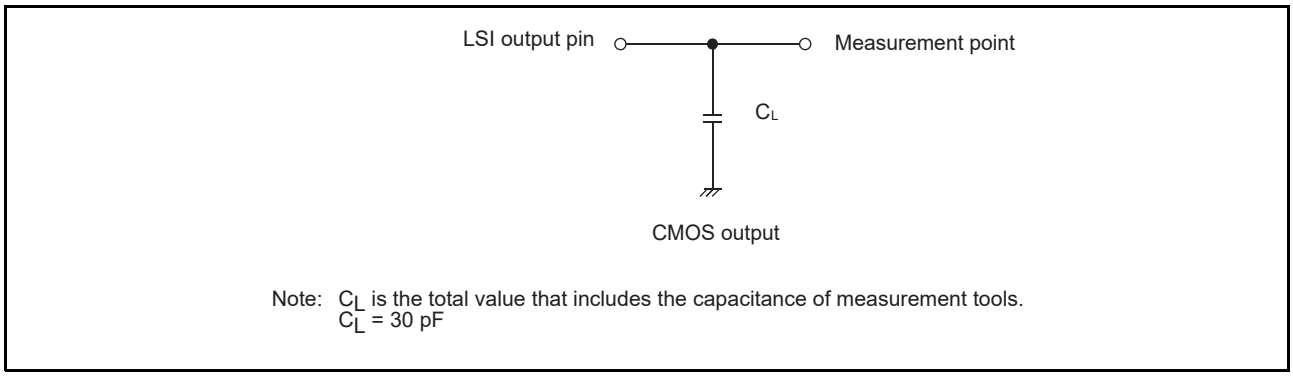


Figure 56.111 Output Load Circuit

56.5 A/D Converter Characteristics

Conditions: Vcc = 1.14 to 1.26 V, PVcc = USBDPVcc0 = USBDPVcc1 = AVcc = 3.0 to 3.6 V,
PVcc_SPI = 3.0 to 3.6 V/1.7 to 1.9 V, PVcc_SD0 = 3.0 to 3.6 V/1.7 to 1.9 V, PVcc_SD1 = 3.0 to 3.6V/1.7 to 1.9V, PLLVcc = 1.14 to 1.26 V, USBAPVcc0 = USBAPVcc1 = 3.0 to 3.6 V, LVDSAPVcc = 3.0 to 3.6 V,
LVDSPLLVcc = 1.14 to 1.26 V, PVcc_HO = MIPIAVcc18 = 1.7 to 1.9 V,
Vss = AVss = USBVss = 0 V
Ta = –40 to +85°C, Tj = –40 to +125°C

Table 56.40 A/D Converter Characteristics

Item	Min.	Typ.	Max.	Unit	Test Conditions
Resolution	8	—	12	bits	
Analog input capacitance	—	—	30	pF	
Conversion time*1 (P1φ = 66 MHz, P0φ = 33 MHz)	1	—	—	us	Sampling in 20 states
Permissible signal-source impedance	—	—	1	kΩ	
DNL	—	—	± 3	LSB	
INL	—	—	± 4	LSB	
Offset error	—	—	± 5.5	LSB	
Full-scale error	—	—	± 5.5	LSB	
Absolute accuracy	—	—	± 7	LSB	

Note: The above specification values apply when there is no access to the external bus during A/D conversion. If access proceeds during A/D conversion, values may not fall within the above ranges.

Note 1. The conversion time includes the sampling time and the comparison time (t_{SPL} + t_{SAM}).

57. States and Handling of Pins

This section describes pin states in each operating mode and how to handle pins.

57.1 Pin States

Table 57.1 shows the pin states in each operating mode.

As for the input/output functions, input buffers are listed on the upper column and output buffers on the lower column.

Table 57.1 Pin States

Pin Function			Pin State						
					Pin State Retained*2			Power-Down State	
					EBUSKEEPE*3 (Other than States at Right)				
Type	Pin Name		Normal State (Other than States at Right)	Power-On Reset*1	0	1	Power-On Reset*4	Deep Standby Mode	Software Standby Mode
Clock	EXTAL*6		I	I	I			I/Z*5	I/Z*5
	XTAL*6		O	O	O			O/Z*5	O/Z*5
	CKIO	Boot mode 0	O/Z*7	O	O	O/Z*7		O/Z*7	O/Z*7
		Other than Boot mode 0	O/Z*7	O	O/Z*7			O/Z*7	O/Z*7
	AUDIO_CLK (P6_4)		I	—	—			Z	Z
	AUDIO_CLK (PH_0)		I	—	—			I/Z*12	I
	AUDIO_X1*6		I	I	I			Z	Z
	AUDIO_X2*6		O	O	O			L	L
	AUDIO_XOUT		O	—	O/Z*9*15			O/Z*9*15	L/Z*9
System control	RES#		I	I	I			I	I
Operating mode control	MD_BOOT2, MD_BOOT1, MD_BOOT0, MD_CLK, MD_CLKS		—	I	—			—	—
	BSCANP		I	I	I			I	I
Interrupts	NMI		I	I	I			I	I
	IRQ0 (PJ_1, P6_2), IRQ1 (PE_1), IRQ2 (PH_1), IRQ3 (PH_0), IRQ4 (PG_2, PJ_5), IRQ5 (PG_6, PK_2), IRQ6 (P3_1, PK_4), IRQ7 (P3_3)		I	—	—			I/Z*12	I
	IRQ0 (other than PJ_1 and P6_2), IRQ1 (other than PE_1), IRQ2 (other than PH_1), IRQ3 (other than PH_0), IRQ4 (other than PG_2 and PJ_5), IRQ5 (other than PG_6 and PK_2), IRQ6 (other than P3_1 and PK_4), IRQ7 (other than P3_3)		I	—	—			Z	I

Table 57.1 Pin States

Pin Function			Pin State						
					Pin State Retained*2			Power-Down State	
					EBUSKEEPE*3 (Other than States at Right)				
Type	Pin Name		Normal State (Other than States at Right)	Power-On Reset*1	0	1	Power-On Reset*4	Deep Standby Mode	Software Standby Mode
Bus state controller	A25 to A21, A0		O	—	O/Z*10			O/Z*10	O/Z*10
	A20 to A1	Boot mode 0	O	—	O	O/Z*10		O/Z*10	O/Z*10
		Other than Boot mode 0	O	—	O/Z*10			O/Z*10	O/Z*10
	D0 to D15	Boot mode 0	I/Z	—	—			Z	Z
			O/Z	—	O/Z	Z		Z	Z
		Other than Boot mode 0	I/Z	—	—			Z	Z
			O/Z	—	Z			Z	Z
	CS0#, RD#	Boot mode 0	O	—	O	H/Z*10		H/Z*10	H/Z*10
		Other than Boot mode 0	O	—	H/Z*10			H/Z*10	H/Z*10
	CS5# to CS1#, RD/WR#, BS#, AH#, WE1#/DQMU, WE0#/DQML		O	—	H/Z*10			H/Z*10	H/Z*10
	WAIT#		I/Z	—	—			Z	Z
	RAS#, CAS#, CKE		O	—	O/Z*11			O/Z*11	O/Z*11
Direct memory access controller	DREQ0		I	—	—			Z	Z
	DACK0, TEND0		O	—	O/Z*9			O/Z*9	O/Z*9
Multi-function timer pulse unit 3	MTCLKA, MTCLKB, MTCLKC, MTCLKD, MTIC5U, MTIC5V, MTIC5W		I	—	—			Z	Z
	MTIOC1A, MTIOC1B (PH_1), MTIOC2B(P6_2), MTIOC3B, MTIOC4C		I	—	—			I/Z*12	I
			O/Z	—	O/Z*9			O/Z*9	O/Z*9
	MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D, MTIOC1B (PK_0), MTIOC2A, MTIOC2B (PH_4), MTIOC3A, MTIOC3C, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4D, MTIOC6A, MTIOC6B, MTIOC6C, MTIOC6D, MTIOC7A, MTIOC7B, MTIOC7C, MTIOC7D, MTIOC8A, MTIOC8B, MTIOC8C, MTIOC8D		I	—	—			Z	Z
			O/Z	—	O/Z*9			O/Z*9	O/Z*9
General PWM Timer	GTIOC0A, GTIOC2A, GTIOC4A (PH_0) , GTIOC4B (PH_1)		I	—	—			I/Z*12	I
			O/Z	—	O/Z*9			O/Z*9	O/Z*9
	GTIOC0B, GTIOC1A , GTIOC1B, GTIOC2B, GTIOC3A, GTIOC3B, GTIOC4A (P0_1), GTIOC4B (P0_2), GTIOC5A, GTIOC5B, GTIOC6A, GTIOC6B, GTIOC7A, GTIOC7B		I	—	—			Z	Z
			O/Z	—	O/Z*9			O/Z*9	O/Z*9
Port Output Enable3	POE8# (PE_1)		I	—	—			I/Z*12	I
	POE0#, POE4#, POE8# (PB_0), POE10#		I	—	—			Z	Z

Table 57.1 Pin States

Pin Function		Pin State					
				Pin State Retained*2		Power-Down State	
				EBUSKEEPE*3 (Other than States at Right)			
Type	Pin Name	Normal State (Other than States at Right)	Power-On Reset*1	0	1	Power-On Reset*4	Deep Standby Mode	Software Standby Mode
General PWM Timer Port Output Enable	GTETRGA, GTETRGB, GTETRGC, GTETRGD	I	—	—			Z	Z
Watchdog timer	WDTOVF#	O	—	H			H	H
Realtime clock	RTC_X1*6	I/Z*13	—	I/Z*13			I/Z*13	I/Z*13
	RTC_X2*6	O/Z*13	—	O/Z*13			O/Z*13	O/Z*13
Serial communication interface with FIFO	TxD0, TxD1, TxD2, TxD3, TxD4	O/Z	—	O/Z*9			O/Z*9	O/Z*9
	RxD1 (PJ_1), RxD2 (PE_1), RxD3 (P6_2)	I	—	—			I/Z*12	I
	RxD0, RxD1 (P7_1), RxD2 (PF_4), RxD3 (PF_1), RxD4	I	—	—			Z	Z
	RTS0#, RTS1#, RTS2#, CTS0#, CTS1#, CTS2#, SCK0, SCK1, SCK2, SCK3, SCK4	I	—	—			Z	Z
		O/Z	—	O/Z*9			O/Z*9	O/Z*9
Serial communication interface	SCI_TXD0, SCI_TXD1, SCI_RXD0, SCI_RXD1, SCI_CTS0#/RTS0#, SCI_CTS1#/RTS1#, SCI_SCK0, SCI_SCK1	I	—	—			Z	Z
		O/Z	—	O/Z*9			O/Z*9	O/Z*9
Renesas serial peripheral interface	MISO0 (P6_2, PK_0), MISO1 (P3_1), MISO2 (P3_3), RSPCK0 (PG_6), RSPCK2 (PK_2)	I	—	—			I/Z*12	I
		O/Z	—	O/Z*9			O/Z*9	O/Z*9
	MISO0 (P8_5), MISO1 (PF_2), MISO2 (PC_2), RSPCK0 (P8_7, PE_0), RSPCK1, RSPCK2 (PC_0), SSL00, SSL10, SSL20, MOSI0 to MOSI2	I	—	—			Z	Z
		O/Z	—	O/Z*9			O/Z*9	O/Z*9

Table 57.1 Pin States

Pin Function			Pin State						
					Pin State Retained*2			Power-Down State	
					EBUSKEEPE*3 (Other than States at Right)				
Type	Pin Name		Normal State (Other than States at Right)	Power-On Reset*1	0	1	Power-On Reset*4	Deep Standby Mode	Software Standby Mode
SPI multi I/O bus controller	QSPI0_SPCLK, QSPI1_SPCLK, QSPI0_SSL, RPC_WP#	Boot mode 3, 4, 5	O	Z	O	O/Z*9		O/Z*9	O/Z*9
		Other than Boot mode 3, 4, 5	O	Z	O/Z*9			O/Z*9	O/Z*9
	RPC_RESET#	Boot mode 3, 4, 5	O	L	O	O/Z*9		O/Z*9	O/Z*9
		Other than Boot mode 3, 4, 5	O	L	O/Z*9			O/Z*9	O/Z*9
	QSPI0_IO0, QSPI0_IO1, QSPI0_IO2, QSPI0_IO3, QSPI1_IO0, QSPI1_IO1, QSPI1_IO2, QSPI1_IO3, QSPI1_SSL	Boot mode 3, 4, 5	I/Z	Z	I		Z	Z	Z
			O/Z	Z	O/Z	O/Z*9		O/Z*9	O/Z*9
		Other than Boot mode 3, 4, 5	I/Z	Z	I		Z	Z	Z
			O/Z	Z	O/Z*9			O/Z*9	O/Z*9
	RPC_INT#		I/Z	Z	I		Z	Z	Z
HyperBus controller/ Octa memory controller	HM_CK/ OM_SCLK, HM_CK#, HM_CS0#/ OM_CS0#, HM_CS1#/ OM_CS1#	Boot mode 6, 7	O	Z	O	O/Z*9		O/Z*9	O/Z*9
		Other than Boot mode 6, 7	O	Z	O/Z*9			O/Z*9	O/Z*9
	HM_RESET#/ OM_RESET#	Boot mode 6, 7	O	L	O	O/Z*9		O/Z*9	O/Z*9
		Other than Boot mode 6, 7	O	L	O/Z*9			O/Z*9	O/Z*9
	HM_RWDS/ OM_DQS, HM_DQ0/ OM_SIO0, HM_DQ1/ OM_SIO1, HM_DQ2/ OM_SIO2, HM_DQ3/ OM_SIO3, HM_DQ4/ OM_SIO4, HM_DQ5/ OM_SIO5, HM_DQ6/ OM_SIO6, HM_DQ7/ OM_SIO7	Boot mode 6, 7	I	Z	I		Z	Z	Z
			O/Z	Z	O/Z	O/Z*9		O/Z*9	O/Z*9
		Other than Boot mode 6, 7	I	Z	I		Z	Z	Z
			O/Z	Z	O/Z*9			O/Z*9	O/Z*9
	HM_RSTO#, HM_INT#/ OM_ECS#		I	—	—			Z	Z

Table 57.1 Pin States

Pin Function		Pin State					
				Pin State Retained*2		Power-Down State	
				EBUSKEEPE*3 (Other than States at Right)			
Type	Pin Name	Normal State (Other than States at Right)	Power-On Reset*1	0	1	Power-On Reset*4	Deep Standby Mode	Software Standby Mode
I2C bus interface	RIIC0SCL to RIIC3SCL, RIIC0SDA to RIIC3SDA	I/Z	—	—			Z	Z
		L/Z	—	Z			Z	Z
Serial sound interface	SSITxD0, SSITxD3, SSITxD1	O/Z	—	O/Z*9			O/Z*9	O/Z*9
	SSIRxD0, SSIRxD1, SSIRxD3	I	—	—			Z	Z
	SSIBCK0, SSILRCK0, SSIBCK2, SSILRCK2 (PB_4, P9_1), SSIDATA2, SSIBCK3, SSILRCK3, SSIBCK1, SSILRCK1	I/Z	—	—			Z	Z
		O/Z	—	O/Z*9			O/Z*9	O/Z*9
	SSILRCK2 (PJ_5)	I/Z	—	—			I/Z*12	I
		O/Z	—	O/Z*9			O/Z*9	O/Z*9
CANFD interface	CAN0TX, CAN0TX_DATARATE_EN, CAN1TX, CAN1TX_DATARATE_EN, CAN0RX_DATARATE_EN, CAN1RX_DATARATE_EN	O	—	O/Z*9			O/Z*9	O/Z*9
	CAN0RX (PK_2), CAN1RX (P3_1)	I	—	—			I/Z*12	I
	CAN1RX (P2_0), CAN_CLK, CAN0RX (P1_1)	I	—	—			Z	Z
Renesas SPDIF interface	SPDIF_OUT	O	—	O/Z*9			O/Z*9	O/Z*9
	SPDIF_IN (PC_4)	I	—	—			Z	Z
	SPDIF_IN (PJ_1)	I	—	—			I/Z*12	I
Ethernet controller 0	ET0_TXEN/RMII0_TXD_EN, ET0_TXD0/RMII0_TXD0, ET0_TXD1/RMII0_TXD1, ET0_TXD2, ET0_TXD3, ET0_TXER, ET0_MDC , ET0_WOL	O	—	O/Z*9			O/Z*9	O/Z*9
	ET0_EXOUT, ET0_SCLKIN	I/Z	—	—			Z	Z
		O	—	O/Z*9			O/Z*9	O/Z*9
	ET0_TXCLK, ET0_RXCLK/REF50CK0, ET0_RXDV, ET0_RXD1/RMII0_RXD1, ET0_CRS/RMII0_CRS_DV, ET0_COL, ET0_RXD3, ET0_LINKSTA, ET0_RXER/RMII0_RXER	I	—	—			Z	Z
	ET0_RXD0/RMII0_RXD0, ET0_RXD2	I	—	—			I/Z*12	I
	ET0_MDIO	I	—	—			Z	Z
		O	—	O/Z*9			O/Z*9	O/Z*9

Table 57.1 Pin States

Pin Function		Pin State					
				Pin State Retained*2		Power-Down State	
				EBUSKEEPE*3 (Other than States at Right)			
Type	Pin Name	Normal State (Other than States at Right)	Power-On Reset*1	0	1	Power-On Reset*4	Deep Standby Mode	Software Standby Mode
Ethernet controller 1	ET1_TXEN/RMII1_TXD_EN, ET1_TXD0/RMII1_TXD0, ET1_TXD1/RMII1_TXD1, ET1_TXD2, ET1_TXD3, ET1_TXER, ET1_MDC, ET1_WOL	O	—	O/Z*9			O/Z*9	O/Z*9
	ET1_EXOUT, ET1_SCLKIN	I/Z	—	—			Z	Z
		O	—	O/Z*9			O/Z*9	O/Z*9
	ET1_TXCLK, ET1_RXCLK/REF50CK1, ET1_RXDV, ET1_CRS/RMII1_CRS_DV, ET1_COL, ET1_RXD2, ET1_RXD3, ET1_RXD1/RMII1_RXD1, ET1_LINKSTA, ET1_RXD0/RMII1_RXD0	I	—	—			Z	Z
	ET1_RXER/RMII1_RXER	I	—	—			I/Z*12	I
	ET1_MDIO	I	—	—			Z	Z
		O	—	O/Z*9			O/Z*9	O/Z*9
A/D converter	AN007 to AN000	I/Z	—	—			Z	Z
	ADTRG#	I	—	—			Z	Z
NAND flash controller	NFRB#	I	—	—			Z	Z
	NFCE#, NFALE, NFRE#, NFCLE, NFWE#	O	—	O/Z*9			O/Z*9	O/Z*9
	NFDATA3 (PK_0), NFDATA5 (PK_2), NFDATA2 (PJ_5)	I	—	—			I/Z*12	I
		O/Z	—	O/Z*9			O/Z*9	O/Z*9
	NFDATA0, NFDATA1, NFDATA2 (PH_5), NFDATA3 (PH_6), NFDATA4, NFDATA5 (PC_0), NFDATA6, NFDATA7	I	—	—			Z	Z
		O/Z	—	O/Z*9			O/Z*9	O/Z*9

Table 57.1 Pin States

Pin Function		Pin State					
				Pin State Retained*2		Power-Down State	
				EBUSKEEPE*3 (Other than States at Right)			
Type	Pin Name	Normal State (Other than States at Right)	Power-On Reset*1	0	1	Power-On Reset*4	Deep Standby Mode	Software Standby Mode
USB 2.0 host/ function module	DP0, DP1, DM0, DM1	I/Z	Z	I/Z			I/Z	I/Z
		O/Z	Z	O/Z			O/Z	O/Z
	RREF0, RREF1	I	I	I			I	I
	USB_X1*6	I	I	I			Z	Z
	USB_X2*6	O	O	O			L	L
	VBUSIN0 (other than PJ_1), VBUSIN1 (other than PE_1), OVRCUR0 (other than PJ_5), OTG_ID0, CC1_Rd0 (other than PH_0), CC1_Ra0, CC2_Rd0 (other than PH_1), CC2_Ra0, OVRCUR1 (other than PK_4), OTG_ID1, CC1_Rd1, CC1_Ra1, CC2_Rd1, CC2_Ra1	I	—	—			Z	I
	VBUSIN0 (PJ_1), VBUSIN1 (PE_1), CC1_Rd0 (PH_0), CC2_Rd0 (PH_1), OVRCUR0 (PJ_5), OVRCUR1 (PK_4)	I	—	—			I/Z*12	I
	VBUSEN0, VBUSEN1, OTG_EXICEN0, OTG_EXICEN1	O/Z	—	O/Z*9			O/Z*9	O/Z*9
Video display controller 6	LCD0_CLK, LCD0_DATA0 to LCD0_DATA23, LCD0_TCON0 to LCD0_TCON6	O	—	O/Z*9			O/Z*9	O/Z*9
	LCD0_EXTCLK, DV0_CLK, DV0_DATA0 to DV0_DATA23, DV0_VSYNC, DV0_HSYNC	I	—	—			Z	Z
LVDS output interface	TXOUT0M, TXOUT0P, TXOUT1M, TXOUT1P, TXOUT2M, TXOUT2P, TXCLKOUTM, TXCLKOUTP	O	—	Z			Z	Z
Capture engine unit	VIO_CLK, VIO_FLD, VIO_D2 to VIO_D6, VIO_D8, VIO_D9, VIO_D11, VIO_D12, VIO_D14, VIO_D15, VIO_HD	I	—	—			Z	Z
	VIO_VD, VIO_D0, VIO_D1, VIO_D7, VIO_D10, VIO_D13	I	—	—			I/Z*12	I

Table 57.1 Pin States

Pin Function			Pin State						
					Pin State Retained*2			Power-Down State	
			Normal State (Other than States at Right)	Power-On Reset*1	EBUSKEEPE*3 (Other than States at Right)		Power-On Reset*4	Deep Standby Mode	Software Standby Mode
Type	Pin Name				0	1			
SD/MMC host interface	SD_CLK_0	Boot mode 1	O	Z	O	O/Z*9		O/Z*9	O/Z*9
		Other than Boot mode 1	O	Z	O/Z*9			O/Z*9	O/Z*9
	SD0_RST#	Boot mode 1	O	L	O	O/Z*9		O/Z*9	O/Z*9
		Other than Boot mode 1	O	L	O/Z*9			O/Z*9	O/Z*9
	SD_CMD_0, SD_D0_0, SD_D1_0, SD_D2_0, SD_D3_0	Boot mode 1	I	Z	I		Z	Z	Z
			O/Z	Z	O/Z	O/Z*9		O/Z*9	O/Z*9
		Other than Boot mode 1	I	Z	I		Z	Z	Z
			O/Z	Z	O/Z*9			O/Z*9	O/Z*9
	SD_CLK_1		O	Z	O/Z*9			O/Z*9	O/Z*9
	SD_CMD_1, SD_D0_1 to SD_D3_1		I	Z	—			Z	Z
			O/Z	Z	O/Z*9			O/Z*9	O/Z*9
	SD_CD_0 (P3_3)		I	—	—			I/Z*12	I
	SD_CD_0 (other than P3_3), SD_WP_0, SD_CD_1, SD_WP_1		I	—	—			Z	Z
DRP*17	DRP00 to DRP31		I/Z	—	—			Z	Z
			O	—	O/Z*9			O/Z*9	O/Z*9
MIPI CSI2 interface	CSI_CLKP, CSI_CLKN, CSI_DATA0P, CSI_DATA0N, CSI_DATA1P, CSI_DATA1N		I/Z*16	Z	Z			Z	Z
General I/O port	PM_0, P5_0 to P5_7, PL_0 to PL_4		I/Z*8	Z	Z			Z	Z
	PM_1		O/Z	Z	O/Z*9			O/Z*9	O/Z*9
	PD_0, PD_1, PD_2, PD_3, PD_4, PD_5, PD_6, PD_7		I/Z*8	Z	Z			Z	Z
			L/Z	Z	Z			Z	Z
	P3_1, P3_3, P6_2, PE_1, PE_6, PH_0, PG_2, PG_6, PJ_1, PK_0, PK_2, PK_4		I	Z	Z			I/Z*12	I
			O/Z	Z	O/Z*9			O/Z*9	O/Z*9
	Other than the above		I/Z*8	Z	Z			Z	Z
			O/Z	Z	O/Z*9			O/Z*9	O/Z*9
Debugger interface	TDI		I	I	I			Z	I
	TDO/SWO		O/Z*14	O/Z*14	O/Z*14			Z	O/Z*14
	TMS/SWDIO		I	I	I			Z	I
			O/Z	O/Z	O/Z			Z	O/Z
	TCK		I	I	I			Z	I
	TRST#		I	I	I			Z	I
	TRACEDATA3 to TRACEDATA0, TRACECLK, TRACECTL		O	—	O/Z*9			O/Z*9	O/Z*9

[Legend]
I: Input
O: Output
H: High-level output
L: Low-level output
Z: High-impedance
-: Condition under which the pin function is not selectable

Note 1. Indicates the power-on reset by low-level input to the RES# pin. The pin states after a power-on reset by the watchdog timer overflow is the same as the initial pin states at normal operation (see section 51, GPIO).
Note 2. After the chip has been released from deep standby mode by the input on pins for canceling the standby mode such as NMI or by the realtime clock alarm interrupt, the pins retain the state until the IOKEEP bit in the deep standby cancel source flag register (DSFR) is cleared (see section 52, Power-Down Modes).
Note 3. The EBUSKEEPE bit in deep standby control register (DSTCR) (see section 52, Power-Down Modes).
Note 4. This LSI enters the power-on reset state for a certain period after recovery from deep standby control mode (see section 52, Power-Down Modes).
Note 5. Depends on the setting of the RTC1XT and RTC0XT bits in the RTCXTAL select register (RTCXTALSEL) (see section 52, Power-Down Modes).
Note 6. When pins for the connection with a crystal resonator are not used, the input pins (EXTAL, RTC_X1, AUDIO_X1, USB_X1) must be fixed (pull-up/down resistor, power supply, or ground.) and the output pins (XTAL, RTC_X2, AUDIO_X2, USB_X2) must be open.
Note 7. Depends on the setting of the CKOEN bit in the frequency control register (FRQCR) of the clock pulse generator (see section 6, Clock Pulse Generator).
Note 8. Depends on the setting in the port direction register (PDR) of the general I/O ports.
Note 9. Depends on the setting of the HIZ bit in the standby control register 2 (STBCR2) (see section 52, Power-Down Modes).
Note 10. Depends on the setting of the HIZMEM bit in the common control register (CMNCR) of the bus state controller (see section 8, Bus State Controller).
Note 11. Depends on the setting of the HIZCNT bit in the common control register (CMNCR) of the bus state controller (see section 8, Bus State Controller).
Note 12. Depends on the setting of the corresponding bit in the deep standby cancel source select register (DSSSR) (see section 52, Power-Down Modes).
Note 13. Depends on the setting of the RTCEN bit in the realtime clock control register 3 (RCR3) (see section 16, Realtime Clock (RTC)).
Note 14. O in serial wire debug (SWD) mode. In modes other than serial wire debug (SWD), Z when the TAP controller is neither the Shift-DR nor Shift-IR state.
Note 15. When this is an output, the output is fixed to either the High or Low level. There is no oscillation.
Note 16. Depends on the setting of the SHUTDOWNZ bit in the PHY Operation Control Register (PHYCNT) (see section 47, MIPI CSI2 Interface).
Note 17. Only in products with a DRP

57.2 Treatment of Unused Pins

How unused pins are to be handled is indicated below.

Table 57.2 Handling of Unused Pins (Except for Debugger Interface Pins)

Pin Name	Handling
NMI	Fix this pin at a high level (pull up or connect to a power supply).
DP1, DP0, DM1, DM0	Connect these pins, via a 10 kΩ resistor, to GND.
RREF0, RREF1	Open-circuit
Dedicated to the USB (USBAPVcc0, USBAPVcc1, USBDPVcc0, USBDPVcc1)	Supply power at 3.3 V.
Dedicated USB ground (USBVss)	Connect to ground.
Dedicated A/D power (AVcc)	Supply power at 3.3 V.
Dedicated A/D ground (AVss)	Connect to ground.
LVDS analog power supply (LVDSAPVcc)	Supply power at 3.3 V.
LVDS PLL power supply (LVDSPLLVcc)	Supply power at 1.2 V.
MIPIAVcc18	Supply power at 1.8 V.
CSI_CLKP, CSI_CLKN, CSI_DATA0P, CSI_DATA0N, CSI_DATA1P, CSI_DATA1N	Connect to ground.
PVcc_SD0, PVcc_SD1	Supply power at 3.3 V.
Input/output pins driven by the PVcc_SD0 or PVcc_SD1 power supply (SD0_CMD, SD0_DAT7 to SD0_DAT0, SD1_CMD, SD1_DAT3 to SD1_DAT0)	Fix the levels on the pins (pull them up or down)*1
PVcc_SPI	Supply power at 3.3 V.
Input/output pins driven by the PVcc_SPI power supply (QSPI0_IO3 to QSPI0_IO0, QSPI1_IO3 to QSPI1_IO0, QSPI1_SSL)	Open-circuit
PVcc_HO	Supply power at 1.8 V.
Input/output pins driven by the PVcc_HO power supply (HM_RWDS/OM_DQS, HM_DQ7/OM_SIO7 to HM_DQ0/ OM_SIO0)	Fix the levels on the pins (pull them up or down)*1 *3
Dedicated input pins other than those listed above	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level)*1.
Input/output pins other than those listed above	Make the input-pin settings and then fix the level (pull them up or down)*2; alternatively, make the output-pin settings and leave the pins open-circuit.
Dedicated output pins	Open-circuit

Note 1. We recommend that the values of pull-up or pull-down resistors are in the range from 4.7 kΩ to 100 kΩ.
Note 2. By setting the ports in accord with section 51, GPIO, setting of a fixed level can be made unnecessary for some pins. For details, see section 51.3.1, Port Direction Register (PDR).
Note 3. In the case of a pull-up resistor, connection is to the 1.8-V power supply.

Table 57.3 Handling of Debugger Interface Pins (when Emulator is not Used)

Pin	Handling
BSCANP	Fix this pin at a low level (pull down or connect to the ground level).
TRST#	Fix this pin at a low level (pull down or connect to the ground level). Or connect to another pin which operates in the same manner as the RES# pin
TCK, TMS, TDI	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level).
TDO	Open-circuit

Note 1. When using the emulator, handle these pins as described in the manual for the emulator.
Note 2. We recommend that the values of pull-up or pull-down resistors are in the range from 4.7 kΩ to 100 kΩ.

57.3 Handling of Pins in Deep Standby Mode

How pins are to be handled in deep standby mode is indicated below.
For the states of pins in deep standby mode, refer to the corresponding items under section 57.1, Pin States. Handling of unused pins as described under section 57.2, Treatment of Unused Pins, also applies in deep standby mode.

Table 57.4 Handling of Pins in Deep Standby Mode

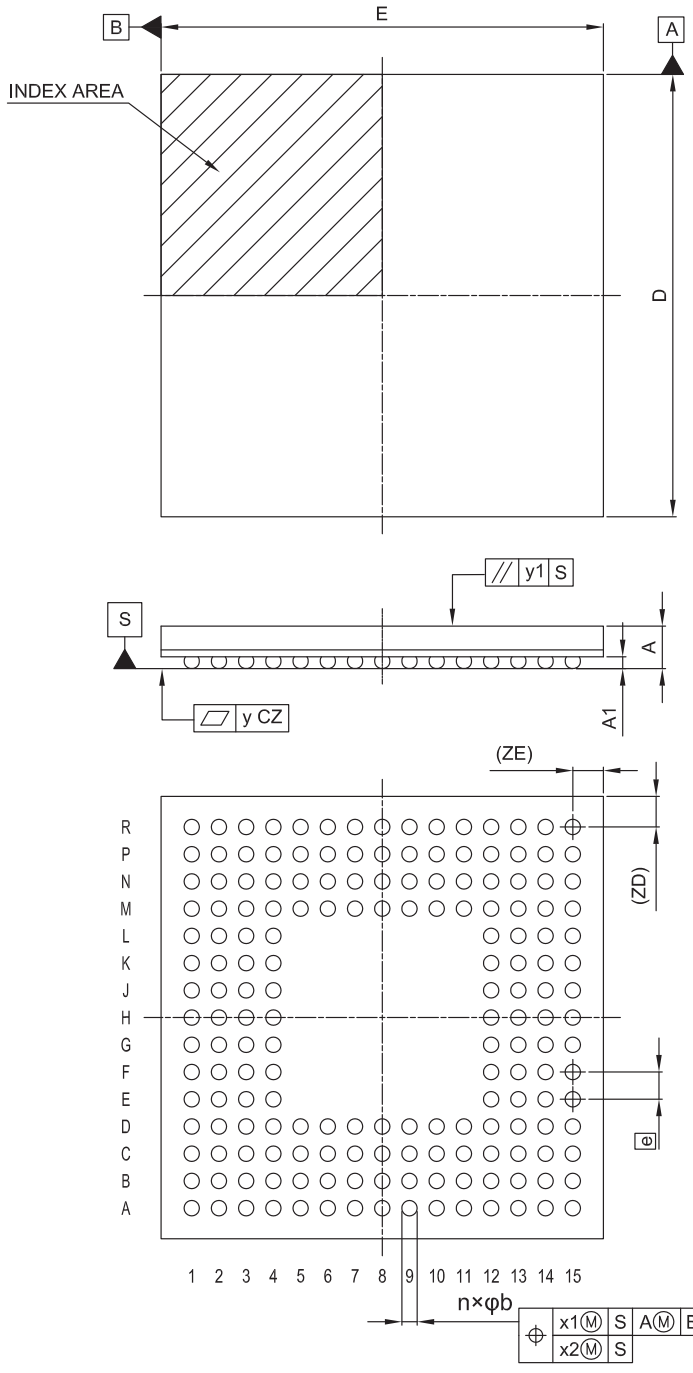
Pin	Handling
1.2-V power (VCC, PLLVcc, LVDSPLL VCC)	Supply power at 1.2 V.
1.8-V power (MIPIAVcc18)	Supply power at 1.8 V.
1.8-V/3.3-V switchable power supply (PVCC_SPI, PVCC_HO, PVCC_SD0, PVCC_SD1)	Supply power at 1.8 V / 3.3 V.
3.3V power (PVCC, AVCC, USBAPVCC1, USBAPVCC0, USBDPVCC1, USBDPVCC0, LVDSAPVCC)	Supply power at 3.3 V.
Ground (Vss, AVss, USBVss)	Connect to ground.
RREF0, RREF1	Connect this pin to USBVss via 2.2 kΩ ± 1% resistor.
DP1, DP0, DM1, DM0	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level) or open circuit.
EXTAL, RTC_X1, AUDIO_X1, USB_X1	Connect the pins to the crystal oscillator or the clock-input signal, or to a fixed level (pull them up or down, or connect them to the power supply or ground level)
XTAL, RTC_X2, AUDIO_X2, USB_X2	Connect the pins to the crystal oscillator or open circuit.
Dedicated input pins other than those listed above	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level).
Input/output pins (other than those listed above) in the input state	Fix the level on the pins (pull them up or down).
Input/output pins (other than those listed above) in the high-impedance state	Fix the level on the pins (pull them up or down) or open circuit.
Input/output pins (other than those listed above) in the output state	Open-circuit
Dedicated output pins other than those listed above	Open-circuit

Note: We recommend that the values of pull-up or pull-down resistors are in the range from 4.7 kΩ to 100 kΩ.

Appendix

A. Package Dimensions

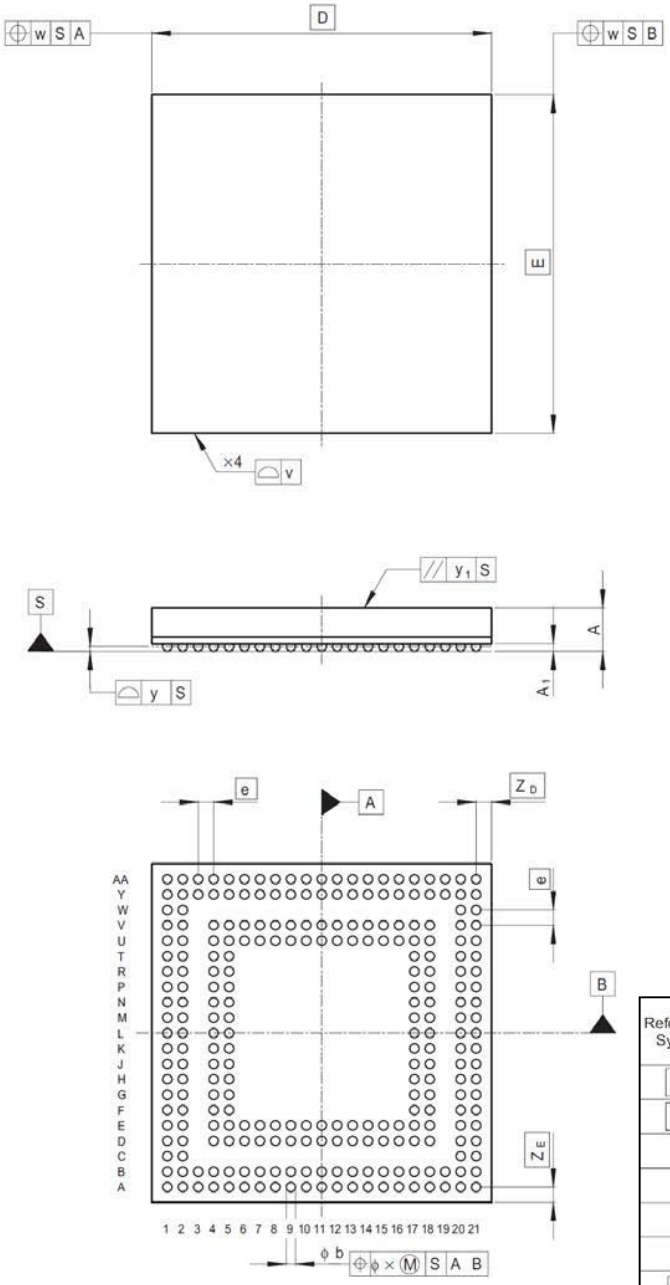
JEITA Package Code	RENESAS Code	MASS(TYP.)[g]
P-LFBGA176-13x13-0.80	PLBG0176GA-B	0.41



Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
D	12.9	13.0	13.1
E	12.9	13.0	13.1
A	—	—	1.40
A_1	0.30	0.35	0.40
e	—	0.8	—
b	0.40	0.45	0.50
x_1	—	—	0.15
x_2	—	—	0.08
y	—	—	0.10
y_1	—	—	0.20
n	—	176	—
Z_D	—	0.9	—
Z_E	—	0.9	—

Figure A.1 Dimensions of 176-Pin BGA Package

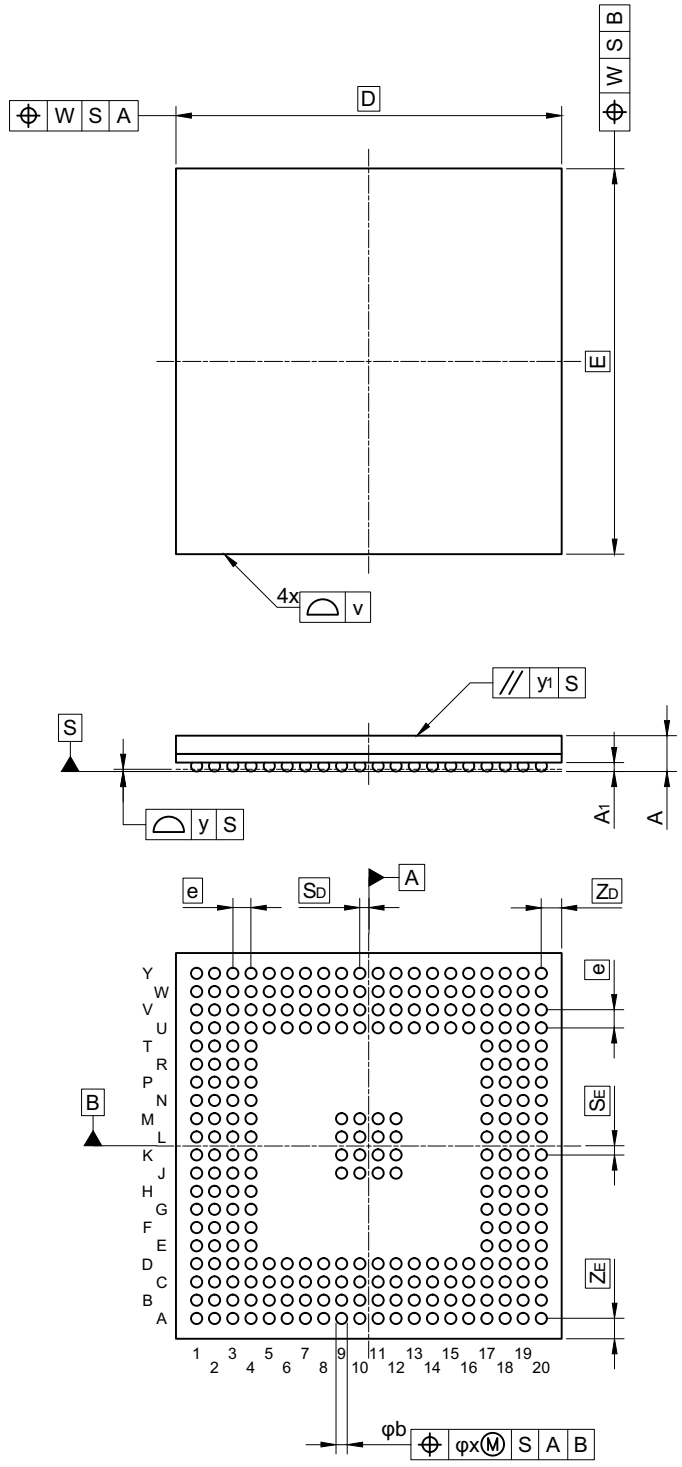
JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-LFBGA256-11x11-0.50	PLBG0256KA-A	BP-256C/BP-256CV	0.3g



Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	—	11.00	—
E	—	11.00	—
v	—	—	0.15
w	—	—	0.20
A	—	—	1.40
A_1	0.20	0.25	0.30
e	—	0.50	—
b	0.25	0.30	0.35
x	—	—	0.05
y	—	—	0.08
y_1	—	—	0.2
S_D	—	—	—
S_E	—	—	—
Z_D	—	0.5	—
Z_E	—	0.5	—

Figure A.2 Dimensions of 256-Pin BGA Package

JEITA Package code	RENESAS code	Previous code	MASS(TYP.)[g]
P-FBGA272-17x17-0.80	PRBG0272GA-A	-	0.90

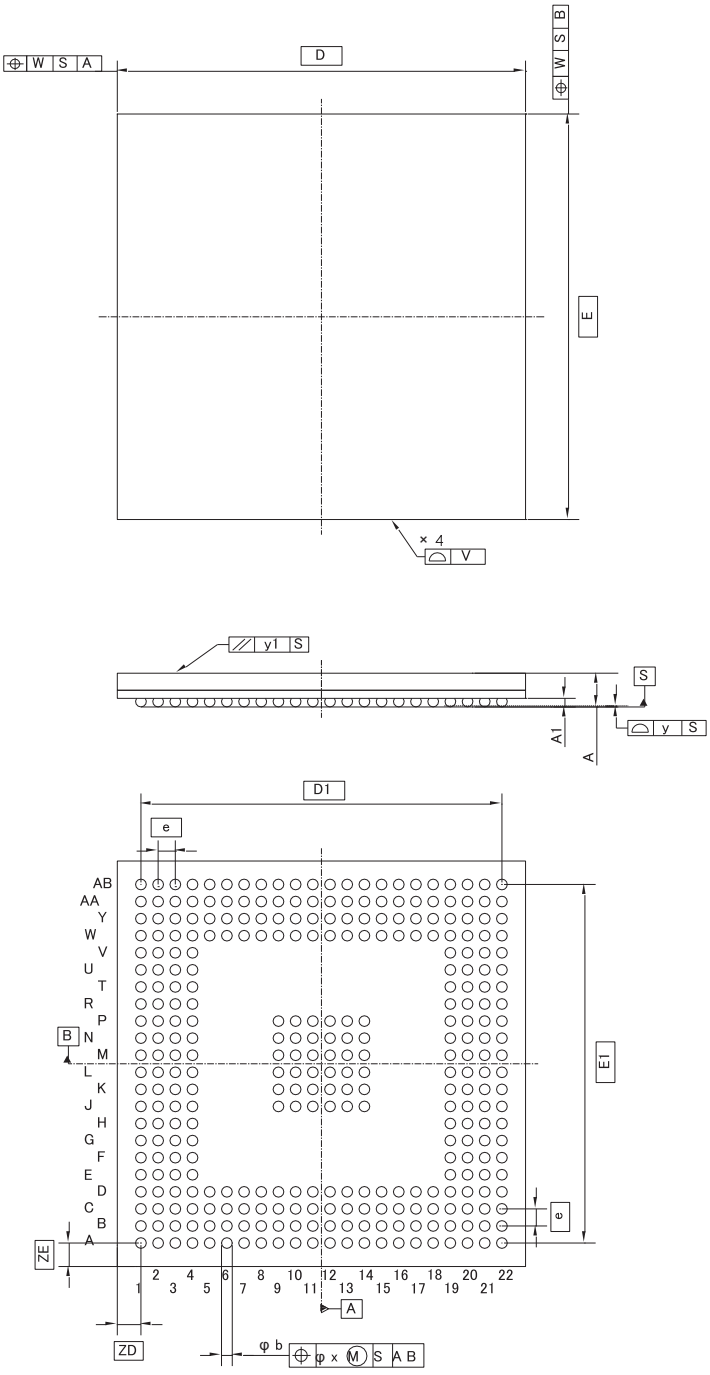


Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
D	–	17.00	–
E	–	17.00	–
v	–	–	0.15
w	–	–	0.30
A	–	–	1.90
A_1	0.35	0.40	0.45
e	–	0.80	–
b	0.45	0.50	0.55
x	–	–	0.08
y	–	–	0.10
y_1	–	–	0.20
S_D	–	0.40	–
S_E	–	0.40	–
Z_D	–	0.90	–
Z_E	–	0.90	–

Figure A.3 Dimensions of 272-Pin BGA Package

JEITA Package code	RENESAS code	Previous code	MASS(TYP.)[g]
P-FBGA324-19x19-0.80	PRBG0324GA-A	-	1.2

Unit:mm



Reference Symbol	Dimension in Millimeters		
	Min	Mon	Max
D	—	19.00	—
D1	—	16.80	—
E	—	19.00	—
E1	—	16.80	—
v	—	—	0.15
w	—	—	0.20
e	—	0.80	—
A	—	1.76	2.10
A1	0.35	0.40	0.45
b	0.45	0.50	0.55
x	—	—	0.08
y	—	—	0.10
y1	—	—	0.20
ZD	—	1.10	—
ZE	—	1.10	—

Figure A.4 Dimensions of 324-Pin BGA Package

B. Thermal Characteristics

The junction temperature (Tj) of an average sample (typ. value) can be calculated from either of the following expressions.

Calculation by using θja

$Tj = Ta + (P_D \times \theta ja)$, where

- Tj: junction temperature;
- Ta: ambient temperature;
- P_D: power consumption of the entire LSI; and
- θja: thermal resistance between Tj and Ta.

Calculation by using Ψjt

$Tj = Tt + (P_D \times \Psi jt)$, where

- Tt: temperature at the center of the top of the package; and
- Ψjt: thermal characteristics parameter representing the difference between Tj and Tt.

Since the value of θja varies greatly with the usage environment (e.g. the board and casing), the junction temperature (Tj) calculated by using θja is less accurate. Accordingly, when calculating Tj by using θja, allow a sufficient margin in designing the system. The value of Ψjt varies little with differences in the usage environment, but pay particular attention to measuring the temperature at the center of the top of the package (Tt) if you use this approach. For the values of θja and Ψjt, contact your local sales representative.

Revision History	RZ/A2M Group User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
1.00	Sep 25, 2018	—	First edition, issued
2.00	Dec 28, 2018	1. Overview	
		1-2	Table 1.1 Features of RZ/A2: Specification of Boot modes (Boot mode 4) corrected
		1-4	Table 1.1 Features of RZ/A2: Specification of SPI multi I/O bus controller corrected
		1-17	Table 1.3 Pin Function: Function of Multi-function timer pulse unit 3 (MTIOC8A, MTIOC8B, MTIOC8C, MTIOC8D) corrected
		1-18	Table 1.3 Pin Function: Function of SPI multi I/O bus controller (QSPI1_SSL) corrected
		1-45	Table 1.7 List of Pins (176-Pin BGA): I/O of Port Function/Dedicated Function (Ball Number: F4, Symbol: HM_CS1#/OM_CS1#) corrected
		3. Boot Mode	
		3-1	3. Boot Mode: Body corrected, Note 2 added
			3.1 Features: Boot mode 4 corrected, Note 4 added
			Table 3.1 External Pin (MD_BOOT2 to MD_BOOT0) Settings and Corresponding Boot Modes: Boot mode 4 corrected
		3-3	Table 3.2 Hardware Used in Each Boot Mode: Boot Mode 4 (Octal-SPI flash booting,1.8-V products) corrected
		3-4	Table 3.3 Exception Vector Address in Each Boot Mode: Boot Mode 4 (Octal-SPI flash booting, 1.8-V products) corrected
		3-16	3.5.5 Boot Mode 4: Body corrected
		3-24	3.7.3 Notes on Serial Flash Booting (Boot Mode 3, 4, 6) after This LSI is Reset: Body corrected
		9. Direct Memory Access Controller	
		9-30	9.4.9 Channel Configuration Register n/nS (CHCFG_n/nS): Description of bits 10 to 8 (AM[2:0]) corrected
		9-57	Table 9.4 On-Chip Module Requests: Setting of AM[2:0] for FIFO on-chip serial communication interface channel 0 to 4 corrected
			Table 9.4 On-Chip Module Requests: SEL[2:0] corrected
		9-77	Table 9.19 DACK0 Output Timing Setting: Purpose of Mask Mode corrected
		9-95	Table 9.29 Descriptor Setting: Descriptor 3 of NXLA (Next Link Address) corrected
		10. Multi-Function Timer Pulse Unit 3 (MTU3a)	
		10-91	Figure 10.7 Periodic Counter Operation: TGI interrupt signal name corrected
		12. General PWM Timer (GPT)	
		12-5	12.2 Register Descriptions: Body corrected
		16. Realtime Clock (RTC)	
		16-28	16.2.20 RTC Control Register 3 (RCR3): Note deleted
		17. Serial Communications Interface with FIFO (SCIFA)	
		17-1	Table 17.1 Specifications of SCIFA: Note 2 deleted
		17-8	17.2.6 Serial Control Register (SCR): Note 2 deleted
		17-11	17.2.7 Serial Status Register (FSR): RDF Bit (Receive FIFO Data Full Flag) [Clearing conditions], corrected
		17-12	17.2.7 Serial Status Register (FSR): TDFE Bit (Transmit FIFO Data Empty Flag) [Clearing conditions], corrected
		17-34	Figure 17.4 Sample Flowchart for Transmitting Serial Data in Asynchronous Mode: Flowchart corrected
		17-36	Figure 17.7 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (1): Flowchart corrected

Rev.	Date	Description	
		Page	Summary
2.00	Dec 28, 2018	17-42	Figure 17.13 Sample Flowchart for Transmitting Serial Data in Clock Synchronous Mode: Flowchart corrected
		17-44	Figure 17.15 Sample Flowchart for Receiving Serial Data in Clock Synchronous Mode: Flowchart corrected
		17-46	Figure 17.17 Sample Flowchart for Simultaneous Transmitting/Receiving Serial Data in Clock Synchronous Mode: Flowchart corrected
		17-52	17.8.1 FTDR Register Writing and TDFE Flag: Body corrected
			17.8.2 FRDR Register Reading and RDF Flag: Body corrected
		20. SPI Multi I/O Bus Controller	
		20-1	20. SPI Multi I/O Bus Controller: Body corrected
			20.1 Features: Body corrected
			20.1.1 Serial Flash Memory Interface: Body corrected, Note deleted
			20.1.2 OctaFlash™, Xccela™ Flash Memory Interface: Body corrected, Note added
		20-4	Table 20.1 Pin Configuration: Octal-SPI flash memory: Name of Octal-SPI flash memory connection corrected
		20-6, 20-7	20.4.1 Common Control Register (CMNCR): Description of bits 23, 22 (MOIIO3[1:0]), bits 21, 20 (MOIIO2[1:0]), bits 19, 18 (MOIIO1[1:0]), bits 17, 16 (MOIIO0[1:0]), and bits 1, 0 (BSZ[1:0]) corrected
		20-14, 20-15	20.4.7 Data Read Enable Setting Register (DRENR): Body corrected Description of bit 14 (CDE), bit 12 (OCDE), and bits 11 to 8 (ADE[3:0]) corrected
		20-19, 20-20	20.4.12 Manual Mode Enable Setting Register (SMENR): Body corrected Description of bit 14 (CDE), bit 12 (OCDE), and bits 11 to 8 (ADE[3:0]) corrected
		20-22	20.4.15 Manual Mode Write Data Register 0 (SMWDR0): Description of bits 31 to 0 (WDATA0[31:0]) corrected
		20-23	20.4.16 Manual Mode Write Data Register 1 (SMWDR1): Description of bits 31 to 0 (WDATA1[31:0]) corrected
		20-26	20.4.19 Data Read DDR Enable Register (DRDRENR): Description of bits14 to 12 (HYPE) corrected
		20-28	20.4.21 Manual Mode DDR Enable Register (SMDRENR): Description of bits 14 to 12 (HYPE[2:0]) corrected
		20-29, 20-30	20.4.22 PHY Control Register (PHYCNT): Description of bit 30 (ALT_ALIGN), bits 23, 22 (OCTA[1:0]), bit 21 (EXDS), bit 20 (OCT), bits 17, 16 (CKSEL[1:0]), and bits 1, 0 (PHYMEM[1:0]) corrected
		20-33	20.4.24 PHY Offset Register 2 (PHYOFFSET2): Description of bits 10 to 8 (OCTTMG) corrected
		20-35	20.4.27 PHY Adjustment Register 2 (PHYADJ2): Description of bits 31 to 0 (ADJ2[31:0]) corrected
		20-40	20.5.2 Address Map: Body corrected
			Table 20.3 Address Map: “Max. Access Area” corrected
		20-41	20.5.3 32-bit Serial Flash Addresses: Body corrected
		20-53	Table 20.7 Enable Register (Octal-SPI Flash Memory): Title corrected External Address Space Read Operation, corrected
		20-57	Table 20.9 Pin Status (2): Transfer Data (1-bit Size) corrected
			Table 20.11 Pin Status (4): Dummy Cycle corrected
		20-58	20.5.10 QSPIn_SSL Pin Control: Note corrected
		20-61	20.5.14 Data Alignment for Octal-SPI Flash Memory: Title corrected, Body corrected
			Figure 20.26 Octal-SPI Flash Memory Sequential Alignment (PHYCNT.OCTA = 10): Title corrected, Note corrected
			Figure 20.27 Octal-SPI Flash Memory Alternative Alignment (PHYCNT.OCTA = 01): Title corrected, Note corrected
		20-62	20.5.16 Supported Protocol for Octal-SPI Flash Memory: Title corrected, Body corrected
			Table 20.15 Supported Protocol for Octal-SPI Flash Memory: Title corrected

Rev.	Date	Description	
		Page	Summary
2.00	Dec 28, 2018	20-63 to 20-65	20.5.17 Timing Adjustment: Body corrected
		20-63	Figure 20.28 (1) Flow of Timing Adjustment (Serial Flash (SDR mode)): Title corrected, Flowchart corrected
		20-64	Figure 20.28 (2) Flow of Timing Adjustment (Serial Flash (DDR mode)): Newly added
		20-65	Figure 20.28 (3) Example of Timing Adjustment (Serial Flash (DDR mode)): Newly added
		20-66	Figure 20.29 Data Alignment in External Address Space Read Mode: "When a HyperFlash memory device is connected (PHYCNT.OCTA[1:0] = 00) or an Octal-SPI flash memory device is connected (PHYCNT.OCTA[1:0] = 01 and PHYCNT.ALT_ALIGN = 1)" Title corrected
		20-67	Figure 20.30 Data Alignment in Manual Mode: "When an Octal-SPI flash memory device is connected (PHYCNT.OCTA[1:0] = 01, PHYCNT.ALT_ALIGN = 0, and PHYCNT.PHYMEM[1:0] = 00)" Title corrected “When an Octal-SPI flash memory device is connected (PHYCNT.OCTA[1:0] = 01, PHYCNT.ALT_ALIGN = 1, and PHYCNT.PHYMEM[1:0] = 01)” Title corrected
		20-69	20.6.6 Write Data when the Octal-SPI Flash Memory is Connected: Title corrected, Body corrected
		21. HyperBus™ Controller	
		21-10	21.5.6 CS0 Memory Timing Register (MTR0): Description of bits 15 to 12 (RCSH[3:0]) and bits 11 to 8 (WCSH[3:0]) corrected
		21-11	21.5.7 CS1 Memory Timing Register (MTR1): Description of bits 15 to 12 (RCSH[3:0]) and bits 11 to 8 (WCSH[3:0]) corrected
		22. Octa Memory Controller	
		22-4	Table 22.3 Octa memory controller register configuration: Initial Value of Device size register 0 (DSR0), Device size register 1 (DSR1), and Memory delay trim register (MDTR) corrected
		22-7	22.4.3 Device Command Setting Register (DCSR): Description of bit 27 (DOPI) corrected
		22-8	22.4.4 Device Size Register 0 (DSR0): Initial Value of bits 29 to 0 (DV0SZ[29:0]) corrected
			22.4.5 Device Size Register 1 (DSR1): Initial Value of bits 31, 30 (DV1TYP[1:0]) and bits 29 to 0 (DV1SZ[29:0]) corrected
		22-9	22.4.6 Memory Delay Trim Register (MDTR): Initial Value of bits 27 to 24 (DQSEDOPI[3:0]), bits 15 to 12 (DQSESOPI[3:0]), and bits 11 to 8 (DQSERAM[3:0]) corrected
		22-14, 22-15	22.4.10 Device Memory Map Read Chip Select Timing Setting Register (DRCSTR): Description of bit 7 (CTR0) and bits 23 to 16 (Reserved) corrected
		23. I^2C Bus Interface	
		23-25, 23-26	23.3.7 RIICnSER — I^2C Bus Status Enable Register: Bit Name of SARy Bit (Slave Address Register y Enable) (y = 0 to 2) corrected
			Table 23.12 RIICnSER register contents: Bit Name of Bit Position 2, 1, and 0 (SAR2, SAR1, and SAR0) corrected
		23-30	23.3.9 RIICnSR1 — I^2C Bus Status Register 1 AASy Flag (Slave Address y Detection) (y = 0 to 2): Bit Name of RIICnSER.SARy (y = 0 to 2) corrected
		23-37	23.3.11 RIICnSARy — I^2C Slave Address Register y (y = 0 to 2) SVA0 Bit (10-Bit Address LSB): Bit Name of RIICnSER.SARy (y = 0 to 2) corrected
			23.3.11 RIICnSARy — I^2C Slave Address Register y (y = 0 to 2) SVA[9:1] Bits (7-Bit Address/10-Bit Address Upper Bits): Bit Name of RIICnSER.SARy (y = 0 to 2) corrected
			23.3.11 RIICnSARy — I^2C Slave Address Register y (y = 0 to 2) FSy Bit (7-Bit/10-Bit Address Format Selection): Bit Name of RIICnSER.SARy (y = 0 to 2) corrected
		23-67	23.9.1 Slave-Address Match Detection: Bit Name of RIICnSER.SARy bit (y = 0 to 2) corrected
		30. 12-Bit A/D Converter	
		30-15	30.2.4 A/D Channel Select Register A0 (ADANSA0): Bits b15 to b8 (Reserved) added
			30.2.5 A/D Channel Select Register B0 (ADANSB0): Bits b15 to b8 (Reserved) added
		30-16	30.2.6 A/D Channel Select Register C0 (ADANSC0): Bits b15 to b8 (Reserved) added

Rev.	Date	Description	
		Page	Summary
2.00	Dec 28, 2018	30-17	A/D-Converted Value Addition/Average Mode Select Register 0 (ADADS0): Bits b15 to b8 (Reserved) added
		32. USB 2.0 Host Module	
		32-53	Table 32.43 SPD_CTRL Register: Description of bit 31 (SUSPENDM_ENABLE), bit 30 (SLEEPM_ENABLE), and bit 23 (WKCNNT_ENABLE) corrected
		33. USB 2.0 Function Module	
		33-3	33.1.2.7 Other functions: Body corrected
		33-7	Table 33.3 List of Registers: D0FIFOSEL (D0FIFO Port Select Register), D0FIFOCTR (D0FIFO Port Control Register), D1FIFOSEL (D1FIFO Port Select Register), and D1FIFOCTR (D1FIFO Port Control Register) Newly added
		33-20	33.2.7.3 CFIFO Port Select Register [CFIFOSEL] <Address: 020H>: Function of bit 14 (REW) corrected
		33-22, 33-23	33.2.7.4 D0FIFO Port Select Register [D0FIFOSEL] <Address: 028H> D1FIFO Port Select Register [D1FIFOSEL] <Address: 02CH>: Newly added
		33-24	33.2.7.5 CFIFO Port Control Register [CFIFOCTR] <Address: 022H> D0FIFO Port Control Register [D0FIFOCTR] <Address: 02AH> D1FIFO Port Control Register [D1FIFOCTR] <Address: 02EH>: Title corrected
		33-55	33.2.15.4 Pipe Maximum Packet Size Register [PIPEMAXP] <Address: 06CH> (1) Maximum packet size (MXPS) bits: Body corrected
		—	Device Address Configuration Registers: deleted
		33-90, 33-91	33.5.3.1 Channel Configuration Register ch0 [CHCFG_0] <Address: 42CH> Channel Configuration Register ch1 [CHCFG_1] <Address: 46CH>: Function of bit 0 (SEL) added
		33-91	Table 33.21 Range of Sizes That Can Be Specified in SDS and DDS Bits, corrected
		33-115	33.9.4.1 FIFO buffer allocation: Body corrected
			Figure 33.9 Example of FIFO Buffer Memory Mapping, corrected
		33-116	Table 33.25 List of FIFO Buffer Clearing Modes, corrected
		33-117	33.9.5 FIFO Port Functions: Body corrected
			Table 33.26 FIFO Port Function Settings, corrected
		33-118	33.9.5.1 FIFO port selection: Body corrected
			Table 33.27 FIFO Port Access for Each Pipe, corrected
		33-119	33.9.5.2 DxFIFO automatic clear mode (DxFIFO port read direction): Newly added
			Table 33.28 Relationship between Packet Reception and Buffer Memory Clear Processing by the Software: Newly added
			33.9.5.3 BRDY interrupt timing selection function: Body corrected
		33-135	33.9.12.1 Register mode/link mode (1) Register mode (a) Operation flow of register mode Description of the register mode operation flow 1. Channel setting: Body corrected
		33-143	33.9.12.1 Register mode/link mode (2) Link mode (c) Descriptor setting: Body corrected
		33-168	Figure 33.35 Flowchart of Resume from Deep Standby in Response to a Bus Reset, corrected
		33-169	Figure 33.36 Flowchart of Resumption from Deep Standby in Response to Reception of the Resume Signal, corrected
		43. 2D Drawing Engine (DRW)	
		43-63	43.6.5.1 Color Channel Blending: Blending formula corrected
		43-65	43.6.5.2 Alpha Channel Blending: Blending formula corrected
		47. MIPI CSI2 Interface	
		47-39	47.2.37 SOT Error Count Register (SERRCNT): Bit Name of bits 7 to 0 (ERRSOTHS_CNT) corrected

Rev.	Date	Description	
		Page	Summary
2.00	Dec 28, 2018	47-40	47.2.39 ECC_CRCT Count Register (ECCCM): Bit Name of bits 3 to 0 (ECC_CRCT_CNT) corrected
		47-41	47.2.40 ECC_ERR Count Register (ECECM): Bit Name of bits 7 to 0 (ECC_ERR_CNT) corrected
			47.2.41 CRC_ERR Count Register (CRCECM): Bit Name of bits 7 to 0 (CRC_ERR_CNT) corrected
		47-48	47.2.53 PHY Timing Register 2 (PHYTIM2): Bit 5 (Reserved) corrected
		48. Video Input Module	
		48-14	48.2.6 Video n Start Pixel Pre-Clip Register (VnSPPrC): Description of bits 10 to 0 (SPPrC[10:0]) corrected
		48-15	48.2.7 Video n End Pixel Pre-Clip Register (VnEPPrC): Description of bits 10 to 0 (EPPrC[10:0]) corrected
		49. SD/MMC Host Interface	
		49-22	49.2.15 SD Buffer Read/Write Register (SD_BUF0): Bit Name of bits 63 to 0 (BUF63 to BUF0) corrected
		49-30	49.2.27 DMAC Reset Register (DM_CM_RST): Note added
		49-61	Figure 49.24 SD_BUF DMA Read Flowchart Example: Flowchart corrected
		49-62	Figure 49.25 SD_BUF DMA Write Flowchart Example: Flowchart corrected
		51. GPIO	
		51-26	Table 51.10 Register Settings for Input/Output Pin Function: Pin name (SSILRCK1) corrected
		51-32	Table 51.16 Register Settings for Input/Output Pin Function in 324-Pin Products: Pin name (SSILRCK3) corrected
		51-34	Table 51.18 Register Settings for Input/Output Pin Function in 324-Pin Products: Pin name (SSILRCK2 and SSILRCK0) corrected
		51-36	Table 51.20 Register Settings for Input/Output Pin Function in 324-/272-/256-Pin Products: Pin name (SSILRCK1 and SSILRCK2) corrected
		51-39	Table 51.23 Register Settings for Input/Output Pin Function: Pin name (SSILRCK0) corrected
		51-40	Table 51.24 Register Settings for Input/Output Pin Function: Pin name (SSILRCK0) corrected
		51-44	Table 51.28 Register Settings for Input/Output Pin Function in 324-/272-/256-Pin Products: Pin name (SSILRCK3 and SSILRCK2) corrected
		51-45	Table 51.29 Register Settings for Input/Output Pin Function in 176-Pin Products: Pin name (SSILRCK3 and SSILRCK2) corrected
		51-62	Table 51.34 Register Settings: "Point to Note" corrected
		51-67	Table 51.38 Conditions for Input of Interrupt Requests between Port Pins in the Same Group: Note 2 corrected
		52. Power-Down Modes	
		52-40	52.2.33 USB Deep standby cancel source flag register (USBDSFR): Bit Name (USBDSF3, USBDSF2, USBDSF1, and USBDSF0) corrected
		52-51	Table 52.5 External Memory Control Pins in Different Modes: Boot Mode 4 (Octal-SPI Flash Boot) corrected
		56. Electrical Characteristics	
		56-4	Table 56.2 DC Characteristics (3) [Except 1.8-/3.3-V Switchable I/O Interface, I²C Bus Interface, and USB 2.0 Host/Function Module-Related Pins]: Schmitt trigger input characteristics, corrected
		56-53, 56-54	Table 56.18 SPI Multi I/O Bus Controller Timing*1, corrected
		56-54	Figure 56.56 SDR Transfer Format Transmission and Reception Timing, corrected
			Figure 56.57 DDR Transfer Format Transmission and Reception Timing, corrected
		56-55	Figure 56.60 Transmit/Receive Timing with Octal-SPI flash memory or HyperFlash™ Connected: Title corrected
		56-56	Table 56.19 HyperBus™ Controller Timing*1*2, corrected
		56-58	Table 56.20 Octa Memory Controller Timing*1*2, corrected
			Figure 56.66 Clock Timing: Title corrected, Signal name (OM_SCLK) corrected

Rev.	Date	Description	
		Page	Summary
2.00	Dec 28, 2018	56-59	Figure 56.67 SPI Transfer Format Transmission and Reception Timing, corrected
			Figure 56.68 SOPI Transfer Format Transmission and Reception Timing, corrected
		56-60	Figure 56.70 DOPI Transfer Format Transmission Timing: Newly added
		56-83	Table 56.36 SD/MMC Host Interface Timing (MMC HS-DDR mode 3.3-/1.8-V power supply selection), corrected
		56-84	Table 56.37 SD/MMC Host Interface Timing (MMC HS200 mode 1.8-V power supply selection, Output load: 15pF), corrected
3.00	Dec 06, 2019	All	12-Bit A/D Converter corrected to A/D Converter
		1. Overview	
		1-4	Table 1.1 Features of RZ/A2M: Specification of Octa memory controller corrected
		1-5	Table 1.1 Features of RZ/A2M: Item and specification of CANFD interface corrected
		1-5	Table 1.1 Features of RZ/A2M: Specification of USB 2.0 host/function module corrected
		1-10	Table 1.2 Product Lineup: Note corrected
		1-18	Table 1.3 Pin Function: Classification of CANFD interface corrected
		1-20	Table 1.3 Pin Function: Function of the RREF1 and RREF0 pins of the USB 2.0 host/function module corrected
		3. Boot Mode	
		3-2	Table 3.1 External Pin (MD_BOOT2 to MD_BOOT0) Settings and Corresponding Boot Modes: Boot mode name of Boot Mode 6 corrected
		3-3	Table 3.2 Hardware Used in Each Boot Mode: QSPI0_SCLK and QSPI1_SCLK respectively corrected to QSPI0_SPCLK and QSPI1_SPCLK, Boot mode name of Boot Mode 6 corrected
		3-4	Table 3.3 Exception Vector Address in Each Boot Mode: Boot mode name of Boot Mode 6 corrected
		3-13	Figure 3.8 Example of connection with eMMC device, corrected
		3-14	Figure 3.9 Control signals output to the serial flash memory through SPI communication conversion, corrected
		3-15	Figure 3.10 Connection example when using boot mode 3, corrected
		3-16	Figure 3.11 Connection example when using boot mode 4, corrected
		3-17	Figure 3.12 Control signal output to HyperFlash by HyperBus protocol conversion, corrected
		3-18	Figure 3.13 Connection example when using boot mode 5, corrected
		3-19	Figure 3.14 Control signals output to the OctaFlash, corrected
		3-19	Table 3.10 Register setting value of each peripheral module in boot mode 6: Initial values after reset of the DSR0 register corrected
		3-20	Figure 3.15 Connection example when using boot mode 6, corrected
		3-21	3.5.8 Boot Mode 7: Body corrected
		3-21	Figure 3.16 Control signal output to HyperFlash by HyperBus protocol conversion, corrected
		3-22	Figure 3.17 Connection example when using boot mode 7, corrected
		3-24	3.7.3 Notes on Serial Flash Booting (Boot Mode 3, 4, 6) after This LSI is Reset: Body corrected
		5. LSI Internal Bus	
		5-3	Table 5.2 List of Peripheral Buses: Description of Connected peripheral modules corrected
		5-13	5.8.5 Slave Area (3) Protection unit information (ARPROT[2:0], AWPROT[2:0]): Note corrected
		5-49	5.11.25 Slave Access Control Register 0 (SLVACCCTL0): Description of bit 14 (POEGNS) corrected
		5-51	5.11.26 Slave Access Control Register 1 (SLVACCCTL1): Description of bit 16 (RCANNS) corrected
		6. Clock Pulse Generator	
		6-2	Figure 6.1 Block Diagram: Peripheral clock 0C deleted
		6-18	Figure 6.8 Clock Signals for Other Modules 1: Description of frequency deleted
		6-20	Figure 6.10 Distribution of Internal Clock Signals: DRP module added
		6-21	Figure 6.11 Distribution of Internal Clock Signals (2): DRP module deleted

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	7. Interrupt Controller	
		7-17	7.4.4 Pin Interrupts: Body corrected
		7-26	Table 7.3 List of Interrupt IDs: Request Source Name(ALM, ALM_S) of Realtime Clock Module corrected
		8. Bus State Controller	
		8-8, 8-9	8.4.2 CSn Space Bus Control Register (CSnBCR) (n = 0 to 5): Name of bits 24 to 22, 21 to 19, and 18 to 16 (IWRRD[2:0], IWRRS[2:0], IWRWS[2:0]) corrected, Description of bits 24 to 22 (IWRRD[2:0]), bits 21 to 19 (IWRRS[2:0]), bits 18 to 16 (IWRWS[2:0]), and bit 14 to 12 (TYPE[2:0]) corrected
		8-76	8.5.11 Others (4) Usage Note on Changing CKIO Clock Selections: Register name (CKIO Select Register) corrected
		9. Direct Memory Access Controller	
		9-53 to 9-56, 9-58	Table 9.4 On-Chip Module Requests: General PWM timer channel 0 to 7, CANFD interface, corrected
		10. Multi-Function Timer Pulse Unit 3 (MTU3a)	
		10-1	Table 10.1 MTU Specifications: Description of Count clock, Available operations, and Low power consumption function corrected
		10-2, 10-3	Table 10.2 MTU Functions: External clocks in phase-counting mode added, MTU Function of DMAC activation, Interrupt sources and Module stop function corrected, Item (Interrupt skipping 1) corrected, Note 3 added, and note number after Note 3 renumbered.
		10-4	Figure 10.1 Block Diagram of MTU (MTU0 to MTU4, MTU8), corrected
		10-5	Figure 10.2 Block Diagram of MTU (MTU5 to MTU7), corrected
		10-6	Table 10.3 Pin Configuration of the MTU: Function of MTCLKA, and MTCLKB corrected
		10-12	10.2.1 Timer Control Register (TCR): Register symbols (MTU0.TCR, MTU1.TCR, MTU2.TCR, MTU3.TCR, MTU4.TCR, MTU6.TCR, MTU7.TCR, and MTU8.TCR) corrected, Description of TPSC[2:0]bit, and CKEG[1:0] bit corrected
		10-13	Table 10.5 CCLR[2:0] (MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8): Location moved
		10-13	Table 10.6 CCLR[2:0] (MTU1 and MTU2): Location moved
		10-13	10.2.1 Timer Control Register (TCR): Register symbol (MTU5.TCRU, MTU5.TCRV, and MTU5.TCRW) corrected, Description of TPSC[1:0] bit corrected
		10-14	10.2.2 Timer Control Register 2 (TCR2): Register symbol (MTU0.TCR2, MTU3.TCR2, MTU4.TCR2, MTU6.TCR2, MTU7.TCR2, and MTU8.TCR2) corrected
		10-14	10.2.2 Timer Control Register 2 (TCR2): Register symbol (MTU1.TCR2, and MTU2.TCR2) corrected, Body corrected, Description of TPSC2[2:0] bit corrected
		10-15	10.2.2 Timer Control Register 2 (TCR2): Register symbols (MTU5.TCR2U, MTU5.TCR2V, MTU5.TCR2W) corrected, Description of Bit table (x: Don't care) added, Description of TPSC2[2:0] bit, and CKEG[1:0] bit, corrected
		10-17	Table 10.10 TPSC[2:0], TPSC2[2:0] (MTU3, MTU4, MTU6, MTU7, and MTU8): Note 1 deleted
		10-17	Table 10.11 TPSC[1:0], TPSC2[2:0] (MTU5): Note deleted
		10-20	10.2.4 Timer Mode Registers 2 (TMDR2A and TMDR2B): Body corrected
		10-21	10.2.5 Timer Mode Register 3 (TMDR3): Description of LWA Bit corrected
		10-23	10.2.6 Timer I/O Control Register (TIOR), MTU5.TIORU, MTU5.TIORV, MTU5.TIORW: Body corrected
		10-24	Table 10.14 TIORH (MTU0): Description of IOB[3:0]=11xx corrected, Note 1 added.
		10-25	Table 10.15 TIORL (MTU0): Description of IOD[3:0]=11xx corrected. Note 2 added.
		10-26	Table 10.16 TIOR (MTU1): Table heading, corrected (MTU1.TGRB/TGRBLW Function)

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	10-29	Table 10.23 TIORL (MTU6): Note 1, corrected
		10-30	Table 10.25 TIORL (MTU7): Note 1, corrected
		10-31	Table 10.26 TIORH (MTU8): Value of Bit 6 (IOB2) corrected. Note 1 added.
		10-31	Table 10.27 TIORL (MTU8): Note 1, corrected
		10-32	Table 10.28 TIORH (MTU0): Description of IOA[3:0] = 1100 corrected. Note 1 added.
		10-32	Table 10.29 TIORL (MTU0): Description of IOC[3:0] = 11xx corrected. Note 2 added.
		10-33	Table 10.30 TIOR (MTU1): Table heading, corrected (MTU1.TGRA/TGRALW Function)
		10-35	Table 10.35 TIORL (MTU4): Note 1, corrected
		10-36	Table 10.37 TIORL (MTU6): Note 1, corrected
		10-37	Table 10.39 TIORL (MTU7): Note 1, corrected
		10-39	Table 10.42 TIORU, TIORV, and TIORW (MTU5): Note 1 added
		10-41	10.2.8 Timer Interrupt Enable Register (TIER): Register symbols (MTU1.TIER, MTU2.TIER, MTU0.TIER, MTU3.TIER, MTU6.TIER, MTU4.TIER, MTU7.TIER, and MTU8.TIER) corrected
		10-43	10.2.8 Timer Interrupt Enable Register (TIER): Register symbols (MTU0.TIER2) corrected, Description of TTGE2 Bit corrected, Register symbol (MTU5.TIER) corrected
		10-44	10.2.9 Timer Status Register (TSR): Register symbols (MTU1.TSR, MTU2.TSR, MTU3.TSR, MTU4.TSR, MTU6.TSR, and MTU7.TSR) corrected
		10-48	10.2.13 Timer Counter (TCNT), MTU8.TCNT: Note of Bit table added, Body corrected
		10-49	10.2.14 Timer Longword Counter (TCNTLW): Note of Bit table added
		10-50	10.2.15 Timer General Register (TGR): Register symbols (MTU0.TGR to MTU7.TGR) added.
		10-50	10.2.15 Timer General Register (TGR), MTU8.TGR: Note of Bit table added
		10-51	10.2.16 Timer Longword General Registers (TGRALW and TGRBLW): Title corrected, Note of Bit table added, Body corrected
		10-52	10.2.17 Timer Start Registers (TSTRA, TSTRB, and TSTR): Title corrected, MTU.TSTRA: Register symbol (MTU.TSTRA) corrected, Body corrected, Description of CSTn Bits corrected
		10-53	10.2.17 Timer Start Registers (TSTRA, TSTRB, and TSTR): Register symbol (MTU.TSTRB) corrected, Description of CSTn Bits corrected
		10-53	10.2.17 Timer Start Registers (TSTRA, TSTRB, and TSTR): Register symbol (MTU5.TSTR) corrected
		10-54	10.2.18 Timer Synchronous Registers (TSYRA and TSYRB): Title corrected, Register symbols (MTU.TSYRA (MTU0, MTU1, MTU2, MTU3, MTU4)) corrected
		10-55	10.2.18 Timer Synchronous Registers (TSYRA and TSYRB): Register symbol (MTU.TSYRB) corrected
		10-56, 10-57	10.2.19 Timer Counter Synchronous Start Register (TCSYSTR): Note 1. of Bit table corrected, Descriptions of SCH7 Bit, SCH6 Bit, SCH4 Bit, SCH3 Bit, SCH2 Bit, SCH1 Bit, and SCH0 Bit corrected
		10-58	10.2.20 Timer Read/Write Enable Registers (TRWERA and TRWERB): Description of the RWE bit corrected
		10-59	10.2.21 Timer Output Master Enable Registers (TOERA and TOERB): Title corrected, Register symbol (MTU.TOERA) corrected, Body corrected
		10-60	10.2.21 Timer Output Master Enable Registers (TOERA and TOERB): Register symbol (MTU.TOERB) corrected, Body corrected
		10-61	10.2.22 Timer Output Control Registers 1 (TOCR1A and TOCR1B): Note 4 of bit map deleted, Note 4 of bit 3 (TOCL) added, Description of OLSP Bit and OLSN Bit corrected
		10-62	Figure 10.3 Example of Output in Complementary PWM Mode, corrected
		10-63	10.2.23 Timer Output Control Registers 2 (TOCR2A and TOCR2B): Body corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	10-67	10.2.25 Timer Gate Control Register A (TGCRA): Description of bit 6 (BDC) corrected, Description of FB Bit and P Bit corrected
		10-71	Table 10.53 Setting of BTE[1:0] Bits in TBTERA and TBTERB: Title corrected
		10-72	10.2.32 Timer Waveform Control Registers (TWCRA and TWCRB): Description of SCC Bit corrected
		10-74	10.2.33 Noise Filter Control Register n (NFCRn) (n = 0 to 4, 6, 7, 8, C): Register symbols (MTU0.NFCR0, MTU1.NFCR1, MTU2.NFCR2, MTU3.NFCR3, MTU4.NFCR4, MTU6.NFCR6, MTU7.NFCR7, and MTU8.NFCR8) corrected, Body corrected
		10-75	10.2.33 Noise Filter Control Register n (NFCRn) (n = 0 to 4, 6, 7, 8, C): Register symbols (MTU0.NFCRC) corrected, Bit Symbol (NFCS[1:0]) corrected, Body corrected
		10-76	10.2.34 Noise Filter Control Register 5 (NFCR5): Pin names (MTIC5U, MTIC5V, and MTIC5W) corrected
		10-77	10.2.35 Timer A/D Converter Start Request Control Register (TADCR): Register symbol (MTU4.TADCR) corrected, Note 4 of the bit map deleted, Notes 1 to 3 of the bit table corrected, Description of bit 3 (ITA3AE), bit 2 (ITA4VE), bit 1 (ITB3AE), and bit 0 (ITB4VE) corrected, Name of bits 15 and 14 (BF[1:0]) corrected, Note numbers corrected, Note numbers corrected, Note, Note 1, and Note 3 corrected
		10-78	Table 10.54 Setting of Transfer Timing by TADCR.BF[1:0] Bits (MTU4), corrected
		10-79	10.2.35 Timer A/D Converter Start Request Control Register (TADCR): Register symbol (MTU7.TADCR) corrected, Note 4 of tje bit map deleted, Note 1 to 3 of the bit table corrected, Description of bit 3 (ITA6AE), bit 2 (ITA7VE), bit 1 (ITB6AE), and bit 0 (ITB7VE) corrected, Name of bits 15 and 14 (BF[1:0]) corrected, Note numbers corrected, Note, Note 1, and Note 3 corrected
		10-80	Table 10.55 Setting of Transfer Timing by TADCR.BF[1:0] Bits (MTU7), corrected
		10-81	10.2.36 Timer A/D Converter Start Request Cycle Set Registers (TADCORA and TADCORB): Note2 corrected
		10-82	10.2.38 Timer Interrupt Skipping Mode Registers (TITMRA and TITMRB): Description of bit 0 (TITM) corrected, Note 1, and Note2 added
		—	Table 10.56 Interrupt Skipping Function Selected through TITM Bit: deleted
		10-83, 10-84	10.2.39 Timer Interrupt Skipping Set Registers 1 (TITCR1A and TITCR1B): Register symbol (MTU.TITCR1A, and MTU.TITCR1B) corrected, the location of the description moved and corrected.
		10-85	10.2.40 Timer Interrupt Skipping Counters 1 (TITCNT1A and TITCNT1B): Register symbol (MTU.TITCNT1A) corrected, Description of bits 7 and 3 corrected
		10-86	10.2.40 Timer Interrupt Skipping Counters 1 (TITCNT1A and TITCNT1B): Register symbol (MTU.TITCNT1B) corrected, Description of bits 7 and 3 corrected.
		10-87	10.2.41 Timer Interrupt Skipping Set Registers 2 (TITCR2A and TITCR2B): Register symbol (MTU.TITCR2A) corrected
		10-88	10.2.41 Timer Interrupt Skipping Set Registers 2 (TITCR2A and TITCR2B): Register symbol (MTU.TITCR2B) corrected
		10-89	10.2.42 Timer Interrupt Skipping Counters 2 (TITCNT2A and TITCNT2B): Register symbol (MTU.TITCNT2A) corrected, Description of bits 7 to 3 corrected
		10-90	10.2.42 Timer Interrupt Skipping Counters 2 (TITCNT2A and TITCNT2B): Register symbol (MTU.TITCNT2B) corrected, Description of bits 7 to 3 corrected
		10-91	10.3.1 Basic Functions: Body corrected
		10-91	10.3.1 Basic Functions (1) Counter Operation: Body corrected
		10-91	Figure 10.5 Example of Count Operation Setting Procedure: Title corrected, Descrptions of [1] and [5], corrected
		10-92	10.3.1 Basic Functions (b) Free-Running Count Operation and Periodic Count Operation: Body corrected
		10-92	Figure 10.6 Free-Running Counter Operation, corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	10-92	Figure 10.7 Periodic Counter Operation, corrected
		10-93	10.3.1 Basic Functions (2) Waveform Output by Compare Match: Body corrected
		10-93	Figure 10.8 Example of Procedure for Setting Waveform Output by Compare Match: Description of [4], corrected
		10-95	10.3.1 Basic Functions (3) Input Capture Function: Body corrected, Note corrected
		10-95	Figure 10.11 Example of Input Capture Operation Setting Procedure: Description of [2], corrected
		10-96	Figure 10.12 Example of Input Capture Operation (n = 0 to 4, 6, 7, 8), corrected
		10-97	10.3.2 Synchronous Operation: Body corrected
		10-97	Figure 10.13 Example of Synchronous Operation Setting Procedure: Descriptions of [1] and [5], corrected
		10-100	Figure 10.17 Example of Buffer Operation Setting Procedure: Description of [3], corrected
		10-102	10.3.3 Buffer Operation (3) Selecting Timing for Transfer from Buffer Registers to Timer General Registers in Buffer Operation: Body corrected
		10-103	10.3.4 Cascaded Operation: Body corrected
		10-104	Figure 10.21 Cascaded Operation Setting Procedure: Descriptions of [1] and [2], corrected
		10-104	Figure 10.22 Cascaded Operation Example (a), corrected
		10-108	10.3.5 PWM Modes: Body corrected
		10-109	10.3.5 PWM Modes (b) PWM Mode 2: Body corrected
		10-110	Figure 10.26 Example of PWM Mode Setting Procedure: Descriptions of [2] to [4] and [7] corrected
		10-112	10.3.5 PWM Modes (2) Examples of PWM Mode Operation: Body corrected
		10-112	Figure 10.29 Examples of PWM Mode Operation (PWM Waveform Output with 0% Duty and 100% Duty) (n = 0 to 4, 6, 7), corrected
		10-114	10.3.6.1 16-Bit Phase Counting Mode: Body corrected
		10-114	Figure 10.30 Example of 16-Bit Phase Counting Mode Setting Procedure: Processing No. corrected ([2] → [3]), Description of [3] corrected
		10-116	Figure 10.32 Example of Operation in Phase Counting Mode 2 (When MTUn.TCR2.PCB[1:0] is 00b (n = 1, 2)): Title corrected
		10-116	Figure 10.33 Example of Operation in Phase Counting Mode 2 (When MTUn.TCR2.PCB[1:0] is 01b (n = 1, 2)): Title corrected
		10-116	Figure 10.34 Example of Operation in Phase Counting Mode 2 (When MTUn.TCR2.PCB[1:0] is 1Xb (n = 1, 2)): Title corrected
		10-118	Figure 10.35 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] is 00b (n = 1, 2)): Title corrected, Fiugre corrected
		10-118	Figure 10.36 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] is 01b (n = 1, 2)): Title corrected, Fiugre corrected
		10-118	Figure 10.37 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] is 1Xb (n = 1, 2)): Title corrected
		10-121	Figure 10.39 Example of Operation in Phase Counting Mode 5 (When MTUn.TCR2.PCB[1:0] = 0xb (n = 1, 2)): Title corrected
		10-121	Figure 10.40 Example of Operation in Phase Counting Mode 5 (When MTUn.TCR2.PCB[1:0] = 1xb (n = 1, 2)): Title corrected
		10-123	10.3.6.1 16-Bit Phase Counting Mode (3) 16-Bit Phase Counting Mode Application Example: Body corrected
		10-124	10.3.6.2 Cascade Connection 32-Bit Phase Counting Mode: Body corrected
		10-125	Figure 10.42 Block Diagram for Operation in Cascade Connection 32-Bit Phase Counting Mode, corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	10-126	Figure 10.43 Procedure for Setting Cascade Connection 32-Bit Phase Counting Mode: Processing corrected, Descriptions of [3] to [9] corrected
		10-128	10.3.7 Reset-Synchronized PWM Mode (1) Example of Procedure for Setting Reset-Synchronized PWM Mode: Body corrected
		10-128	Figure 10.44 Procedure for Selecting Reset-Synchronized PWM Mode: Descriptions of [2] and [6] corrected
		10-132	Table 10.76 Register Settings for Complementary PWM Mode (2/2): Descriptions under "Read/Write from CPU" for TEMP1A to TEMP6B corrected
		10-135	Figure 10.48 Example of Complementary PWM Mode Setting Procedure: Descriptions of [2] and [6] corrected
		10-136	10.3.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation: Destination for reference in Body corrected
		10-136	10.3.8 Complementary PWM Mode (a) Counter Operation: Body corrected
		10-136	Figure 10.49 Count Operation in Complementary PWM Mode (MTU3 and MTU4): Title corrected
		10-138	Figure 10.50 Example of Operation in Complementary PWM Mode (MTU3 and MTU4), corrected
		10-139	Table 10.77 Registers and Counters Requiring Initial Setting: Settings of the MTU3.TGRC, MTU6.TGRC, TDDRA, and TDDRB registers corrected
		10-141	Figure 10.51 Example of Operation without Dead Time (MTU3 and MTU4), corrected
		10-143	10.3.8 Complementary PWM Mode (h) Register Data Updating: Body corrected
		10-145	Figure 10.54 Example of Initial Output in Complementary PWM Mode (MTU3 and MTU4) (1), corrected
		10-146	Figure 10.55 Example of Initial Output in Complementary PWM Mode (MTU3 and MTU4) (2), corrected
		10-147	Figure 10.56 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (1), corrected
		10-148	Figure 10.57 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (2), corrected
		10-148	Figure 10.58 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (3), corrected
		10-149	Figure 10.59 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (1), corrected
		10-149	Figure 10.60 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (2), corrected
		10-150	Figure 10.61 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (3), corrected
		10-150	Figure 10.62 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (4), corrected
		10-151	Figure 10.63 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (5), corrected
		10-154	Figure 10.67 Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode (MTU3 and MTU4): Descriptions of [1] to [3] corrected
		10-158	Figure 10.73 Example of Procedure for Suppressing Synchronous Counter Clearing for MTU0 to MTU2, and MTU6 and MTU7: Descriptions of [1] and [3] corrected, Note 1 corrected
		10-162	10.3.8 Complementary PWM Mode (q) Example of Waveform Output for Driving AC Synchronous Motor (Brushless DC Motor): Body corrected
		10-164, 10-165	10.3.8 Complementary PWM Mode (s) Double Buffer Function in Complementary PWM Mode, corrected
		10-168	Figure 10.88 Example of Interrupt Skipping Function 1, corrected
		10-171	Figure 10.91 Relationship between Bits T3AEN and T4VEN in TITCR1A and Buffer Transfer-Enabled Period, corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	10-172	10.3.8 Complementary PWM Mode (a) Register and Counter Miswrite Prevention Function: Applicable registers corrected
		10-173	10.3.9 A/D Converter Start Request Delaying Function: Body corrected
		10-173	Figure 10.92 Example of Procedure for Specifying A/D Converter Start Request Delaying Function (MTU3 and MTU4): Description of [2] corrected, Note corrected, Note added
		10-174	Figure 10.93 Basic Example of A/D Converter Start Request Signal (TRG4AN) Operation, corrected
		10-174	10.3.9 A/D Converter Start Request Delaying Function (3) Period in which A/D Converter Start Requests are Enabled: added
		10-175	10.3.9 A/D Converter Start Request Delaying Function (4) Buffer Transfer: Body corrected
		10-175	Figure 10.94 Example of A/D Converter Start Request Signal (TRG4AN) and Buffer Transfer Operation, corrected
		10-176	10.3.9 A/D Converter Start Request Delaying Function (5) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 1: Body corrected
		10-176	Figure 10.95 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping Function 1 (UT4AE and DT4AE = 1), corrected
		10-177	Figure 10.96 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping Function 1 (UT4AE = 1, DT4AE = 0), corrected
		10-178	10.3.9 A/D Converter Start Request Delaying Function (6) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 2: Body corrected
		10-178	Figure 10.98 Example of Interrupt Skipping 2 Operation (Skipping Count is Set to Four), corrected
		10-179	10.3.10 Synchronous Operation of MTU0 to MTU4, MTU6, and MTU7: Title corrected
		10-179	10.3.10 Synchronous Operation of MTU0 to MTU4, MTU6, and MTU7 (1) Synchronous Counter Start for MTU0 to MTU4, MTU6, and MTU7: Title corrected, Body corrected
		10-179	10.3.10 Synchronous Operation of MTU0 to MTU4, MTU6, and MTU7 (a) Example of Procedure for Setting Synchronous Counter Start for MTU0 to MTU4, MTU6, and MTU7: Title corrected, Body corrected
		10-179	Figure 10.99 Example of Procedure for Specifying Synchronous Counter Start in MTU0 to MTU4, MTU6, and MTU7: Title corrected, Processing after start of counting corrected, Description of [4] corrected, Note corrected
		10-179	10.3.10 Synchronous Operation of MTU0 to MTU4, MTU6, and MTU7 (b) Examples of Synchronous Counter Start Operation: Body corrected
		10-179	Figure 10.100 Example of Synchronous Counter Start Operation in MTU0 to MTU4, MTU6, and MTU7: Title corrected
		10-180	10.3.10 Synchronous Operation of MTU0 to MTU4, MTU6, and MTU7 (a) Example of Procedure for Specifying Synchronous Counter Clearing for MTU6 and MTU7: Title corrected, Body corrected
		10-180	Figure 10.101 Example of Procedure for Specifying Synchronous Counter Clearing in MTU6 and MTU7: Title corrected, Descriptions of start and end processing corrected
		10-180	10.3.10 Synchronous Operation of MTU0 to MTU4, MTU6, and MTU7 (b) Examples of Synchronous Counter Clearing in MTU6 and MTU7: Title corrected, Body corrected
		10-180	Figure 10.102 Example of Synchronous Counter Clearing in MTU6 and MTU7 (1): Title corrected
		10-181	Figure 10.103 Example of Synchronous Counter Clearing in MTU6 and MTU7 (2): Title corrected
		10-182	Figure 10.105 Example of External Pulse Width Measurement (Measuring High Pulse Width), corrected
		10-183	10.3.12 Dead Time Compensation: Body corrected
		10-183	Figure 10.107 Delay in Dead Time in Complementary PWM Operation: Location moved
		10-184	Figure 10.108 Example of Dead Time Compensation Setting Procedure: Processing corrected
		10-185	10.3.13 TCNTU, TCNTV, and TCNTW Capture at Crest and/or Trough in Complementary PWM Mode: Title corrected, Body corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	10-185	Figure 10.109 TCNTU Capture at Crest and Trough in Complementary PWM Operation: Title corrected, Fiugre corrected
		10-188	Table 10.78 MTU Interrupt Sources: Note 1 corrected
		10-188	10.4.1 Interrupt Sources and Priorities (1) Input Capture/Compare Match Interrupt: Body corrected
		10-188	10.4.1 Interrupt Sources and Priorities (2) Overflow Interrupt: Body corrected
		10-188	10.4.2 DMAC Activation (1) DMAC Activation: Body corrected
		10-189	10.4.3 A/D Converter Activation (1) A/D Converter Activation by TGRA Input Capture/Compare Match or at MTU4.TCNT (MTU7.TCNT) Trough in Complementary PWM Mode: Description of suffix in body corrected
		10-189	10.4.3 A/D Converter Activation (2) A/D Converter Activation by Compare Match between MTU0.TCNT and MTU0.TGRE: Body corrected
		10-190	Table 10.79 Interrupt Sources and A/D Converter Start Request Signals: Note 1 corrected
		10-191	Figure 10.111 Count Timing in Internal Clock Operation (MTU0 to MTU4 and MTU6 to MTU8), corrected
		10-191	Figure 10.112 Count Timing in Internal Clock Operation (MTU5), corrected
		10-191	Figure 10.113 Count Timing in External Clock Operation (MTU0 to MTU4, MTU6 to MTU8): Title corrected, Figure corrected
		10-191	Figure 10.114 Count Timing in External Clock Operation (Phase Counting Mode), corrected
		10-192	10.5.1 Input/Output Timing (2) Output Compare Output Timing: Body corrected
		10-192	Figure 10.115 Output Compare Output Timing (Normal Mode or PWM Mode) (n = 0 to 4, 6, 7, 8; m = A to D), corrected
		10-192	Figure 10.116 Output Compare Output Timing (Complementary PWM Mode or Reset-Synchronized PWM Mode) (n = 0 to 4, 6, 7, 8; m = A to D), corrected
		10-196	Figure 10.124 Transfer Timing from Buffer Register to Temporary Register (TCNTSA Stopped), corrected
		10-196	Figure 10.125 Transfer Timing from Buffer Register to Temporary Register (TCNTSA Operating), corrected
		10-197	Figure 10.127 TGI Interrupt Timing (Compare Match) (MTU0 to MTU4 and MTU6 to MTU8), corrected
		10-197	Figure 10.128 TGI Interrupt Timing (Compare Match) (MTU5), corrected
		10-199	Figure 10.131 TCIV Interrupt Timing, corrected
		10-199	Figure 10.132 TCIU Interrupt Timing, corrected
		10-200	10.6.2 Count Clock Restrictions: Title corrected, Body corrected
		10-200	Figure 10.133 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode, corrected
		10-200	10.6.3 Note on Cycle Setting: Formula for frequency (f) in the case of MTU0 to MTU4 and MTU6 to MTU8 corrected
		10-201	Figure 10.135 Contention between TCNT Write and Increment Operations, corrected
		10-203	10.6.8 Contention between Buffer Register Write and TCNT Clear Operations: Body corrected
		10-205	10.6.12 Contention between MTU2.TCNT Write Operation and Overflow/Underflow in Cascaded Operation: Body corrected
		10-205	Figure 10.143 Contention between MTU2.TCNT Write Operation and Overflow/Underflow in Cascaded Operation, corrected
		10-207	10.6.15 Buffer Operation and Compare Match in Reset-Synchronized PWM Mode: Body corrected
		10-207	Figure 10.145 Buffer Operation and Compare Match in Reset-Synchronized PWM Mode, corrected
		10-208	10.6.16 Overflow in Reset-Synchronized PWM Mode: Body corrected
		10-209	10.6.17 Contention between Overflow/Underflow and Counter Clearing: Body corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	10-209	Figure 10.147 Contention between Overflow and Counter Clearing, corrected
		10-210	10.6.18 Contention between TCNT Write Operation and Overflow/Underflow: Body corrected
		10-210	Figure 10.148 Contention between TCNT Write Operation and Overflow, corrected
		10-210	10.6.20 Output Level in Complementary PWM Mode and Reset-Synchronized PWM Mode: Body corrected
		10-212	10.6.24 Notes Regarding Timer Counter (MTU5.TCNT) and Timer General Register (MTU5.TGR): Body corrected
		10-213	10.6.25 Notes to Prevent Malfunctions in Synchronous Clearing for Complementary PWM Mode: Body corrected
		10-215	Figure 10.151 Continuous Output of Interrupt Signal in Response to a Compare Match, corrected
		10-216	10.6.27 Notes on A/D Converter Start Request Delaying Function in Complementary PWM Mode, added
		10-216	Figure 10.152 A/D Converter Start Request when TADCOBRA is Set to 0 in MTU4, added
		10-217	Figure 10.153 A/D Converter Start Request when TADCOBRA is Set to the Same Value as TCDR in MTU4, added
		10-219	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation: Body corrected
		10-220	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (1) Operation When Error Occurs in Normal Mode and Operation is Restarted in Normal Mode: Descriptions of operations (3), (6), (10), and (14) corrected
		10-221	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (2) Operation When Error Occurs in Normal Mode and Operation is Restarted in PWM Mode 1: Description of operation (14) corrected
		10-221	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (3) Operation When Error Occurs in Normal Mode and Operation is Restarted in PWM mode 2: Description of operation (12) corrected
		10-223	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (5) Operation When Error Occurs in Normal Mode and Operation is Restarted in Complementary PWM Mode: Descriptions of operations (13), (14), (16), and (18) corrected
		10-224	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (6) Operation When Error Occurs in Normal Mode and Operation is Restarted in Reset-Synchronized PWM Mode: Destinations for reference of operations (1) to (13) corrected, Descriptions of operations (14), (16), and (18) corrected
		10-225	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (7) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Normal Mode: Descriptions of operations (3), (6), (10), and (14) corrected
		10-226	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (8) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in PWM mode 1: Description of operation (14) corrected
		10-226	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (9) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in PWM mode 2: Description of operation (12) corrected
		10-228	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (11) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Complementary PWM Mode: Descriptions of operations (14), (15), (17), and (19) corrected
		10-229	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (12) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Reset-Synchronized PWM Mode: Descriptions of operations (15), (17), and (19) corrected
		10-230	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (13) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in Normal Mode: Description of operation (3) corrected
		10-232	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (15) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in PWM Mode 2: Description of operation (11) corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	10-234	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (19) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in PWM Mode 2: Description of operation (11) corrected
		10-236	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (21) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Normal Mode: Descriptions of operations (2), (4), (6), (10), and (14) corrected
		10-237	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (22) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in PWM Mode 1: Description of operation (14) corrected
		10-238	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (23) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Complementary PWM Mode: Description of operation (12) corrected
		10-239	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (24) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Complementary PWM Mode with New Settings: Descriptions of operations (12), (13), (15), and (17) corrected
		10-240	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (25) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Reset-Synchronized PWM Mode: Descriptions of operations (12), (13), (15), and (17) corrected
		10-241	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (26) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Normal Mode: Descriptions of operations (2), (4), (6), (10), and (14) corrected
		10-242	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (27) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in PWM Mode 1: Description of operation (14) corrected
		10-243	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (28) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Complementary PWM Mode: Descriptions of operations (11), (12), (14), and (16) corrected
		10-244	10.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation (29) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Reset-Synchronized PWM Mode: Description of operation (12) corrected
		11. Port Output Enable 3 (POE3)	
		11-16	11.2.11 Port Output Enable Control Register 5 (POECR5): Symbol of bit 4 (IC4ADDMT0ZE), bit 2 (IC2ADDMT0ZE), and bit1 (IC1ADDMT0ZE) corrected
		12. General PWM Timer (GPT)	
		All	Suffix corrected as follows: GTADTRm → GTADTRn, GTADTBRm → GTADTBRn, GTADTDBRm → GTADTDBRn, GTADTBRm → GTADTBRn, GTDVm → GTDVn, GTDBm → GTDBn
		12-3	Figure 12.1 GPT block diagram, corrected
		12-7	12.2.1 General PWM Timer Write-Protection Register (GTWP): Register name corrected (GTIBCSR → GTICBSR)
		12-8	12.2.2 General PWM Timer Software Start Register (GTSTR): Body corrected
		12-9	12.2.3 General PWM Timer Software Stop Register (GTSTP): Body corrected
		12-9	12.2.4 General PWM Timer Software Clear Register (GTCLR): Body corrected
		12-23	12.2.8 General PWM Timer Up Count Source Select Register (GTUPSR): Body corrected
		12-26	12.2.9 General PWM Timer Down Count Source Select Register (GTDNSR): Body corrected
		12-34	12.2.12 General PWM Timer Control Register (GTCR): Description of b18 to b16 (MD[2:0]) in Bit table corrected
		12-36, 12-37	12.2.13 General PWM Timer Count Direction and Duty Setting Register (GTUDDTYC): b17 and b16 of Bit table corrected, Description of the OmDTYR bit corrected
		12-39, 12-40	12.2.14 General PWM Timer I/O Control Register (GTIOR): Description of the OADF[1:0] and OBDF[1:0] bits corrected
		12-44	12.2.15 General PWM Timer Interrupt Output Setting Register (GTINTAD): Description of the GRP[1:0] bit corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	12-45 to 12-47	12.2.16 General PWM Timer Status Register (GTST): Bit names of bit 5 (TCFF), bit 4 (TCFE), bit 3 (TCFD), and bit 2 (TCFC) corrected
		12-51, 12-52	12.2.17 General PWM Timer Buffer Enable Register (GTBER): Descriptions of bits 25 and 24 (ADTTA[1:0]) and bits 29 and 28 (ADTTB[1:0]) in Bit table corrected, Description of the PR[1:0] bits corrected
		12-54	12.2.18 General PWM Timer Interrupt and A/D Converter Start Request Skipping Setting Register (GTITC): Body corrected
		12-58	12.2.24 A/D Converter Start Request Timing Register n (GTADTRn) (n = A, B): Suffix corrected
		12-58	12.2.25 A/D Converter Start Request Timing Buffer Register n (GTADTBRn): Suffix corrected, Register address of GPT32Em.GTADTBRB added
		12-58	12.2.26 A/D Converter Start Request Timing Double-Buffer Register n (GTADTDBRn) (n = A, B): Suffix corrected
		12-59	12.2.27 General PWM Timer Dead Time Control Register (GTDTCR): Description of the TDE bit corrected
		12-60	12.2.28 General PWM Timer Dead Time Value Register n (GTDVn) (n = U, D): Suffix corrected, Body corrected
		12-60	12.2.29 General PWM Timer Dead Time Buffer Register n (GTDBn) (n = U, D): Suffix corrected
		12-62	12.2.31 General PWM Timer Output Protection Function Temporary Release Register (GTSOTR): Description of b31 to b1 in Bit table corrected
		12-67	12.3.1.1 Counter Operation (4) Event count operation in up-counting using hardware sources: Body corrected
		12-68	12.3.1.1 Counter Operation (5) Event count operation in down-counting using hardware sources: Body corrected
		12-68	Figure 12.9 Example setting for an event count operation in down-counting using hardware sources: Description of setting count source corrected
		12-69	12.3.1.1 Counter Operation (6) Counter clear operation: Body corrected
		12-76	12.3.2 Buffer Operation: Body corrected
		12-76	12.3.2.1 GTPR Register Buffer Operation: Body corrected
		12-79	12.3.2.2 Buffer Operation for GTCCRA and GTCCRB (1) When GTCCRA or GTCCRB functions as an output compare register: Body corrected
		12-83	Figure 12.27 Example setting for GTCCRA and GTCCRB buffer operation with input capture: Description of setting operating mode corrected
		12-86	Figure 12.31 Example setting for GTADTRA and GTADTRB buffer operation: Description of setting buffer operation corrected
		12-89	12.3.3.2 Saw-Wave One-Shot Pulse Mode: Body corrected
		12-99	12.3.4 Automatic Dead Time Setting Function: Body corrected
		12-106	Figure 12.49 Example of output duty 0% and 100% functions: Setting examples corrected
		12-109	Figure 12.55 Example setting for count start/stop operation by hardware source: Descriptions of setting hardware count start and setting hardware count stop corrected
		12-111	Figure 12.58 Example setting for count clearing operation by hardware source: Description of setting hardware source operation corrected
		12-112	12.3.7.3 Hardware Clear Operation: Body corrected
		12-115	12.3.8.2 Synchronized Operation by Hardware: Body corrected
		12-115	Figure 12.62 Example of a simultaneous start, stop, and clear by the hardware sources, with the same count cycle (GTPR register value): Title corrected
		12-132	Table 12.18 Conditions of up-counting and down-counting in phase counting mode 5 (B): Register setting corrected
		12-136	12.4.1 Interrupt Sources and Priorities (3) CCMPAn interrupt (n = 0 to 7): Body corrected
		12-136	12.4.1 Interrupt Sources and Priorities (4) CCMPBn interrupt (n = 0 to 7): Body corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	12-143	Figure 12.87 Example setting for A/D converter start request timing operation: Descriptions of setting operating mode and starting count operation corrected
		12-147	12.7 Noise Filter Function: Body corrected
		12-148	12.8.1 Write-Protection for Registers: Register name corrected (GTIBCSR → GTICBSR)
		12-158	Figure 12.100 Example of temporary cancellation of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low, corrected
		12-160	12.10.1 Settings for the Module-Stop Function: Body corrected
		12-161	12.10.4 Starting and Stopping the GTCNT Counter: Body corrected
		12-161	12.10.5 Priority On Conflicts (3) GTCCRn registers (n = A to F): Suffix corrected
		12-162	12.10.5 Priority On Conflicts (5) GTADTRn registers (n = A, B): Suffix corrected
		12-162	12.10.5 Priority On Conflicts (6) GTDVn registers (n = U, D): Suffix corrected
		13. Port Output Enable for GPT (POEG)	
		13-2	Figure 13.1 POEG block diagram, corrected
		13-5	13.3 Output-Disable Control Operation: Body corrected
		13-8	13.5 External Trigger Output to the GPT: Bit name corrected (POEGG.INV → POEGGn.INV)
		13-9	13.6.2 Specifying Pins Associated with the GPT: Body corrected
		16. Realtime Clock (RTC)	
		16-2	Table 16.2 RTC specifications: Description of Interrupts corrected
		16-3	Figure 16.1 RTC block diagram, corrected
		16-5	Table 16.4 Register Configuration: Note corrected to Note 1, Note 2 added
		16-6	16.2.1 64-Hz Counter (R64CNT): Counter notation corrected (R64CNT counter → R64CNT)
		16-7	16.2.2 Second Counter (RSECCNT) / Binary Counter 0 (BCNT0) (1) In calendar count mode: Counter notation corrected (RSECCNT counter → RSECCNT)
		16-8	16.2.3 Minute Counter (RMINCNT) / Binary Counter 1 (BCNT1) (1) In calendar count mode: Counter notation corrected (RMINCNT counter → RMINCNT)
		16-8	16.2.3 Minute Counter (RMINCNT) / Binary Counter 1 (BCNT1) (2) In binary count mode: Counter notation corrected (BCNT1 counter → BCNT1)
		16-9	16.2.4 Hour Counter (RHRCNT) / Binary Counter 2 (BCNT2) (1) In calendar count mode: Counter notation corrected (RHRCNT counter → RHRCNT)
		16-9	16.2.4 Hour Counter (RHRCNT) / Binary Counter 2 (BCNT2) (2) In binary count mode: Counter notation corrected (BCNT2 counter → BCNT2)
		16-10	16.2.5 Day-of-Week Counter (RWKCNT) / Binary Counter 3 (BCNT3) (1) In calendar count mode: Counter notation corrected (RWKCNT counter → RWKCNT)
		16-11	16.2.6 Day Counter (RDAYCNT): Counter notation corrected (RDAYCNT counter → RDAYCNT), Body corrected
		16-11	16.2.7 Month Counter (RMONCNT): Counter notation corrected (RMONCNT counter → RMONCNT)
		16-12	16.2.8 Year Counter (RYRCNT): Description of bits 7 to 4 (YR10[3:0]) Function of b7 to b4 (YR10[3:0]) in Bit table corrected, Counter notation corrected (RYRCNT counter → RYRCNT)
		16-13	16.2.9 Second Alarm Register (RSECAR) / Binary Counter 0 Alarm Register (BCNT0AR) (1) In calendar count mode: Description of bit7 (ENB) Counter notation corrected (RSECCNT counter → RSECCNT)
		16-14	16.2.10 Minute Alarm Register (RMINAR) / Binary Counter 1 Alarm Register (BCNT1AR) (1) In calendar count mode: Description of bit7 (ENB) Counter notation corrected (RMINCNT counter → RMINCNT)

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	16-15	16.2.11 Hour Alarm Register (RHRAR) / Binary Counter 2 Alarm Register (BCNT2AR) (1) In calendar count mode: Description of bit7 (ENB) Counter notation corrected (RHRCNT counter → RHRCNT)
		16-16	16.2.12 Day-of-Week Alarm Register (RWKAR) / Binary Counter 3 Alarm Register (BCNT3AR) (1) In calendar count mode: Description of bit7 (ENB) Counter notation corrected (RWKCNT counter → RWKCNT)
		16-17	16.2.13 Day Alarm Register (RDAYAR) / Binary Counter 0 Alarm Enable Register (BCNT0AER) (1) In calendar count mode: Description of bit7 (ENB) Counter notation corrected (RDAYCNT counter → RDAYCNT)
		16-18	16.2.14 Month Alarm Register (RMONAR) / Binary Counter 1 Alarm Enable Register (BCNT1AER) (1) In calendar count mode: Description of bit7 (ENB) Counter notation corrected (RMONCNT counter → RMONCNT)
		16-20	16.2.16 Year Alarm Enable Register (RYRAREN) / Binary Counter 3 Alarm Enable Register (BCNT3AER) (1) In calendar count mode: Register and counter notation corrected (RYRCNT counter → RYRCNT)
		16-23	16.2.18 RTC Control Register 1 (RCR1): Register notation corrected (RCR1 register → RCR1), Description of the PIE bit corrected
		16-24	16.2.19 RTC Control Register 2 (RCR2) (1) In calendar count mode: Body corrected
		16-28	16.2.20 RTC Control Register 3 (RCR3): Register notation corrected (RCR3 register → RCR3)
		16-28	16.2.21 RTC Control Register 4 (RCR4): Register notation corrected (RCR4 register → RCR4)
		16-29	16.2.22 Frequency Register H/L (RFRH/L): Undefined value x and its description added to Bit map of the RFRH register
		16-32	Figure 16.2 Outline of initial settings after a power on: Processing of "Set the time capture control register" deleted
		16-39	16.3.8.1 Automatic Adjustment: Register notation corrected (RADJ register → RADJ)
		16-41	16.3.8.2 Adjustment by Software: Register notation corrected (RADJ register → RADJ)
		16-44	16.5.1 Register Writing during Counting: RCR1.RTCOS and RCR2.RTCOE registers deleted
		16-45	16.5.4 Notes When Writing to and Reading from Registers: Body corrected
		18. Serial Communications Interface (SCIg)	
		18-1	18. Serial Communications Interface (SCIg): Body corrected
		18-23	18.2. 11 Bit Rate Register (BRR): Body corrected
		18-23	Table 18.6 Relationship between N Setting in BRR and Bit Rate B: "Clock synchronous" row corrected
		18-29	Table 18.14 BRR Settings for Various Bit Rates (Smart Card Interface Mode, n = 0, S = 372): "Error" column corrected
		18-31	Table 18.17 Relationship between MDDR Setting (M) and Bit Rate (B) When Bit Rate Modulation Function is Used: "Clock synchronous*1" and "Smart card interface" under the "Mode" column corrected, Note 1 corrected
		18-79	18.9.4 Receive Error Flags and Transmit Operations (Clock Synchronous Mode): Title corrected
		20. SPI Multi I/O Bus Controller	
		All	Mode names corrected as follows: Manual operating mode → Manual mode and External Address Space Read Operation → External Address Space Read Mode
		20-2	20.1.5 Manual Mode: Body corrected
		20-6	20.4.1 Common Control Register (CMNCR): Body corrected, R/W of bit 24 corrected, Description of bit 31 (MD) corrected
		20-15	20.4.7 Data Read Enable Setting Register (DRENR): Description of bits 11 to 8 (ADE[3:0]) corrected
		20-20	20.4.12 Manual Mode Enable Setting Register (SMENR): Description of bits 11 to 8 (ADE[3:0]) corrected
		20-24	20.4.16 Manual Mode Write Data Register 1 (SMWDR1): Body corrected
		20-26	20.4.18 Data Read Dummy Cycle Setting Register (DRDMCR): Body corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	20-28	20.4.20 Manual Mode Dummy Cycle Setting Register (SMDMCR): Body corrected
		20-36	20.4.27 PHY Adjustment Register 2 (PHYADJ2): Description of bits 31 to 0 (ADJ2[31:0]) corrected
		20-37	20.5.1 System Configuration: Body corrected
		20-39	Figure 20.4 System Configuration Example with HyperFlash Connected, corrected
		20-40	Figure 20.5 System Configuration Example with OctaFlash Connected, corrected
		20-41	20.5.2 Address Map: Body corrected
		20-52	Table 20.4 Data Registers: "Dummy cycle" row corrected
		20-52	Table 20.5 Data Registers (HyperFlash): "Dummy cycle" row corrected
		20-53	20.5.8 Command Sequence (2) Data Enable: Body corrected
		20-53	Figure 20.18 Data and Enable (Serial Flash), corrected
		20-55	20.5.8 Command Sequence (3) Bit Size: Body corrected
		20-55	Figure 20.19 Transfer Format Example with 1-Bit Data Size and One Serial Flash Memory Connected, corrected
		20-55	Figure 20.20 Transfer Format Example with 1-Bit Data Size and Two Serial Flash Memories Connected, corrected
		20-56	Figure 20.21 Transfer Format Example with 4-Bit Data Size and One Serial Flash Memory Connected, corrected
		20-57	Figure 20.23 Transfer Format Example in the HyperFlash Command and Address Phase, corrected
		20-64	20.5.17 Timing Adjustment: Body corrected
		20-64	Figure 20.28 (1) Flow of Timing Adjustment when the Serial Flash Memory is Connected (SDR Mode): Title corrected
		20-65	Figure 20.28 (2) Flow of Timing Adjustment when the Serial Flash Memory is Connected (SDR Mode): Title corrected
		20-66	Figure 20.28 (3) Example of Timing Adjustment when the Serial Flash Memory is Connected (SDR Mode): Title corrected
		20-67	20.5.18 Data Alignment: Body corrected
		21. HyperBus™ Controller	
		21-2	Figure 21.2 Example of Connections between this LSI and Hyper and HyperRAM Devices, corrected
		21-3	Table 21.1 List of Input/Output Pins: Note added
		21-15	21.6.2.3 One-Byte Write Access: Title corrected, Body corrected
		21-15	Figure 21.6 Waveforms of One-Byte Writing: Title and figure corrected
		21-16	Table 21.3 Command/Address Bit Assignments: Description of bit 45 (Burst type) corrected
		21-17	21.6.3.1 Write Operation Flow: Body corrected
		21-19	21.6.3.3 Flow of Write and Read Operations for the Configuration Register: Body corrected
		21-20	Figure 21.11 Timing Chart of Read and Write Operations, corrected
		22. Octa Memory Controller	
		All	Notation of hexadecimal numbers corrected
		22-3	Table 22.1 Configuration of Hyper Memory Controller Pins: Note added
		22-3	22.3.1 Device Interface: Body corrected
		22-4	22.4 Register Descriptions: Body corrected
		22-7	22.4.3 Device Command Setting Register (DCSR): Description of bits 15 to 8 (DMLEN[7:0]) corrected
		22-8	22.4.4 Device Size Register 0 (DSR0): Description of bits 29 to 0 (DV0SZ[29:0]) corrected
		22-8	22.4.5 Device Size Register 1 (DSR1): Description of bits 29 to 0 (DV1SZ[29:0]) corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	22-9	22.4.6 Memory Delay Trim Register (MDTR): Description of bits 27 to 24 (DQSEDOPI[3:0]), bits 23 to 16 (DV1DEL[7:0]), bits 15 to 12 (DQSESOPI[3:0]), bits 11 to 8 (DQSERAM[3:0]), and bit 7 to 0 (DV0DEL[7:0]) corrected
		22-10	Table 22.4 Examples of Setting the OM_DQS Enable Counter for OctaRAM: Title and heading corrected
		22-10	Table 22.5 Examples of Setting the OM_DQS Enable Counter for OctaFlash Memory: Title and header corrected
		22-11	22.4.7 Auto-Calibration Timer Register (ACTR): Body corrected
		22-11	22.4.8 Auto-Calibration Address Register 0 (ACAR0): Body corrected, Description of bits 31 to 0 (CAD0[31:0]) corrected
		22-12	22.4.9 Auto-Calibration Address Register 1 (ACAR1): Body corrected, Description of bits 31 to 0 (CAD1[31:0]) corrected
		22-13, 22-14	22.4.10 Device Memory Map Read Chip Select Timing Setting Register (DRCSTR): Description of bits 29 to 27 (DVRDHI1[2:0]) and bits 13 to 11 (DVRDHI0[2:0]) corrected
		22-18	22.4.12 Device Chip Select Timing Setting Register (DCSTR): Body corrected, Description of bits 13 to 11 (DVSELHI[2:0]) corrected
		22-20, 22-21	22.4.13 Controller and Device Setting Register (CDSR): Body corrected, Note of Bit table corrected
		22-22	22.4.14 Memory Map Dummy Length Register (MDLR): Body corrected, Description of bits 31 to 24 (DV1WDL[7:0]), bits 23 to 16 (DV1RDL[7:0]), bits 15 to 8 (DV0WDL[7:0]), and bits 7 to 0 (DV0RDL[7:0]) corrected, Note of Bit table corrected
		22-26	22.4.19 Configure Write without Data Register (CWNDR): Body corrected
		22-27	22.4.20 Configure Write Data Register (CWDR): Body corrected
		22-27	22.4.21 Configure Read Register (CRR): Body corrected
		22-29	Figure 22.2 Example of Connections between this LSI and OctaFlash and OctaRAM Devices, corrected
		22-32	Figure 22.6 Waveform of Write Operation in OctaRAM Interface, corrected
		22-33	Figure 22.8 Waveform of Read Operation in OctaFlash Interface (SOPI Mode), corrected
		22-33	Figure 22.9 Waveform of Read Operation in OctaFlash Interface (DOPI Mode with Pre-cycle Enabled), corrected
		22-34	Figure 22.10 Waveform of Read Operation in OctaRAM Interface (with Pre-cycle Enabled), corrected
		22-37	22.5.5 One-Byte Write Access to an Octa Memory Device in the DOPI Mode: Title corrected, Body corrected
		22-37	Figure 22.13 Waveforms in One-Byte Write Access in the DOPI Mode: Title and figure corrected
		22-38	Table 22.6 Example of Initial Settings of the Octa Memory Controller: Data of steps 4 and 6 corrected, Description of step 8 corrected
		22-42	22.6 Delaying OM_DQS and Auto-Calibration: Title corrected
		22-42	22.6.1 Delaying OM_DQS: Title corrected, Body corrected
		22-42	Figure 22.14 Example of Securing the Latching of Data by Delaying OM_DQS: Title and figure corrected
		22-42	22.6.2 OM_DQS Auto-Calibration: Body corrected
		—	22.6.2.1 Automatic Calibration Setting Flow: Title deleted
		22-43	Figure 22.15 Flow of Settings for the Auto-Calibration Mode: Title and figure corrected
		22-44	Figure 22.16 Example of the Initial Setting of the DV0DEL or DV1DEL Bits in MDTR, added
		22-45	22.7 OM_DQS Enable Counter: Body corrected, Note 1, Note 2, and Note 3 corrected
		22-46	Figure 22.17 Timing of the OM_DQS Enable Counter in DOPI Mode for the Flash Memory with 4 as the Number of Dummy Cycles (Latency), and OM_DQS Pre-Cycle Off: Title and figure corrected
		23. I^2C Bus Interface	
		23-3	23.2.1 Functional Overview: Description of SCL clock corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	23-24	23.3.6 RIICnFER — I²C Bus Function Enable Register: Description of NACKE bit corrected
		23-66	Figure 23.23 Block Diagram of Digital Noise Filter Circuit, corrected
		23-85	23.13.2 Extra SCL Clock Cycle Output Function: Body corrected
		23-85	Figure 23.40 Extra SCL Clock Cycle Output Function (CLO Bit), corrected
		24. Serial Sound Interface (SSIF-2)	
		All	I2S→I²S, corrected
		24-9	24.4.1.1 Control Register (SSICR) Descriptions of bits 21 to 19 (DWL[2:0]) and bits 18 to 16 (SWL[2:0]) in Bit table corrected
		25. CANFD Interface (RS-CANFD)	
		All	CAN FD→CANFD, corrected
		25-2	Table 25.3 Index: Descriptions of indexes d and b corrected
		25-4	Table 25.7 Interrupt Requests: RSCFD0 deleted
		25-4	Table 25.8 External Input/Output Signals: RSCFD0 deleted, CAN_CLK unit signal added
		25-5	Table 25.9 RS-CANFD Module Specifications: Specifications of Buffer and Reception filter function corrected
		25-12	25.3.2.1 RSCANnGRMCFG — Global Interface Mode Select Register: Description of RCMC bit corrected
		25-32	25.3.4.4 RSCANnGERFL — Global Error Flag Register: R/W of bits 17 and 16 in Bit map corrected
		25-38	Table 25.27 RSCANnGAFLECTR Register Contents: Function of bits 4 to 0 (AFLPN[4:0]) corrected
		25-38	25.3.5.1 RSCANnGAFLECTR — Receive Rule Entry Control Register: Description of AFLPN[4:0] bits corrected
		25-111	Table 25.82 RSCANnGTSTCFG Register Contents: Function of bits 22 to 16 (RTMPS[6:0]) corrected
		25-111	25.3.14.1 RSCANnGTSTCFG — Global Test Configuration Register: Description of RTMPS[6:0] bits corrected
		25-119	25.4.2.1 RSCFDnCFDGRMCFG — Global Interface Mode Select Register: Description of RCMC bit corrected
		25-135	25.4.3.5 RSCFDnCFDCmDCFG — Channel Data Bit Rate Configuration Register (m = 0, 1): Description of DBRP[7:0] bits corrected
		25-137, 25-138	25.4.3.6 RSCFDnCFDCmFDCFG — Channel CANFD Configuration Register (m = 0, 1): Description of GWEN bit corrected
		25-141	25.4.3.8 RSCFDnCFDCmFDSTS — Channel CANFD Status Register (m = 0, 1): Title corrected
		25-144, 25-145	Table 25.103 RSCFDnCFDGCFG Register Contents: Function of bit 12 (TSSS) and bit 4 (DCS) corrected
		25-151	25.4.4.4 RSCFDnCFDGERFL — Global Error Flag Register: R/W of bits 17 and 16 in Bit map corrected
		25-156	25.4.4.7 RSCFDnCFDGFDCFG — Global FD Configuration Register: Bit notation corrected (TSCCFG Bit → TSCCFG[1:0] Bits)
		25-157	25.4.5.1 RSCFDnCFDGAFLECTR — Receive Rule Entry Control Register: Initial values after reset of bits 3, 2, and 0 in Bit map corrected, Description of AFLPN[4:0] bits corrected
		25-157	Table 25.110 RSCFDnCFDGAFLECTR Register Contents: Function of bits 4 to 0 (AFLPN[4:0]) corrected
		25-166	25.4.6.2 RSCFDnCFDRMNDy — Receive Buffer New Data Register (y = 0): Title corrected
		25-184	25.4.8.1 RSCFDnCFDCFCCk — Transmit/receive FIFO Buffer Configuration and Control Register k (k = 0 to 5): Description of CFITSS bit corrected
		25-205	25.4.10.1 RSCFDnCFDCDTCT — DMA Enable Register: Function of bits 13 to 10 corrected (CFDMAE[5:2] → Reserved)
		25-207	25.4.10.2 RSCFDnCFDCDTSTS — DMA Status Register: Function of bits 13 to 10 corrected (CFDMASTS[5:2] → Reserved)

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	25-241	Table 25.167 RSCFDnCFDGTSTCFG Register Contents: Function of bits 22 to 16 (RTMPS[6:0]) corrected
		25-241	25.4.15.1 RSCFDnCFDGTSTCFG — Global Test Configuration Register: Description of RTMPS[6:0] bits corrected
		25-251	Table 25.173 Global Mode Transition Time: *1 added to "Two CAN bit times" and "Two CAN frames" under "Maximum Transition Time"
		25-255	Table 25.175 Operation in Transitions to Channel Reset Mode/Channel Halt Mode: Title corrected
		25-257	Table 25.176 Registers Initialized in Global Reset Mode or Channel Reset Mode: Note corrected
		25-258	Table 25.177 Registers Initialized Only in Global Reset Mode: Note corrected
		25-259	Figure 25.6 Entry of Receive Rules (for Setting Channels 0 and 1): Title corrected
		25-267	Figure 25.10 Interval Timer Block Diagram, corrected
		25-267	25.8.3.1 Interval Transmission Function: Description of (4) corrected
		25-269	25.8.6 Transmit History Function: Body corrected
		25-270	25.9.1 CAN-CANFD Gateway (Only in CANFD Mode): Body corrected
		25-273	25.10.5 RAM Test: Body corrected
		25-275	25.11.1 Initial Settings: Body corrected
		25-275	Figure 25.16 CAN Setting Procedure after the MCU is Reset, corrected
		25-276	Figure 25.17 Bit Timing Chart, corrected
		25-277	Figure 25.18 CAN Clock Control Block Diagram, corrected
		25-280	25.11.1.5 Buffer Setting: Body corrected
		25-280	Figure 25.20 Buffer Configuration, corrected
		25-281	Figure 25.21 Buffer Setting Procedure: Description of "Enable interrupts of buffers to be used" processing corrected
		25-283	Figure 25.23 Receive Buffer Reading Procedure: Note corrected
		25-285	Figure 25.25 Transmit/Receive FIFO Buffer Reading Procedure, corrected
		25-303	Figure 25.37 RAM Test Setting Procedure: *1 added
		27. Ethernet Controller (ETHERC)	
		27-36	27.5.3 Handling when Control Information Included a Mismatch: Body corrected
		27-36	27.5.4 Points to Note when the EPTPC Is not in Use, added
		27-36	27.5.5 Procedure of Resetting the Ethernet Controller, added
		28. PTP Module for the Ethernet Controller (EPTPCa)	
		28-12 to 28-54	28.2.1 MINT Interrupt Source Status Register (MIESR) to 28.2.35 Cut-Through Transfer Start Threshold Register (TRNCTTDR): Address notation corrected
		28-100	28.2.82 Response Message Reception Timeout Register (RSTOUTR): Register symbol added to Address
		28-101	28.2.83 PTP Reset Register (PTRSTR): Address notation corrected
		28-102	28.2.84 STCA Clock Select Register (STCSELR): Address notation corrected
		28-139	Table 28.27 Limits on the Values that can be Specified for a Pulse Output Timer: "Resolution of the cycle" and "Resolution of the pulse width" rows corrected
		28-146	28.5.1 Setting of the Module-Stop Function, added
		28.146	Table 28.29, added
		28.147	28.5.1.1 Release from the Module-Stop State, added
		28.147	28.5.1.2 Transition to the Module-Stop State, added
		28-147	28.5.2 Notes on Placing the CPU on Software Standby, added
		28-149	28.5.5 Transfer by PTPEDMAC when the Transparent Clock (TC) is in Use, added
		30. A/D Converter	
		30-1	30.1 Overview: Body corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	30-4	Table 30.2 Functions of A/D Converter: Names of triggers from GPT corrected
		30-17	30.2.7 A/D-Converted Value Addition/Average Mode Select Register 0 (ADADS0): Function of bits 7 to 0 in Bit table corrected
		30-23	Table 30.7 Selection of A/D Activation Sources by the TRSB[5:0] Bits (for Group B only): Names of triggers from GPT corrected
		30-24	Table 30.8 Selection of A/D Activation Sources by the TRSA[5:0] Bits: Names of triggers from GPT corrected
		30-27	Table 30.9 Selection of A/D Activation Sources by the TRSC[5:0] Bits (for Group C only): Names of triggers from GPT corrected
		30-40	30.2.21 A/D Compare Function Window-B Status Register (ADCMPBSR): Bits 15 to 8 in Bit map deleted, Bit numbers in Bit table corrected (b15 to b1 → b7 to b1)
		30-45	30.3.1 Scanning Operation: Names of triggers from GPT corrected
		30-49	30.3.2.4 Extended Operations When Double Trigger Mode is Selected: Names of triggers from GPT corrected
		30-54	30.3.4.2 A/D Conversion in Double Trigger Mode: Names of triggers from GPT corrected
		30-90	30.5.9 Range of Voltage on the Analog Input Pins, added
		30-90	30.5.10 Notes on Board Design, added
		30-90	30.5.11 Notes on Noise Prevention, added
		30-90	Figure 30.31 Example of a Protection Circuit for the Analog Inputs, added
		31. NAND Flash Controller	
		All	Sequence name corrected as follows: Universal command sequence, Generic Command Sequence → Generic Sequence
		31-1	31.1.1 Features: Description of DMA controller corrected
		31-2	Figure 31.1 Block Diagram: INTNFMA interrupt signal added
		31-2	31.1.2 Block Diagram: Body corrected
		31-5, 31-6	31.2.2 Main configurations register (CONTROL): Descriptions of bits 20 to 18 and bits 2 and 1 (ECC_BLOCK_SIZE[1:0]) in Bit table corrected
		31-7, 31-8	31.2.3 GENERIC_SEQ register (GEN_SEQ_CTRL): Descriptions of bits 11 and 10 (ROW_A1[1:0]), 9 and 8 (ROW_A0[1:0]), 7 and 6 (COL_A1[1:0]), and 5 and 4 (COL_A0[1:0]) corrected
		31-9	31.2.4 Controller status register (STATUS): Body corrected, Description of bit 9 (DATASIZE_ERROR_ST) in Bit table corrected
		31-10	31.2.5 LUN status register (LUN_STATUS_0): Body corrected
		31-14	31.2.8 ECC module control register (ECC_CTRL): Description of bits 2 to 0 (ECC_CAP[2:0]) in Bit table corrected
		31-41	31.2.34 MLUN register (MLUN): Description of bits 2 to 0 (MLUN_IDX[2:0]) in Bit table corrected
		31-43	Table 31.5 Instruction Encoding: Description of INPUT_SEL field corrected
		31-45	31.3.1 Command Generation (c) SEQ_2 Sequence: Body corrected
		31-46	31.3.1 Command Generation (d) SEQ_3 Sequence: Body corrected
		31-47	31.3.1 Command Generation (h) SEQ_7 Sequence: Body corrected
		31-48	31.3.1 Command Generation (j) SEQ_9 Sequence: Body corrected
		31-48	31.3.1 Command Generation (k) SEQ_10 Sequence: Body corrected
		31-48	31.3.1 Command Generation (l) SEQ_11 Sequence: Body corrected
		31-49	31.3.1 Command Generation (o) SEQ_14 Sequence: Body corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	31-50	31.3.1 Command Generation (p) SEQ_15 sequence: Body corrected
		31-51	31.3.1 Command Generation (v) SEQ_22 sequence: Body corrected
		31-63	31.3.3 Instructions (11) SELECT LUN WITH STATUS Command (a) Command Description: Body corrected
		31-66	31.3.3 Instructions (17) READ PAGE Command (a) Command Description: Body corrected
		31-83	Figure 31.32 Configuration: Processing 5 corrected
		31-84	31.3.7 Setup and Configuration (1) Send Data to NAND Flash via Slave Interface: Description numbered 2, and body corrected
		31-85	31.3.7 Setup and Configuration (2) Read Data from NAND Flash via Slave Interface: Descriptions numbered 2 and 4, and body corrected
		31-86	31.3.7 Setup and Configuration (3) Send Data to NAND Flash via Master Interface (using DMA): Descriptions numbered 2 and 4, and body corrected
		31-88	31.3.7 Setup and Configuration (4) Fast Writing and Reading of Several Pages from the Memory using DMA: Description numbered 2 corrected
		31-89	Figure 31.36 Fast Writing and Reading of Several Pages from the Memory Using DMA, corrected
		31-90	31.3.7 Setup and Configuration (5) Writing to Partial Pages: Description numbered 2 corrected
		31-90	Figure 31.37 Writing to Partial Pages: Description of processing 3 corrected
		31-91	31.3.7 Setup and Configuration (6) Reading Partial Pages: Descriptions numbered 2 and 4 corrected
		31-92	Figure 31.39 Block Diagram, corrected
		31-92	31.4.1 Block Diagram: Body corrected
		31-93	31.4.2 DMA (2) DMA Description: Body corrected
		31-94, 31-96	31.4.2 DMA (b) Scatter-Gather Mode: Body corrected
		—	Figure 31.40 Descriptor Fields for Scatter-Gather DMA (64 bits), deleted
		—	Table 31.47 Descriptor Fields for Scatter-Gather DMA (64 bits), deleted
		31-94	Figure 31.40 Descriptor Fields for Scatter-Gather DMA: Title corrected
		31-95	Table 31.47 Descriptor Fields for Scatter-Gather DMA: Title corrected
		31-96	Figure 31.41 Example of Scatter-Gather DMA Data Transfer, corrected
		31-98	31.4.4 BCH Algorithm Implementation: Body corrected
		32. USB 2.0 Host Module	
		32-1	32.1 Overview: Body corrected
		32-1	32.1.2 Features: Table corrected, Note 2 added to the table
		32-7	32.2 Register Descriptions: Title corrected
		32-12	Table 32.5 HcCommandStatus Register: Descriptions of bit 2 (BLF) and bit 1 (CLF) corrected
		32-25	Table 32.21 HcRhDescriptorA Register: Description of bit 12 (NOCP) corrected
		32-66	Table 32.54 BCCTRL1 Register: Descriptions of bit 5 (DCPMODE), bit 4 (VDMSRCE), bit 3 (IDPSINKE), bit 2 (VDPSRCE), bit 1 (IDMSINKE) and bit 0 (IDPSRCE) corrected
		32-67	32.2.4.5 UCOM Register (7) CC STATUS Register (offset: 840h): Initial values after reset of bits 28 to 25 in Bit map corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	32-67	Table 32.55 CC_STATUS Register: Description of bit 31 (CC_INT_SEL) and bit 5 (CC_PERI_STA) corrected
		32-68	Table 32.56 Conditions for Detecting the Connection of a Peripheral Device: Title corrected
		32-70	32.3 Clock Signals: Title corrected
		32-72	32.4 Interrupt Sources: Title corrected
		32-80	32.5 Power-Saving Function: Title corrected
		32-81	32.6 Battery Charging: Title corrected
		32-82	32.7 Bus Master: Title corrected
		32-82	32.7.1 Functional specifications of the bus master: Title corrected
		32-82	32.7.1.2 Issuing requests for different types of bus transfer: Title corrected
		32-85	32.8 Overcurrent Control and VBUS Control: Title corrected
		32-85	32.8.2 Overcurrent detection timer setting: Body corrected
		32-88	32.9 Procedure for Setting this Module: Title corrected
		32-88	Figure 32.1 Sequence Common to Both Host and Peripheral Modes: Title added, Processing corrected
		32-89	Figure 32.2 Initialization Sequence: Title added
		33. USB 2.0 Function Module	
		33-1	33.1.1 Overview: Body corrected
		33-5	33.1.4.2 Notes (2) Setting for battery charging when the peripheral controller is selected, added
		33-30	33.2.10.1 Interrupt Status Register 0 [INTSTS0] <Address: 040H>: Name of bit 15 (VBINT) corrected
		33-48	33.2.15.2 Pipe Configuration Register [PIPECFG] <Address: 068H>: Title corrected
		33-71	33.2.19.1 Low Power Status Register [LPSTS] <Address: 102H>: Bit number of SUSPM bit in Bit table corrected (9 → 14)
		33-74	33.2.22.1 Peripheral L1 Control Register 1 [PL1CTRL1] <Address: 144H>: Bit name of bits 7 to 4 in Bit map corrected
		33-100	33.8.2.1 Descriptor Interval Register [DSCITVL] <Address: 704H>: Name of bit 15 to 8 (DITVL) corrected
		33-147	33.9.12.1 Register mode/link mode • Notes on descriptors: Body corrected
		33-166	Table 33.58 List of Registers to be Saved: System configuration control register 1 (SYSCFG1) deleted
		41. LVDS Output Interface	
		41-6	41.4.4 LVDS PLL Setting Register (LPLLSETR): Description of bits 22 to 16 (LVDSPLL_FD[6:0]) corrected
		43. 2D Drawing Engine (DRW)	
		43-10	43.2.2 Surface Control Register (CONTROL2): Bit numbers in Bit table corrected (b31, 30 → b31, b30)
		45. JPEG Codec Unit	
		45-5	45.2.1 JPEG Code Mode Register (JCMOD): Description of bit 3 (DSP) corrected
		45-30	Figure 45.2 Compression Initial Setting Flow: Description of software reset corrected
		45-37	Figure 45.5 Decompression Initial Setting Flow: Description of software reset corrected
		46. Capture Engine Unit	
		46-20	46.4.6 Capture Interface Width Register (CAPWR): Bit number in Bit table corrected (12 to 10 → 12 to 0)
		46-65	46.5.3 Display in the Video Display Controller 6: Title corrected, Body corrected
		47. MIPI CSI2 Interface	
		47-2	Figure 47.1 CSI2 Block Diagram, corrected

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	47-9	47.2.5 Channel Data Type Select Register (VCDT): R/W of bits 20 to 16, descriptions of bits 9 and 8 (SEL_VC) and bit 6 (SEL_DT_ON), and initial values of bits 5 to 0 (SEL_DT) in Bit table corrected
		47-12	47.2.8 Automatic Standby Control Register (ASTBY): Initial value of bit 5 (VD_MSK_EN) in Bit table corrected
		47-40	47.2.39 ECC_CRCT Count Register (ECCCM): Bit numbers in Bit table corrected (31 to 4 → 31 to 8 and 3 to 0 → 7 to 0)
		47-53	47.3.5 Single-Channel Output: Destination for reference (Figure 47.8) added to body
		47-58 to 47-60	47.3.8 PHY Timing Setting, added
		47-59	Figure 47.11 Timing of TCLK_PREPARE and TCLK_SETTLE, added
		47-60	Figure 47.12 Timing of THS_PREPARE and THS_SETTLE, added
		47-60	Table 47.8 Setting Examples of PHY Timing Registers (when Bφ = 132 MHz, 1 UIINST = 1 ns), added
		47-61	Figure 47.13 Example 1 of Initial Setting Procedure for the Video Modules, corrected
		47-61	Figure 47.14 Example 2 of Initial Setting Procedure for the Video Modules, corrected
		47-65	Figure 47.17 Procedure for Starting PHY Block 1: Title corrected
		47-65	Figure 47.18 Procedure for Starting PHY Block 2: Title corrected
		—	Table 47.9 Example of PHY Timing Register Settings Based on Internal Bus Clock (Bφ) Frequency and Data Transfer Rate, deleted
		48. Video Input Module	
		48-1	Table 48.1 Input Interface for VIN: *2 added to "8-bit user defined data (RAW8)" under "Data Type", Note 2 corrected
		48-28	48.2.20 YC-RGB Conversion Coefficient Registers: Formula for YC → RGB color space conversion corrected
		49. SD/MMC Host Interface	
		49-13	49.2.7 SD Card Interrupt Flag Register (SD_INFO2): Description of bit 13 (SCLKDIVEN) corrected
		51. GPIO	
		51-4, 51-5	Table 51.2 Port Functions: *1 added to PG_0 to PG_7 and PJ_0 to PJ_7 under "Switching of Driving Ability", Note 1 added, Note corrected to Note 2
		51-19	51.3.5 GPIO Driving Ability Control Register (DSCR): Body corrected, Function in Bit table corrected, Note 1 corrected, Note 3 added, Note deleted
		51-28	Table 51.12 Register Settings for Input/Output Pin Function in 324-/272-/256-Pin Products: Function of P6_1 pin when the register setting is 111b corrected
		51-29	Table 51.13 Register Settings for Input/Output Pin Function in 176-Pin Products: Function of P6_1 pin when the register setting is 111b corrected
		52. Power-Down Modes	
		52-7	52.2.3 Standby Control Register 3 (STBCR3): Bit 1 in Bit map: Function corrected (MSTP31 → reserved), R/W corrected Bit 2 (MSTP32) in Bit table: Description corrected (CAN-FD → CANFD); Bit 1 in Bit table: Function corrected (MSTP31 → reserved), R/W and description corrected
		52-44	52.3.2 Software Standby Mode (1) Transition to Software Standby Mode Step 4 in the description titled "The procedure for switching to software standby mode is as follows:" corrected
		52-47	52.3.4 Deep Standby Mode (1) Transition to Deep Standby Mode: Step 5 in the procedure for the transition to deep standby mode corrected
		52-48	Figure 52.3 Flowchart of Transition to Deep Standby Mode, corrected
		52-49	Figure 52.4 Flowchart of Canceling Deep Standby Mode, corrected
		52-51	Table 52.5 External Memory Control Pins in Different Modes: Heading corrected to "Boot Mode 6 (OctaFlash Boot)"

Rev.	Date	Description	
		Page	Summary
3.00	Dec 06, 2019	54. Trusted Secure IP	
		54-5	Figure 54.4 Key Installation Process, corrected
		55. Register States	
		55-3	Table 55.1 Register States: Note numbers of the "Power-On Reset" and "Deep Standby" columns of DSFR, USBBSFR, XTALCTR, RTCXTALSEL registers in "Power-down modes" row corrected
		56. Electrical Characteristics	
		56-2	56.3 DC Characteristics: Condition corrected
		56-3	Table 56.2 DC Characteristics (2) [Current Consumption]: Typical values of Icc, PLLIcc, LVDSPLLIcc, MIPIAIcc18, UAPIcc, LVDSAPIcc, AIcc in the "Current consumption in normal operation" row and I_{sleep} in the "Current consumption in sleep mode" row corrected
		56-7	Table 56.3 Permissible Output Currents, corrected
		56-8	56.4 AC Characteristics: Conditions corrected
		56-13	Table 56.6 Control Signal Timing: Note corrected to Note 1, Note 2 added
		56-16	Table 56.8 Bus Timing: Address delay time 2 (t_{AD2}) corrected
		56-48	Table 56.15 SCIFA Timing: t_{SEcyc} corrected to t_{p1cyc}
		56-50	Table 56.17 Renesas Serial Peripheral Interface Timing: t_{cyc} corrected to t_{p1cyc}, Note added
		56-53	Table 56.18 SPI Multi I/O Bus Controller Timing: Clock cycle (t_{SPBcyc}) corrected, Data output buffer off time during SDR transfer (t_{BOFF}) corrected
		56-55	Table 56.20 Octa Memory Controller Timing: Minimum value of OM_CS hold time in DOPI writing corrected
		56-65	56.4.19 CANFD Interface Timing: Title corrected
		56-65	Table 56.23 CANFD Interface Timing: Title and heading corrected
		56-65	Figure 56.77 CANFD Interface Condition: Title and figure corrected
		56-74	Table 56.30 Video Display Controller 6 Timing: DV0_CLK input clock cycle (t_{Dcyc}), LCD0_EXTCLK input clock cycle (t_{Ecyc}), LCD0_CLK output clock cycle (t_{Lcyc}) corrected
		56-77	Table 56.32 Capture Engine Unit Signal Timing: Camera clock cycle (t_{VCYC})corrected
		56-89	56.5 A/D Converter Characteristics: Condition corrected
		56-89	Table 56.40 A/D Converter Characteristics: Title corrected
		57. States and Handling of Pins	
		57-5, 57-8	Table 57.1 Pin States: Interface names corrected (CAN interface → CANFD interface, MIPI → MIPI CSI2 interface)
		57-10	Table 57.2 Handling of Unused Pins (Except for Debugger Interface Pins): Pins added, Note 3 added
		57-12	Table 57.4 Handling of Pins in Deep Standby Mode: Handling of RREF0, RREF1 corrected
		Appendix	
		Appendix -3	Figure A.3 Dimensions of 272-Pin BGA Package, corrected
4.00	Nov 27, 2020	1. Overview	
		1-9	Table 1.1 Features of RZ/A2M: In specification of package, RENESAS Code of 256-pin BGA package corrected
		1-10	Table 1.2 Product Lineup: DRP Function and Trusted Secure IP of R7S921040VCBG to R7S921043VCBG, Trusted Secure IP of R7S921051VCBG to R7S921053VCBG, and RENESAS Code of 256-pin BGA package corrected
		3. Boot Mode	
		3-1	Body corrected and Note 4 added
		5. LSI Internal Bus	
		5-60	Table 5.7 Interrupt Signals for the Write Buffers: X2HPERI12_ERRINT added, and Write Buffer Connection Source/Destination of X2HPERI34_ERRINT, X2HPERI5_ERRINT, X2HPERI67_ERRINT, and X2HDBGR_ERRINT corrected

Rev.	Date	Description	
		Page	Summary
4.00	Nov 27, 2020	6. Clock Pulse Generator	
		6-16	Figure 6.5 Clock Signals for the System and Realtime Clock: Description of peripheral clock 1C added
		7. Interrupt Controller	
		7-28	Table 7.3 List of Interrupt IDs: Internal bus added
		8. Bus State Controller	
		8-8, 8-9	8.4.2 CSn Space Bus Control Register (CSnBCR): Bit name and descriptions of bits 24 to 22, 21 to 19, and 18 to 16 corrected
		9. Direct Memory Access Controller	
		9-56, 9-57	Table 9.4 On-Chip Module Requests: DMA transfer request signals of A/D converter and I^2C channels 0 to 3 corrected
		9-58	Table 9.4 On-Chip Module Requests: Setting of the CHCFG_n/nS.LVL bit for serial communication interface channels 0 and 1 and Ethernet MAC controller corrected
		9-59	Table 9.4 On-Chip Module Requests: DMA transfer request signals of DRP corrected
		16. Realtime Clock (RTC)	
		16-33	Figure 16.3 Clock and count mode setting procedure: Notes 1 and 2 added, and Notes 1 and 2 corrected to Notes 3 and 4
		18. Serial Communications Interface (SCIg)	
		18-2	Figure 18.1 Block Diagram of SCIg (SCI0 and SCI1): Input and output of RXDn and TXDn corrected
		20. SPI Multi I/O Bus Controller	
		20-4 to 20-71	All (Description of operation in Octal-SPI flash memory protocol mode): Octal-SPI flash memory corrected to Octal-SPI flash memory protocol mode
		20-1	20.1.2 OctaFlash™, Xccela™ Flash Memory Interface: Note added
		20-7	20.4.1 Common Control Register (CMNCR): Note added in description of bits 9 and 8 (IO0FV[1:0])
		20-12	20.4.5 Data Read Extended Address Setting Register (DREAR): Body corrected
		20-12	20.4.5 Data Read Extended Address Setting Register (DREAR): Descriptions of bits 23 to 16 (EAV[7:0]) and bits 2 to 0 (EAC[2:0]) corrected
		20-17	20.4.10 Manual Mode Address Setting Register (SMADR): Description of bits 31 to 24 (ADR[31:24]) corrected
		20-20	20.4.12 Manual Mode Enable Setting Register (SMENR): Description of bits 11 to 8 (ADE[3:0]) corrected
		20-27	20.4.19 Data Read DDR Enable Register (DRDRENR): Description of bits 14 to 12 (HYPE) corrected
		20-29	20.4.21 Manual Mode DDR Enable Register (SMDRENR): Description of bits 14 to 12 (HYPE[2:0]) corrected
		20-31	20.4.22 PHY Control Register (PHYCNT): Note 3 added in description of bit 18 (HS)
		20-32	20.4.23 PHY Offset Register 1 (PHYOFFSET1): Body corrected
		20-32	20.4.23 PHY Offset Register 1 (PHYOFFSET1): Description of bits 29 and 28 (DDRTMG) corrected
		20-36	20.4.26 PHY Adjustment Register 1 (PHYADJ1): Description of bits 31 to 0 (ADJ1[31:0]) corrected
		20-36	20.4.27 PHY Adjustment Register 2 (PHYADJ2): Description of bits 31 to 0 (ADJ2[31:0]) corrected
		20-41	20.5.2 Address Map: Body corrected
		20-41	Table 20.3 Address Map: Max. Access Area corrected
		20-42	20.5.3 32-bit Serial Flash Addresses: Description in the last paragraph corrected
		20-50	Figure 20.15 Data Transfer Timing using the SSLKP Bit: QSPIn_SSL corrected
		20-54	Figure 20.19 Data and Enable (Octal-SPI Flash Memory Protocol Mode), added
		20-64	Table 20.14 Supported Protocol for Serial Flash Memory: Bit Width 1-1-0 added
		20-64	Table 20.15 Supported Protocol for Octal-SPI Flash Memory: Bit Width 1-1-0 and 8-8-0 added
		20-65	20.5.17 Timing Adjustment: Body corrected

Rev.	Date	Description	
		Page	Summary
4.00	Nov 27, 2020	20-65	Figure 20.29 (1) Flow of Timing Adjustment when the Serial Flash Memory is Connected (SDR Mode Except in the Case of Octal-SPI Flash Memory): Title corrected
		20-66	Figure 20.29 (2) Flow of Timing Adjustment when the Serial Flash Memory is Connected (DDR Mode): Title corrected
		20-67	Figure 20.29 (3) Example of Timing Adjustment when the Serial Flash Memory is Connected (DDR Mode): Title corrected
		20-71	20.6.2 Transfer to read data while the signal on the QSPIn_SSL pin is asserted, deleted
		22. Octa Memory Controller	
		22-11	22.4.7 Auto-Calibration Timer Register (ACTR): In description of bits 31 to 0 (CTP[31:0]), automatic calibration cycle time corrected
		22-26	22.4.19 Configure Write without Data Register (CWNDR): Body corrected
		22-27	22.4.20 Configure Write Data Register (CWDR): Body corrected
		25. CANFD Interface (RS-CANFD)	
		25-3	Table 25.6 Range of Operating Frequency Depending on the Transfer Rate and the Number of Channels in Use in this LSI: Transfer Rate, clk_xincan, and clkc of CANFD corrected
		25-5	Table 25.9 RS-CANFD Module Specifications: In specification of communication speed, data bit rate in CANFD mode corrected
		25-305	25.12 Notes on the RS-CANFD Module: In the first bullet, "When changing interface mode without resetting the RS-CANFD" corrected to "When changing the interface mode without applying a power-on reset"
		32. USB 2.0 Host Module	
		32-4	32.1.2.6 Usage Notes, added
		32-6	32.1.4 Input/Output Pins: Pin name of OTG Power IC ID of channel 1 corrected
		32-27	Table 32.23 HcRhStatus Register: Description of bit 15 (DRWE) corrected (HcRhPortStatus register → HcRhPortStatus1 register)
		32-53	Table 32.43 SPD_CTRL Register: Description of bit 31 (SUSPENDM_ENABLE) corrected (HcRhPortStatus register → HcRhPortStatus1 register)
		32-62	Table 32.50 OBINTSTA Register: Descriptions of bit 6 (CHGDETCHG1_STA) and bit 4 (PDDETCHG1_STA) corrected
		32-65	Table 32.53 LINECTRL1 Register: Description of bits 21 and 20 (DSDP[1:0]) corrected
		32-66	Table 32.54 BCCTRL1 Register, corrected
		32-81	Table 32.59 Function of Charging Port: Title added
		32-81	32.6.1 Support of charging port: Description added
		32-82	32.6.1.1 When the host controller is to be used in the CDP mode, added
		32-83	32.6.1.2 When the host controller is to be used in the DCP mode, added
		32-83	32.6.2 Support of portable device: Description added
		32-84	Figure 32.2 Example of the Control Flow for Portable Devices, added
		32-95	32.11 Points for Caution, added
		32-95 to 32-104	32.11.1 Actions after Device Disconnection, added
		47. MIPI CSI2 Interface	
		47-18	47.2.12 Interrupt Source Mask Register (INTCLOSE): Note added
		47-62	47.3.9.1 Terminology, added
		48. Video Input Module	
		48-77	Table 48.14 Specifications: Description of UDS scaler setup sequence corrected
		51. GPIO	
		51-9	Figure 51.7 (7) TTL AND Input Buffer: JP0_3 corrected to TCK/SWDCLK
		51-10	Figure 51.9 (9) Bidirectional Buffer with TTL AND Input and Latch: JP0_2 corrected to TMS/SWDIO

Rev.	Date	Description	
		Page	Summary
4.00	Nov 27, 2020	51-10	Figure 51.10 (10) Schmitt AND Input Buffer: JP0_4 corrected to TRST#
		52. Power-Down Modes	
		52-57	52.4.3 Usage Notes when USB 2.0 Host/Function Modules are Not to be Used: Body corrected
		52-57	Figure 52.5 Flow of Settings when USB 2.0 Host/Function Modules are Not to be Used: Title added
		56. Electrical Characteristics	
		56-65	Table 56.23 CANFD Interface Timing: Maximum transfer rate of CANFD corrected
		Appendix	
		Appendix -2	Figure A.2 Dimensions of 256-Pin BGA Package, corrected

RZ/A2M Group User's Manual: Hardware

Publication Date: Rev.1.00 Sep 25, 2018
Rev.4.00 Nov 27, 2020

Published by: Renesas Electronics Corporation

RZ/A2M Group

Renesas Electronics Corporation

R01UH0746EJ0400