LAN9730/LAN9730i

High-Speed Inter-Chip (HSIC) USB 2.0 to 10/100 Ethernet Controller

Highlights

- Single Chip HSIC USB 2.0 to 10/100 Ethernet Controller
- Integrated 10/100 Ethernet MAC with Full-Duplex Support
- Integrated 10/100 Ethernet PHY with HP Auto-MDIX Support
- Integrated USB 2.0 Hi-Speed Device Controller
- Integrated HSIC Interface
- Implements Reduced Power Operating Modes

Target Applications

- Embedded Systems
- Set-Top Boxes
- PVRs
- CE Devices
- Networked Printers
- USB Port Replicators
- Test Instrumentation
- Industrial

Key Features

- USB Device Controller
  - Fully compliant with Hi-Speed Universal Serial Bus Specification, revision 2.0
  - Supports HS (480 Mbps) mode
  - Four Endpoints supported
  - Supports vendor specific commands
  - Integrated HSIC Interface
  - Remote wakeup supported
- High-Performance 10/100 Ethernet Controller
  - Fully compliant with IEEE 802.3/802.3u
  - Integrated Ethernet MAC and PHY
  - 10BASE-T and 100BASE-TX support
  - Full- and half-duplex support
  - Full- and half-duplex flow control
  - Preamble generation and removal
  - Automatic 32-bit CRC generation and checking
  - Automatic payload padding and pad removal
  - Loop-back modes
  - TCP/UDP/IP/ICMP checksum offload support
  - Flexible address filtering modes
    - One 48-bit perfect address
    - 64 hash-filtered multicast addresses
    - Pass all multicast
    - Promiscuous mode
    - Inverse filtering
    - Pass all incoming with status report
  - Wakeup packet support
  - Integrated Ethernet PHY
    - Auto-negotiation
    - Automatic polarity detection and correction
    - HP Auto-MDIX support
    - Link status change wake-up detection
  - Support for three status LEDs
  - External MII and Turbo MII support HomePNA® and HomePlug® PHY
- Power and I/Os
  - Various low power modes
  - Supports PCI-like PME wake when USB host disabled
  - 11 GPIOs
  - Supports bus-powered and self-powered operation
  - Integrated power-on reset circuit
  - Single external 3.3 V I/O supply
    - Optional internal core regulator
- Miscellaneous Features
  - EEPROM controller
  - Supports custom operation without EEPROM
  - IEEE 1149.1 (JTAG) boundary scan
  - Requires single 25 MHz crystal
- Software
  - Windows® 8/7/XP/Vista driver
  - Linux® driver
  - Win CE driver
  - MAC® OS driver
  - EEPROM utility
- Packaging
  - 56-pin VQFN (8 x 8 mm), RoHS-compliant
- Environmental
  - Commercial Temperature Range (0°C to +70°C)
  - Industrial Temperature Range (-40°C to +85°C)

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@microchip.com**. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000000A is version A of document DS30000000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site: **http://www.microchip.com**
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include -literature number) you are using.

Customer Notification System

Register on our web site at **www.microchip.com** to receive the most current information on all of our products.

Table of Contents

NOTES:

1.0 INTRODUCTION

1.1 General Terms

Byte	8 bits
CSR	Control and Status Registers
DWORD	32 bits
FCT	FIFO Controller
FIFO	First In First Out buffer
Frame	In the context of this document, a frame refers to transfers on the Ethernet interface.
FSM	Finite State Machine
GPIO	General Purpose I/O
HSIC	High-Speed Inter-Chip
Host	External system (includes processor, application software, etc.)
Level-Triggered Sticky Bit	This type of status bit is set whenever the condition that it represents is asserted. The bit remains set until the condition is no longer true and the status bit is cleared by writing a zero.
LFSR	Linear Feedback Shift Register
MAC	Media Access Controller
MII	Media Independent Interface
N/A	Not Applicable
Packet	In the context of this document, a packet refers to transfers on the USB interface.
POR	Power on Reset
RESERVED	Refers to a reserved bit field or address. Unless otherwise noted, reserved bits must always be zero for write operations. Unless otherwise noted, values are not ensured when reading reserved bits. Unless otherwise noted, do not read or write to reserved addresses.
SCSR	System Control and Status Register
SMI	Serial Management Interface
TLI	Transaction Layer Interface
URX	USB Bulk-Out Packet Receiver
UTX	USB Bulk-In Packet Transmitter
WORD	16 bits
ZLP	Zero Length USB Packet

1.2 Block Diagram

FIGURE 1-1: BLOCK DIAGRAM

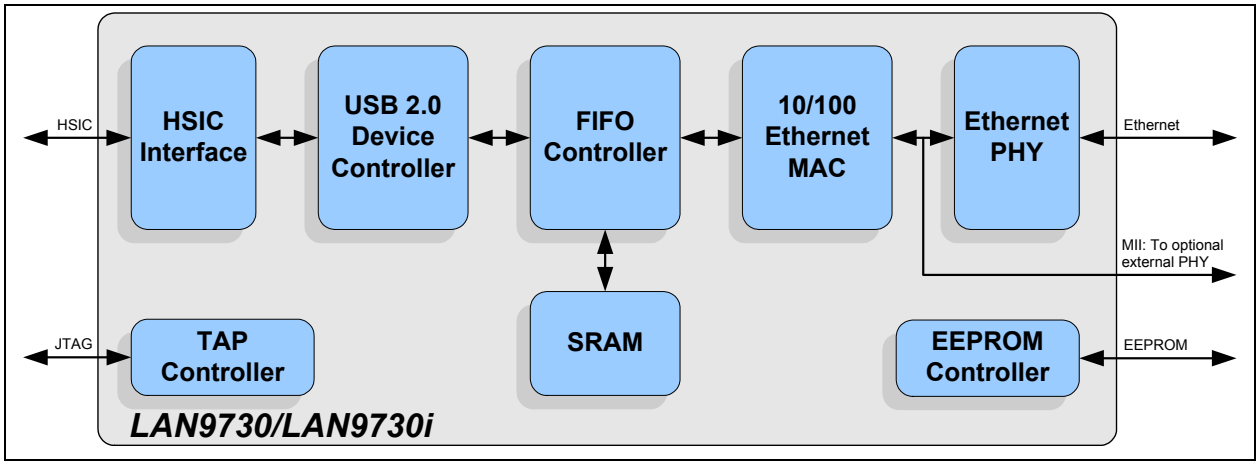
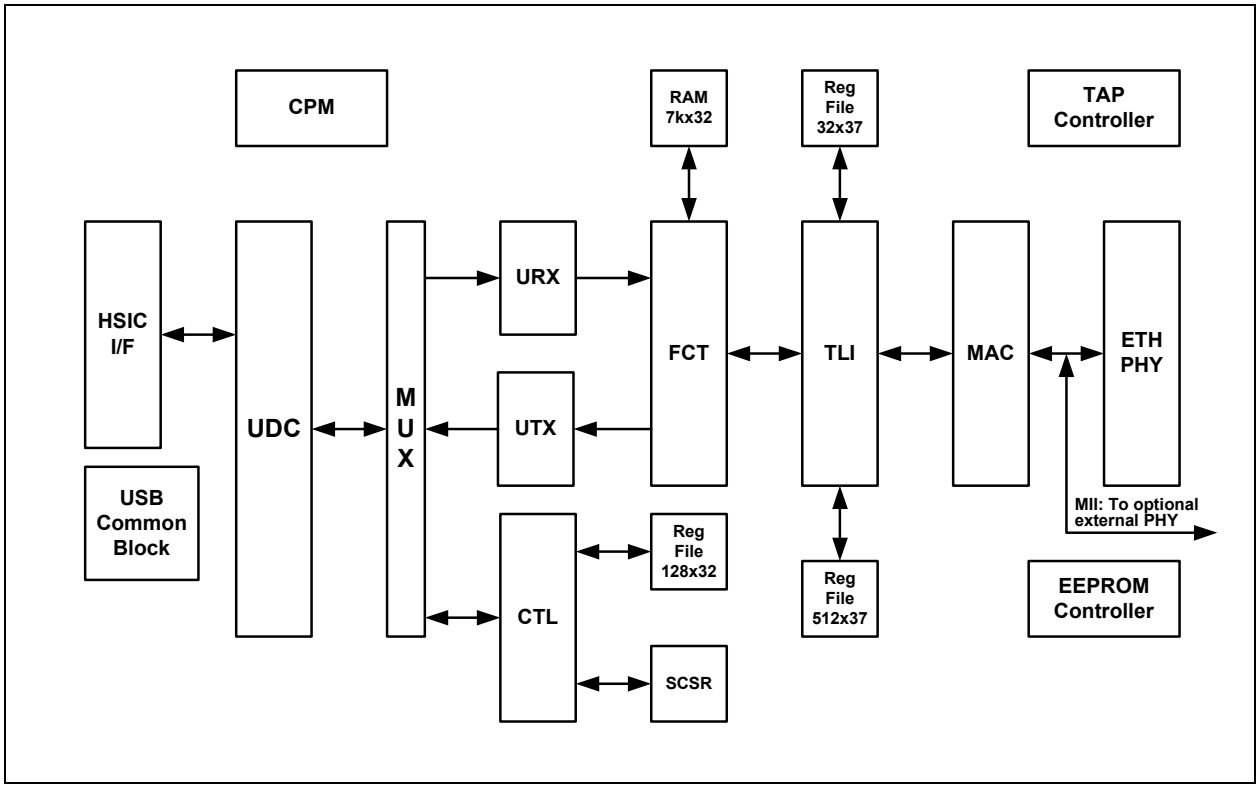


FIGURE 1-2: SYSTEM DIAGRAM



1.2.1 OVERVIEW

The LAN9730/LAN9730i is a high performance solution for USB to 10/100 Ethernet port bridging. With applications ranging from embedded systems, set-top boxes, and PVRs, to USB port replicators, and test instrumentation, the device is targeted as a high-performance, low-cost USB/Ethernet connectivity solution.

The LAN9730/LAN9730i contains an integrated 10/100 Ethernet PHY, HSIC interface, Hi-Speed USB 2.0 device controller, 10/100 Ethernet MAC, TAP controller, EEPROM controller, and a FIFO controller with a total of 30 kB of internal packet buffering. Two kB of buffer memory are allocated to the Transaction Layer Interface (TLI), while 28 kB are allocated to the FIFO Controller (FCT).

The internal USB 2.0 device controller is compliant with the USB 2.0 Hi-Speed standard. The HSIC interface is compliant with the High-Speed Interchip USB Electrical Specification Revision 1.0. High-Speed Inter-Chip (HSIC) is a digital interconnect bus that enables the use of USB technology as a low-power chip-to-chip interconnect at speeds up to 480 Mb/s. The device implements Control, Interrupt, Bulk-In and Bulk-Out USB Endpoints.

The Ethernet controller supports auto-negotiation, auto-polarity correction, HP Auto-MDIX, and is compliant with the IEEE 802.3 and 802.3u standards. An external MII interface provides support for an external Fast Ethernet PHY, HomePNA, and HomePlug functionality.

Multiple power management features are provided, including various low-power modes, and Magic Packet, Wake On LAN and Link Status Change wake events. These wake events can be programmed to initiate a USB remote wakeup. A PCI-like PME wake is also supported when the host controller is disabled.

An internal EEPROM controller exists to load various USB configuration information and the device MAC address. The integrated IEEE 1149.1 compliant TAP controller provides boundary scan via JTAG.

1.2.2 USB

The USB portion of the LAN9730/LAN9730i consists of the USB Device Controller (UDC), USB Bulk-Out Packet Receiver (URX), USB Bulk-In Packet Transmitter (UTX), Control Block (CTL), System Control and Status Registers (SCSR), and HSIC interface.

The USB device controller (UDC) contains a USB low-level protocol interpreter which implements the USB bus protocol, packet generation/extraction, PID/Device ID parsing, and CRC coding/decoding with autonomous error handling. It has autonomous protocol handling functions such as stall condition clearing on setup packets, suspend/resume/reset conditions, and remote wakeup. It also autonomously handles contingency operations for error conditions such as retry for CRC errors, Data toggle errors, and generation of NYET, STALL, ACK, and NACK depending on the Endpoint buffer status. The UDC implements four USB Endpoints: Control, Interrupt, Bulk-In, and Bulk-Out.

The Control block (CTL) manages traffic to/from the control Endpoint that is not handled by the UDC and constructs the packets used by the interrupt Endpoint. The CTL is responsible for handling some USB standard commands and all vendor specific commands. The vendor specific commands allow for efficient statistics collection and access to the SCSR.

The URX and UTX implement the Bulk-Out and Bulk-In pipes, respectively, which connect the USB host and the UDC. They perform the following functions:

The URX passes USB Bulk-Out packets to the FIFO Controller (FCT). It tracks whether or not a USB packet is erroneous. It instructs the FCT to flush erroneous packets by rewinding its write pointer.

The UTX retrieves Ethernet frames from the FCT and constructs USB Bulk-In packets from them. If the handshake for a transmitted Bulk-In packet does not complete, the UTX is capable of retransmitting the packet. The UTX will not instruct the FCT to advance its read head pointer until the current USB packet has been successfully transmitted to the USB host.

Both the URX and UTX are responsible for handling Ethernet frames encapsulated over USB by one of the following methods:

- Multiple Ethernet frames per USB Bulk packet
- Single Ethernet frame per USB Bulk packet

The UDC also implements the System Control and Status Register (SCSR) space used by the host to obtain status and control overall system operation.

The integrated HSIC interface is compliant with the High-Speed Interchip USB Electrical Specification Revision 1.0 (09-23-07) and supports the Hi-Speed mode of operation.

1.2.3 FIFO CONTROLLER (FCT)

The FIFO controller uses a 28 kB internal SRAM to buffer RX and TX traffic. 20 kB are allocated for received Ethernet-USB traffic (RX buffer), while 8 kB are allocated for USB-Ethernet traffic (TX buffer). Bulk-Out packets from the USB controller are directly stored into the TX buffer. The FCT is responsible for extracting Ethernet frames from the USB packet data and passing the frames to the MAC. Ethernet frames are directly stored into the RX buffer and become the basis for Bulk-In packets. The FCT passes the stored data to the UTX in blocks typically 512 bytes in size.

1.2.4 ETHERNET

LAN9730/LAN9730i integrates an IEEE 802.3 PHY for twisted pair Ethernet applications and a 10/100 Ethernet Media Access Controller (MAC).

The PHY can be configured for either 100 Mbps (100BASE-TX) or 10 Mbps (10BASE-T) Ethernet operation in either full- or half-duplex configurations. The PHY block includes auto-negotiation, auto-polarity correction, and Auto-MDIX. Minimal external components are required for the utilization of the integrated PHY.

Optionally, an external PHY may be used via the MII (Media Independent Interface) port, effectively bypassing the internal PHY. This option allows support for HomePNA and HomePlug applications.

The transmit and receive data paths within the 10/100 Ethernet MAC are independent, allowing for the highest performance possible, particularly in full-duplex mode. The Ethernet MAC operates in store and forward mode, utilizing an independent 2 kB buffer for transmitted frames, and a smaller 128 byte buffer for received frames. The Ethernet MAC data paths connect to the FIFO controller. The MAC also implements a Control and Status Register (CSR) space used by the host to obtain status and control its operation.

The Ethernet MAC/PHY supports numerous power management wakeup features, including Magic Packet, Wake on LAN, and Link Status Change. Eight Wakeup Frame Filters are provided by the device.

1.2.5 POWER MANAGEMENT

The LAN9730/LAN9730i features four variations of USB suspend: SUSPEND0, SUSPEND1, SUSPEND2, and SUSPEND3. These modes allow the application to select the ideal balance of remote wakeup functionality and power consumption.

- **SUSPEND0:** Supports GPIO, Wake On LAN and Magic Packet events. This state reduces power by stopping the clocks of the MAC and other internal modules.
- **SUSPEND1:** Supports GPIO and Link Status Change for remote wakeup events. This suspend state consumes less power than SUSPEND0.
- **SUSPEND2:** Supports only GPIO assertion for a remote wakeup event. This is the default suspend mode for the device.
- **SUSPEND3:** Supports GPIO and Good Packet events. A Good Packet is a received frame passing certain filtering constraints independent of those imposed on Wake On LAN and Magic Packet frames. This SUSPEND state consumes power at a level similar to the full operational state, however, it allows for power savings in the host CPU.

Refer to Section 4.12, "Wake Events" for more information on the USB suspend states and the wake events supported in each state.

1.2.6 EEPROM CONTROLLER (EPC)

LAN9730/LAN9730i contains an EEPROM controller for connection to an external EEPROM. This allows for the automatic loading of static configuration data upon Power on Reset, pin reset or software reset. The EEPROM can be configured to load USB descriptors, USB device configuration, and MAC address.

1.2.7 GENERAL PURPOSE I/O

When configured for Internal PHY mode, up to eleven GPIOs are supported. All GPIOs can serve as remote wakeup events when the LAN9730/LAN9730i is suspended.

1.2.8 TAP CONTROLLER

IEEE 1149.1 compliant TAP Controller supports boundary scan and various test modes.

The device includes an integrated JTAG boundary-scan test port for board-level testing. The interface consists of five pins (TDO, TDI, TCK, TMS, and nTRST) and includes a state machine, data register array and an instruction register. The JTAG pins are described in Table 2-3, “JTAG Pins”. The JTAG interface conforms to the IEEE Standard 1149.1 - *1990 Standard Test Access Port (TAP) and Boundary-Scan Architecture*.

All input and output data is synchronous to the TCK test clock input. TAP input signals TMS and TDI are clocked into the test logic on the rising edge of TCK, while the output signal TDO is clocked on the falling edge.

The JTAG logic is reset via Power on Reset (POR) or when the nTRST pin is asserted active-low.

The implemented IEEE 1149.1 instructions and their op codes are shown in Table 1-1.

TABLE 1-1: IEEE 1149.1 OP CODES

Instruction	Op Code	Comment
Bypass	111111b	Mandatory Instruction
Sample/Preload	000100b	Mandatory Instruction
EXTEST	000001b	Mandatory Instruction
HIGHZ	000011b	Optional Instruction
IDCODE	001010b	Optional Instruction

Note: The JTAG device ID is 00091445h.

Note: All digital I/O pins support IEEE 1149.1 operation. Analog pins and the XI/XO pins do not support IEEE 1149.1 operation.

1.2.9 CONTROL AND STATUS REGISTERS (CSR)

LAN9730/LAN9730i’s functions are controlled and monitored by the host via the Control and Status Registers (CSRs). This register space includes registers that control and monitor the USB controller, as well as elements of overall system operation (System Control and Status Registers - SCSRs), the MAC (MAC Control and Status Registers - MCSRs), and the PHY (accessed indirectly through the MAC via the MII_ACCESS and MII_DATA registers). The CSR may be accessed via the USB Vendor Commands (REGISTER READ/REGISTER WRITE). Refer to Section 4.3.3, "USB Vendor Commands" for more information.

1.2.10 RESETS

LAN9730/LAN9730i supports the following system reset events:

- Power on Reset (POR)
- Hardware Reset Input Pin Reset (nRESET)
- Lite Reset (LRST) (Does not affect the UDC)
- Software Reset (SRST)
- USB Reset

The device supports the following module level reset events:

- Ethernet PHY Software Reset (PHY_RST)
- nTRST Pin Reset for Tap Controller

1.2.11 TEST FEATURES

Read/write access to internal SRAMs is provided via the CSRs. JTAG-based USB BIST is available. Full internal scan and At Speed scan are supported.

1.2.12 SYSTEM SOFTWARE

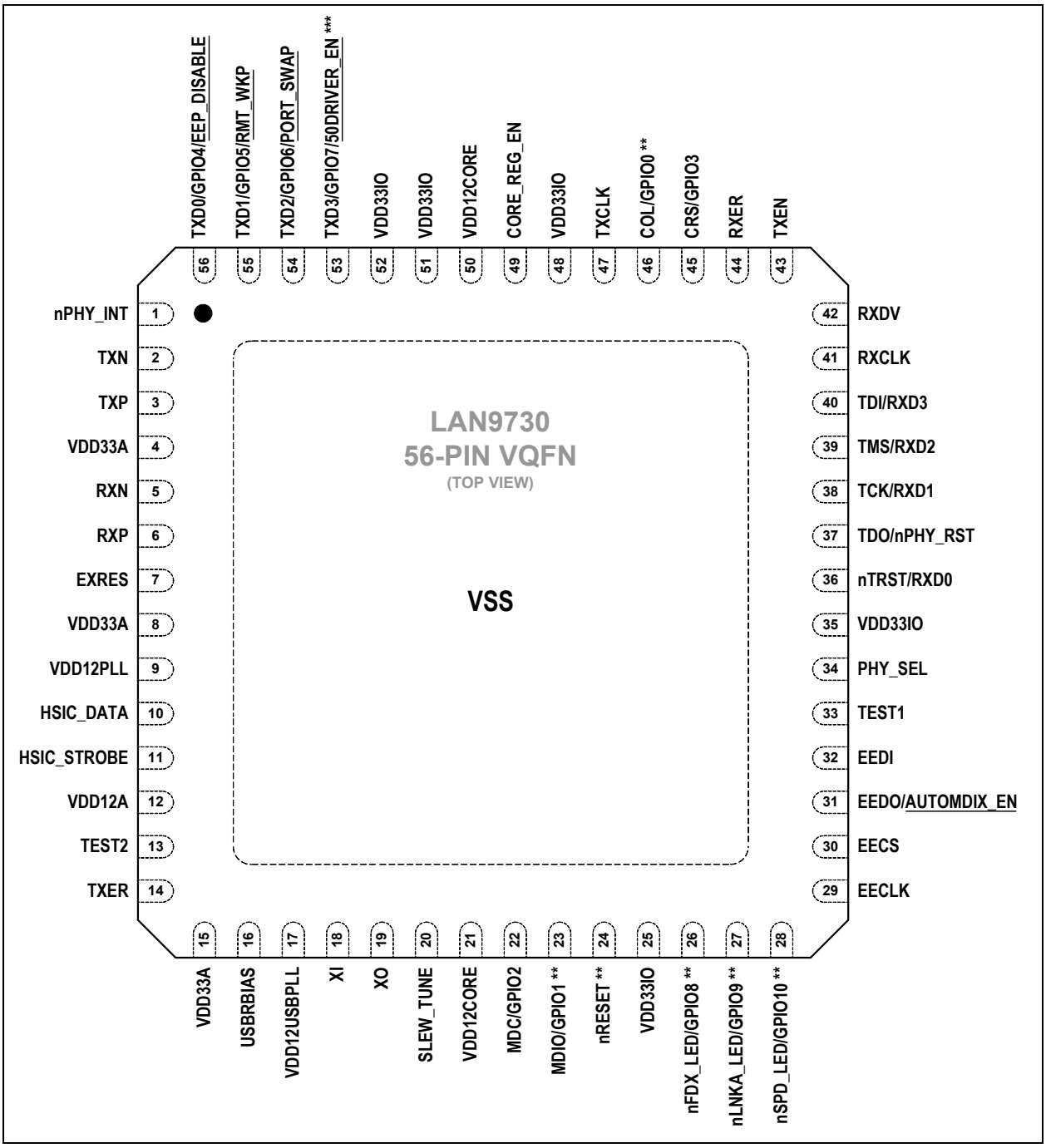
LAN9730/LAN9730i software drivers are available for the following operating systems:

- Windows 8
- Windows 7
- Windows Vista
- Windows XP
- Linux
- Win CE
- MAC OS

In addition, an EEPROM programming utility is available for configuring the external EEPROM.

2.0 PIN DESCRIPTION AND CONFIGURATION

FIGURE 2-1: PIN ASSIGNMENTS (TOP VIEW)



Note: ** This pin provides additional PME related functionality. Refer to the respective pin descriptions and Chapter 5.0, "PME Operation" for additional information.

Note: *** GPIO7 may provide additional PHY Link Up related functionality. Refer to Section 4.12.2.4, "Enabling External PHY Link Up Wake Events" for additional information.

Note: When HP Auto-MDIX is activated, the TXN/TXP pins can function as RXN/RXP and vice-versa.

Note: Exposed pad (VSS) on bottom of package must be connected to ground.

TABLE 2-1: MII INTERFACE PINS

Num Pins	Name	Symbol	Buffer Type	Description
1	Receive Error (Internal PHY Mode)	RXER	IS/O8 (PD)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Receive Error (External PHY Mode)	RXER	IS (PD)	In External PHY Mode, the signal on this pin is input from the external PHY and indicates a receive error in the packet.
1	Transmit Error (Internal PHY Mode)	TXER	IS/O8 (PD)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Transmit Error (External PHY Mode)	TXER	O8 (PD)	In External PHY Mode, this pin functions as an output to the external PHY and indicates a transmit error.
1	Transmit Enable (Internal PHY Mode)	TXEN	IS/O8 (PD)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Transmit Enable (External PHY Mode)	TXEN	O8 (PD)	In External PHY Mode, this pin functions as an output to the external PHY and indicates valid data on TXD[3:0].
1	Receive Data Valid (Internal PHY Mode)	RXDV	IS/O8 (PD)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Receive Data Valid (External PHY Mode)	RXDV	IS (PD)	In External PHY Mode, the signal on this pin is input from the external PHY and indicates valid data on RXD[3:0].
1	Receive Clock (Internal PHY Mode)	RXCLK	IS/O8 (PD)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Receive Clock (External PHY Mode)	RXCLK	IS (PD)	In External PHY Mode, this pin is the receiver clock input from the external PHY.

TABLE 2-1: MII INTERFACE PINS (CONTINUED)

Num Pins	Name	Symbol	Buffer Type	Description
1	Transmit Clock (Internal PHY Mode)	TXCLK	IS/O8 (PU)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Transmit Clock (External PHY Mode)	TXCLK	IS (PU)	In External PHY Mode, this pin is the transmitter clock input from the external PHY.
1	Carrier Sense (Internal PHY Mode)	CRS	IS/O8 (PU)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Carrier Sense (External PHY Mode)	CRS	IS (PD)	In External PHY Mode, the signal on this pin is input from the external PHY and indicates a network carrier.
	General Purpose I/O 3 (Internal PHY Mode Only)	GPIO3	IS/O8/ OD8 (PU)	This General Purpose I/O pin is fully programmable as either a push-pull output, an open-drain output or a Schmitt-triggered input.
1	MII Collision Detect (Internal PHY Mode)	COL	IS/O8 (PU)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	MII Collision Detect (External PHY Mode)	COL	IS (PD)	In External PHY Mode, the signal on this pin is input from the external PHY and indicates a collision event.
	General Purpose I/O 0 (Internal PHY Mode Only)	GPIO0	IS/O8/ OD8 (PU)	This General Purpose I/O pin is fully programmable as either a push-pull output, an open-drain output or a Schmitt-triggered input. **Note:** This pin may be used to signal PME when Internal PHY and PME Modes of operation are in effect. Refer to Chapter 5.0, "PME Operation" for additional information.

TABLE 2-1: MII INTERFACE PINS (CONTINUED)

Num Pins	Name	Symbol	Buffer Type	Description
1	Management Data (Internal PHY Mode)	MDIO	IS/O8 (PU)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Management Data (External PHY Mode)	MDIO	IS/O8 (PD)	In External PHY Mode, this pin provides the management data to/from the external PHY.
	General Purpose I/O 1 (Internal PHY Mode Only)	GPIO1	IS/O8/ OD8 (PU)	This General Purpose I/O pin is fully programmable as either a push-pull output, an open-drain output or a Schmitt-triggered input. **Note:** This pin may serve as the PME_MODE_SEL input when Internal PHY and PME Modes of operation are in effect. Refer to Chapter 5.0, "PME Operation" for additional information.
1	Management Clock (Internal PHY Mode)	MDC	IS/O8 (PU)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Management Clock (External PHY Mode)	MDC	O8 (PD)	In External PHY Mode, this pin outputs the management clock to the external PHY.
	General Purpose I/O 2 (Internal PHY Mode Only)	GPIO2	IS/O8/ OD8 (PU)	This General Purpose I/O pin is fully programmable as either a push-pull output, an open-drain output or a Schmitt-triggered input.

TABLE 2-1: MII INTERFACE PINS (CONTINUED)

Num Pins	Name	Symbol	Buffer Type	Description
1	Transmit Data 3 (Internal PHY Mode)	TXD3	IS/O8 (PU)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Transmit Data 3 (External PHY Mode)	TXD3	O8 (PU)	In External PHY Mode, this pin functions as the transmit data 3 output to the external PHY.
	General Purpose I/O 7 (Internal PHY Mode Only)	GPIO7	IS/O8/ OD8 (PU)	This General Purpose I/O pin is fully programmable as either a push-pull output, an open-drain output or a Schmitt-triggered input. **Note:** GPIO7 may provide additional external PHY Link Up related functionality. Refer to Section 4.12.2.4, "Enabling External PHY Link Up Wake Events" for additional information.
	HSIC Output Impedance Configuration Strap	50DRIVER_EN	IS (PU)	The 50DRIVER_EN strap selects the driver output impedance for the HSIC_DATA and HSIC_STROBE pins. 0 = 40 Ω output impedance 1 = 50 Ω output impedance See Note 2-1 for more information on configuration straps.

TABLE 2-1: MII INTERFACE PINS (CONTINUED)

Num Pins	Name	Symbol	Buffer Type	Description
1	Transmit Data 2 (Internal PHY Mode)	TXD2	IS/O8 (PD)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Transmit Data 2 (External PHY Mode)	TXD2	O8 (PD)	In External PHY Mode, this pin functions as the transmit data 2 output to the external PHY.
	General Purpose I/O 6 (Internal PHY Mode Only)	GPIO6	IS/O8/ OD8 (PU)	This General Purpose I/O pin is fully programmable as either a push-pull output, an open-drain output, or a Schmitt-triggered input.
	HSIC Port Swap Configuration Strap	PORT_SWAP	IS (PD)	Swaps the mapping of HSIC_DATA and HSIC_STROBE. 0 = The HSIC_DATA and HSIC_STROBE pin functionality is not swapped. 1 = The HSIC_DATA and HSIC_STROBE pin functionality is swapped. See Note 2-1 for more information on configuration straps.
1	Transmit Data 1 (Internal PHY Mode)	TXD1	IS/O8 (PD)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Transmit Data 1 (External PHY Mode)	TXD1	O8 (PD)	In External PHY Mode, this pin functions as the transmit data 1 output to the external PHY.
	General Purpose I/O 5 (Internal PHY Mode Only)	GPIO5	IS/O8/ OD8 (PU)	This General Purpose I/O pin is fully programmable as either a push-pull output, an open-drain output or a Schmitt-triggered input.
	Remote Wakeup Configuration Strap	RMT_WKP	IS (PD)	This strap configures the default descriptor values to support remote wakeup. This strap is overridden by the EEPROM. 0 = Remote wakeup is not supported. 1 = Remote wakeup is supported. See Note 2-1 for more information on configuration straps.

TABLE 2-1: MII INTERFACE PINS (CONTINUED)

Num Pins	Name	Symbol	Buffer Type	Description
1	Transmit Data 0 (Internal PHY Mode)	TXD0	IS/O8 (PD)	In Internal PHY Mode, this pin can be configured to display the respective internal MII signal. Refer to the Internal MII Visibility Enable (IME) bit of the Hardware Configuration Register (HW_CFG) on page 125 for additional information.
	Transmit Data 0 (External PHY Mode)	TXD0	O8 (PD)	In External PHY Mode, this pin functions as the transmit data 0 output to the external PHY.
	General Purpose I/O 4 (Internal PHY Mode Only)	GPIO4	IS/O8/ OD8 (PU)	This General Purpose I/O pin is fully programmable as either a push-pull output, an open-drain output or a Schmitt-triggered input.
	EEPROM Disable Configuration Strap	EEP_DISABLE	IS (PD)	This strap disables the autoloading of the EEPROM contents. The assertion of this strap does not prevent register access to the EEPROM. 0 = EEPROM is recognized if present. 1 = EEPROM is not recognized even if it is present. See Note 2-1 for more information on configuration straps.

TABLE 2-2: EEPROM PINS

Num Pins	Name	Symbol	Buffer Type	Description
1	EEPROM Data In	EEDI	IS (PD)	This pin is driven by the EEDO output of the external EEPROM.
1	EEPROM Data Out	EEDO	O8 (PU)	This pin drives the EEDI input of the external EEPROM.
	Auto-MDIX Enable Configuration Strap	AUTOMDIX_EN	IS (PU)	Determines the default Auto-MDIX setting. 0 = Auto-MDIX is disabled. 1 = Auto-MDIX is enabled. See Note 2-1 for more information on configuration straps.
1	EEPROM Chip Select	EECS	O8	This pin drives the chip select output of the external EEPROM. **Note:** The EECS output may tri-state briefly during power-up. Some EEPROM devices may be prone to false selection during this time. When an EEPROM is used, an external pull-down resistor is recommended on this signal to prevent false selection. Refer to your EEPROM manufacturer's datasheet for additional information.
1	EEPROM Clock	EECLK	O8 (PD)	This pin drives the EEPROM clock of the external EEPROM. **Note:** This pin must be pulled-up externally for proper operation.

Note 2-1 Configuration strap values are latched on Power on Reset (POR) or External Chip Reset (nRESET). Configuration straps are identified by an underlined symbol name. Pins that function as configuration straps must be augmented with an external resistor when connected to a load. Refer to Section 4.14, "Configuration Straps" for additional information.

TABLE 2-3: JTAG PINS

Num Pins	Name	Symbol	Buffer Type	Description
1	JTAG Test Port Reset (Internal PHY Mode)	nTRST	IS (PU)	In Internal PHY Mode, this active-low pin functions as the JTAG test port reset input.
	Receive Data 0 (External PHY Mode)	RXD0	IS (PD)	In External PHY Mode, this pin functions as the receive data 0 input from the external PHY.
1	JTAG Test Data Out (Internal PHY Mode)	TDO	O8	In Internal PHY Mode, this pin functions as the JTAG data output.
	PHY Reset (External PHY Mode)	nPHY_RST	O8	In External PHY Mode, this active-low pin functions as the PHY reset output.
1	JTAG Test Clock (Internal PHY Mode)	TCK	IS (PU)	In Internal PHY Mode, this pin functions as the JTAG test clock. The maximum operating frequency of this clock is 25 MHz.
	Receive Data 1 (External PHY Mode)	RXD1	IS (PD)	In External PHY Mode, this pin functions as the receive data 1 input from the external PHY.
1	JTAG Test Mode Select (Internal PHY Mode)	TMS	IS (PU)	In Internal PHY Mode, this pin functions as the JTAG test mode select.
	Receive Data 2 (External PHY Mode)	RXD2	IS (PD)	In External PHY Mode, this pin functions as the receive data 2 input from the external PHY.
1	JTAG Test Data Input (Internal PHY Mode)	TDI	IS (PU)	In Internal PHY Mode, this pin functions as the JTAG data input.
	Receive Data 3 (External PHY Mode)	RXD3	IS (PD)	In External PHY Mode, this pin functions as the receive data 3 input from the external PHY.

TABLE 2-4: MISCELLANEOUS PINS

Num Pins	Name	Symbol	Buffer Type	Description
1	PHY Select	PHY_SEL	IS (PD)	Selects whether to use the internal Ethernet PHY or the external PHY connected to the MII port. 0 = Internal PHY is used. 1 = External PHY is used. **Note:** When in External PHY Mode, the internal PHY is placed into general power down after a POR. Refer to Section 4.6, "10/100 Internal Ethernet PHY" for details.
1	System Reset	nRESET	IS (PU)	This active-low pin allows external hardware to reset the device. **Note:** This pin may be used to signal PME_CLEAR when PME Mode of operation is in effect. Refer to Chapter 5.0, "PME Operation" for additional information.
1	Ethernet Full-Duplex Indicator LED	nFDX_LED	OD12 (PU)	This pin is driven low (LED on) when the Ethernet link is operating in Full-Duplex mode.
	General Purpose I/O 8	GPIO8	IS/O12/ OD12 (PU)	This General Purpose I/O pin is fully programmable as either a push-pull output, an open-drain output or a Schmitt-triggered input. **Note:** This pin may be used to signal PME when External PHY and PME Modes of operation are in effect. Refer to Chapter 5.0, "PME Operation" for additional information. **Note:** By default this pin is configured as a GPIO.

TABLE 2-4: MISCELLANEOUS PINS (CONTINUED)

Num Pins	Name	Symbol	Buffer Type	Description
1	Ethernet Link Activity Indicator LED	nLNKA_LED	OD12 (PU)	This pin is driven low (LED on) when a valid link is detected. This pin is pulsed high (LED off) for 80 ms whenever transmit or receive activity is detected. This pin is then driven low again for a minimum of 80 ms, after which time it will repeat the process if TX or RX activity is detected. Effectively, LED2 is activated solid for a link. When transmit or receive activity is sensed, LED2 will function as an activity indicator.
	General Purpose I/O 9	GPIO9	IS/O12/ OD12 (PU)	This General Purpose I/O pin is fully programmable as either a push-pull output, an open-drain output or a Schmitt-triggered input. **Note:** This pin may serve as the PME_MODE_SEL input when External PHY and PME Modes of operation are in effect. Refer to Chapter 5.0, "PME Operation" for additional information. **Note:** By default this pin is configured as a GPIO.
1	Ethernet Speed Indicator LED	nSPD_LED	OD12 (PU)	This pin is driven low (LED on) when the Ethernet operating speed is 100 Mbs, or during auto-negotiation. This pin is driven high during 10 Mbs operation or during line isolation.
	General Purpose I/O 10	GPIO10	IS/O12/ OD12 (PU)	This General Purpose I/O pin is fully programmable as either a push-pull output, an open-drain output or a Schmitt-triggered input. **Note:** This pin may serve as a wakeup pin whose detection mode is selectable when External PHY and PME Modes of operation are in effect. Refer to Chapter 5.0, "PME Operation" for additional information. **Note:** By default this pin is configured as a GPIO.
1	Core Regulator Enable	CORE_REG_EN	AI	This pin enables/disables the internal core logic voltage regulator. When tied low to VSS, the internal core regulator is disabled and +1.2 V must be supplied to the device by an external source. When tied high to +3.3 V, the internal core regulator is enabled. Refer to Chapter 3.0, "Power Connections" and the device reference schematics for connection information.
1	Test 1	TEST1	-	This pin must always be connected to VSS for proper operation.

TABLE 2-4: MISCELLANEOUS PINS (CONTINUED)

Num Pins	Name	Symbol	Buffer Type	Description
1	Test 2	TEST2	-	This pin must always be connected to +3.3 V for proper operation.
1	Crystal Input	XI	ICLK	External 25 MHz crystal input. **Note:** This pin can also be driven by a single-ended clock oscillator. When this method is used, XO should be left unconnected.
1	Crystal Output	XO	OCLK	External 25 MHz crystal output.

TABLE 2-5: USB PINS

Num Pins	Name	Symbol	Buffer Type	Description
1	HSIC Data	HSIC_DATA	HSIC	Bi-directional Double Data Rate (DDR) data signal that is synchronous to the HSIC_STROBE signal as defined in the *High-Speed Inter-Chip USB Electrical Specification, Version 1.0*.
1	HSIC Strobe	HSIC_STROBE	HSIC	Bi-directional data strobe signal as defined in the *High-Speed Inter-Chip USB Electrical Specification, Version 1.0*.
1	HSIC Slew Tune	SLEW_TUNE	IS (PD)	Applies a 30% slew rate boost to the HSIC_DATA and HSIC_STROBE pins when driven high.
1	External USB Bias Resistor	USBRBIAS	AI	Used for setting HS transmit current level and on-chip termination impedance. Connect to an external 12.0 kΩ 1.0% resistor to ground.

TABLE 2-6: ETHERNET PHY PINS

Num Pins	Name	Symbol	Buffer Type	Description
1	Ethernet TX Data Out Negative	TXN	AIO	The transmit data outputs may be swapped internally with receive data inputs when Auto-MDIX is enabled.
1	Ethernet TX Data Out Positive	TXP	AIO	The transmit data outputs may be swapped internally with receive data inputs when Auto-MDIX is enabled.
1	Ethernet RX Data In Negative	RXN	AIO	The receive data inputs may be swapped internally with transmit data outputs when Auto-MDIX is enabled.
1	Ethernet RX Data In Positive	RXP	AIO	The receive data inputs may be swapped internally with transmit data outputs when Auto-MDIX is enabled.
1	PHY Interrupt (Internal PHY Mode)	nPHY_INT	O8	In Internal PHY Mode, this pin can be configured to output the internal PHY interrupt signal. **Note:** The internal PHY interrupt signal is active-high.
	PHY Interrupt (External PHY Mode)	nPHY_INT	IS (PU)	In External PHY Mode, the active-low signal on this pin is input from the external PHY and indicates a PHY interrupt has occurred.
1	External PHY Bias Resistor	EXRES	AI	Used for the internal bias circuits. Connect to an external 12.0 kΩ 1.0% resistor to ground.

TABLE 2-7: POWER PINS AND GROUND PAD

Num Pins	Name	Symbol	Buffer Type	Description
5	+3.3 V I/O Power	VDD33IO	P	Refer to Chapter 3.0, "Power Connections" and the device reference schematics for connection information.
3	+3.3 V Analog Power	VDD33A	P	Refer to Chapter 3.0, "Power Connections" and the device reference schematics for connection information.
2	+1.2 V Digital Core Power	VDD12CORE	P	Refer to Chapter 3.0, "Power Connections" and the device reference schematics for connection information.
1	+1.2 V USB PLL Power	VDD12USBPLL	P	This pin must be connected to VDD12CORE for proper operation. Refer to Chapter 3.0, "Power Connections" and the device reference schematics for additional connection information.
1	+1.2 V HSIC Power	VDD12A	P	This pin must be connected to VDD12CORE for proper operation. Refer to Chapter 3.0, "Power Connections" and the device reference schematics for connection information.
1	+1.2 V Ethernet PLL Power	VDD12PLL	P	This pin must be connected to VDD12CORE for proper operation. Refer to Chapter 3.0, "Power Connections" and the device reference schematics for additional connection information.
Exposed pad on package bottom (Figure 2-1)	Ground	VSS	P	Common Ground

2.1 Pin Assignments

TABLE 2-8: 56-VQFN PACKAGE PIN ASSIGNMENTS

Pin Num	Pin Name	Pin Num	Pin Name	Pin Num	Pin Name	Pin Num	Pin Name
1	nPHY_INT	15	VDD33A	29	EECLK	43	TXEN
2	TXN	16	USBRBIAS	30	EECS	44	RXER
3	TXP	17	VDD12USBPLL	31	EEDO/ AUTOMDIX_EN	45	CRS/GPIO3
4	VDD33A	18	XI	32	EEDI	46	COL/GPIO0 Note 2-2
5	RXN	19	XO	33	TEST1	47	TXCLK
6	RXP	20	SLEW_TUNE	34	PHY_SEL	48	VDD33IO
7	EXRES	21	VDD12CORE	35	VDD33IO	49	CORE_REG_EN
8	VDD33A	22	MDC/GPIO2	36	nTRST/RXD0	50	VDD12CORE
9	VDD12PLL	23	MDIO/GPIO1 Note 2-2	37	TDO/nPHY_RST	51	VDD33IO
10	HSIC_DATA	24	nRESET Note 2-2	38	TCK/RXD1	52	VDD33IO
11	HSIC_STROBE	25	VDD33IO	39	TMS/RXD2	53	TXD3/GPIO7/ 50DRIVER_EN
12	VDD12A	26	nFDX_LED/ GPIO8	40	TDI/RXD3	54	TXD2/GPIO6/ PORT_SWAP
13	TEST2	27	nLNKA_LED/ GPIO9 Note 2-2	41	RXCLK	55	TXD1/GPIO5/ RMT_WKP
14	TXER	28	nSPD_LED/ GPIO10 Note 2-2	42	RXDV	56	TXD0/GPIO4/ EEP_DISABLE
EXPOSED PAD MUST BE CONNECTED TO VSS							

Note 2-2 This pin provides additional PME-related functionality. Refer to the respective pin descriptions and Chapter 5.0, "PME Operation" for additional information.

2.2 Buffer Types

TABLE 2-9: BUFFER TYPES

Buffer Type	Description
IS	Schmitt-triggered input
O8	Output with 8 mA sink and 8 mA source
OD8	Open-drain output with 8 mA sink
O12	Output with 12 mA sink and 12 mA source
OD12	Open-drain output with 12 mA sink
HSIC	*High-Speed Inter-Chip (HSIC) USB Electrical Specification, Version 1.0* compliant input/output
PU	50 µA (typical) internal pull-up. Unless otherwise noted in the pin description, internal pull-ups are always enabled. **Note:** Internal pull-up resistors prevent unconnected inputs from floating. Do not rely on internal resistors to drive signals external to the device. When connected to a load that must be pulled high, an external resistor must be added.
PD	50 µA (typical) internal pull-down. Unless otherwise noted in the pin description, internal pull-downs are always enabled. **Note:** Internal pull-down resistors prevent unconnected inputs from floating. Do not rely on internal resistors to drive signals external to the device. When connected to a load that must be pulled low, an external resistor must be added.
AI	Analog input
AIO	Analog bi-directional
ICLK	Crystal oscillator input pin
OCLK	Crystal oscillator output pin
P	Power pin

3.0 POWER CONNECTIONS

The LAN9730/LAN9730i can be operated with the internal core regulator enabled or disabled. Figure 3-1 illustrates the power connections for operating the device with the internal regulator enabled. Figure 3-2 illustrates the power connections for operating the device with the internal regulator disabled. In this mode, +1.2 V must be supplied to the device by an external source.

FIGURE 3-1: POWER CONNECTIONS - INTERNAL REGULATOR ENABLED

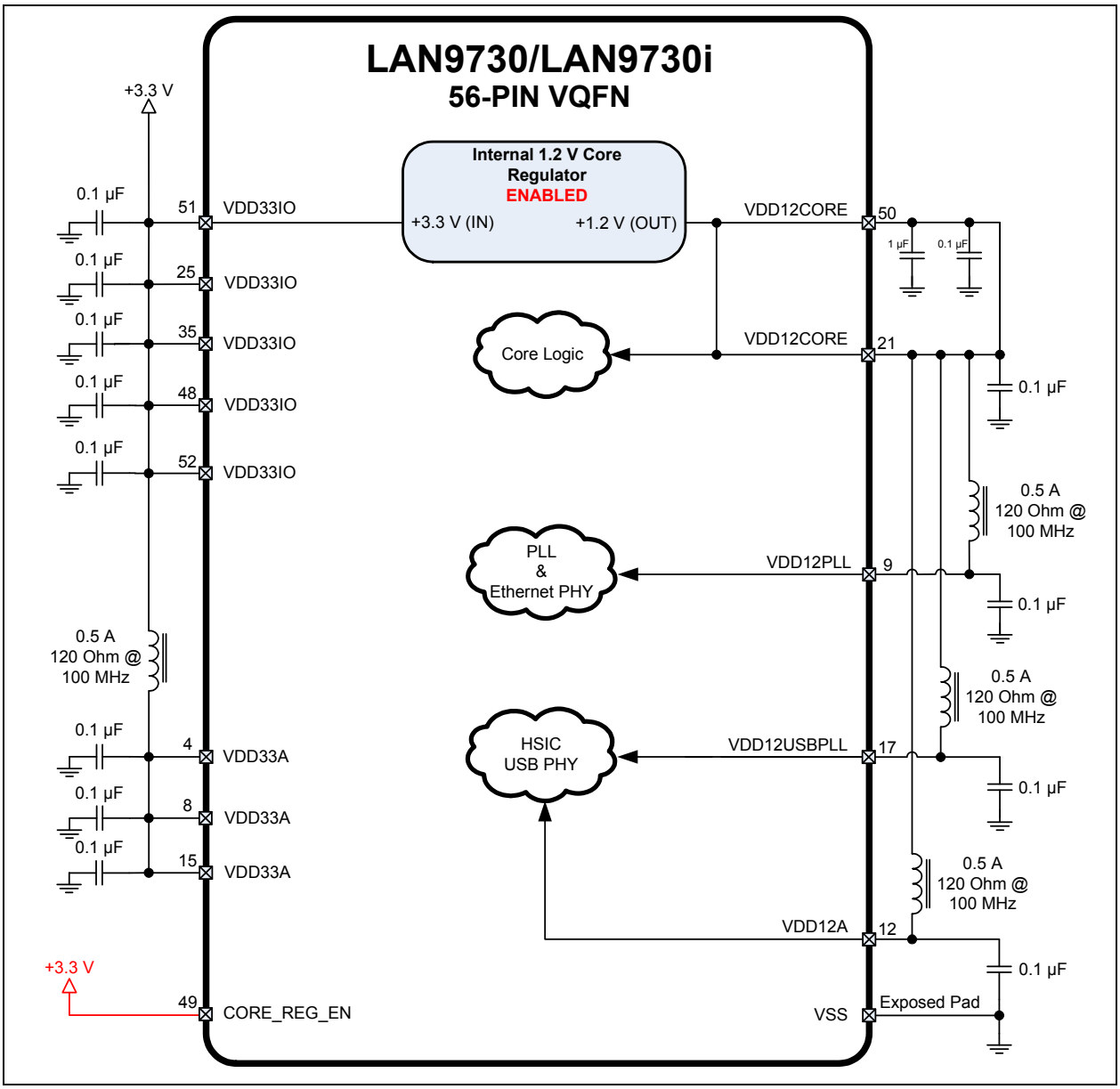
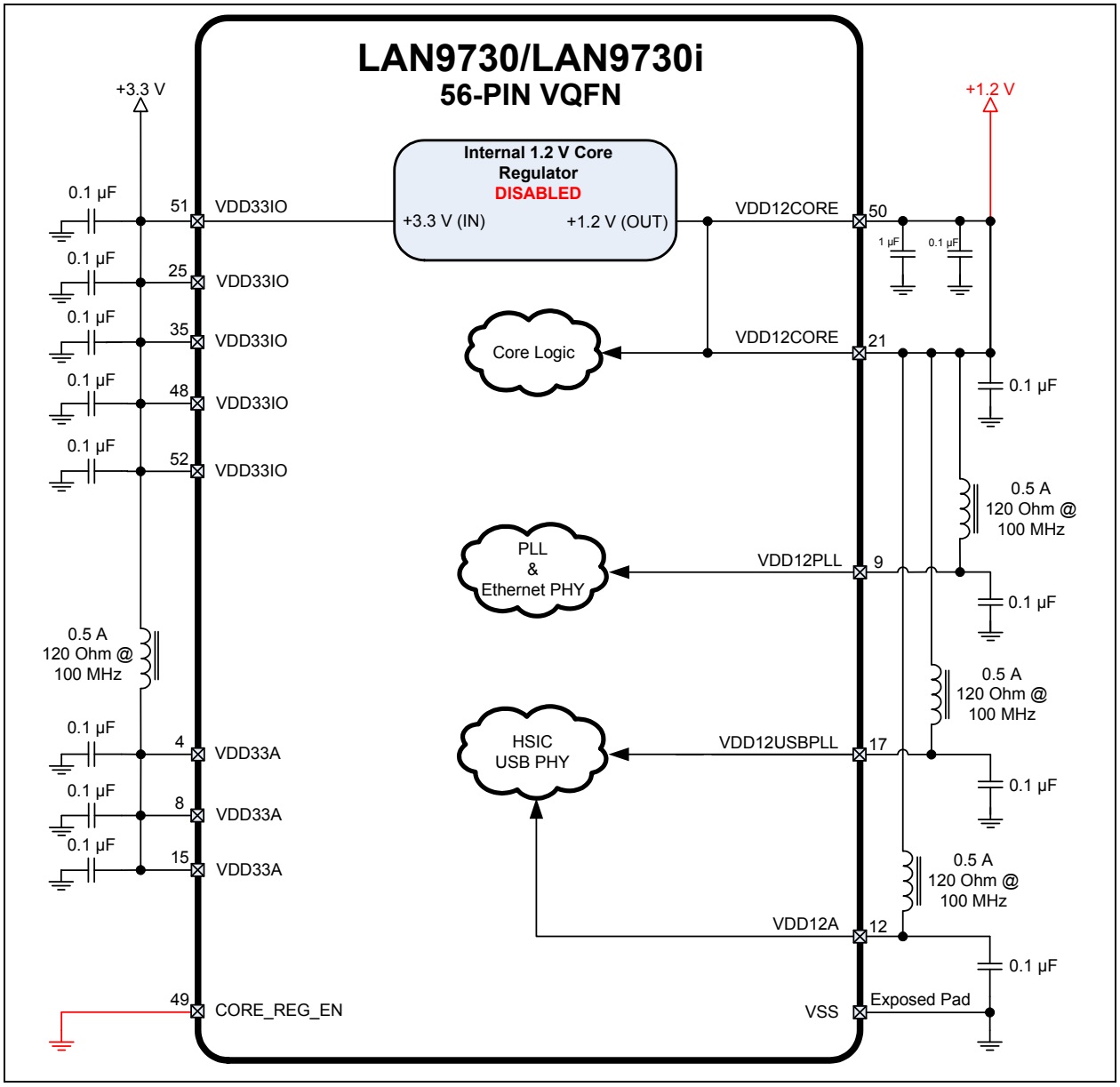


FIGURE 3-2: POWER CONNECTIONS - INTERNAL REGULATOR DISABLED



4.0 FUNCTIONAL DESCRIPTION

4.1 Functional Overview

The LAN9730/LAN9730i USB 2.0 to 10/100 Ethernet Controller consists of the following major functional blocks:

- HSIC Interface
- USB 2.0 Device Controller (UDC)
- FIFO Controller (FCT) and Associated SRAM
- 10/100 Ethernet MAC
- 10/100 Internal Ethernet PHY
- IEEE 1149.1 Tap Controller
- EEPROM Controller (EPC)

The following sections discuss the features of each block. A block diagram of the device is shown in Figure 1-1.

4.2 HSIC Interface

The HSIC interface is compliant with the High-Speed Interchip USB Electrical Specification Revision 1.0. High-Speed Inter-Chip (HSIC) is a digital interconnect bus that enables the use of USB technology as a low-power chip-to-chip interconnect at speeds up to 480 Mb/s.

4.3 USB 2.0 Device Controller (UDC)

The USB functionality in the device consists of five major parts. The HSIC interface (discussed in Section 4.2), UCB (USB Common Block), UDC (USB Device Controller), URX (USB Bulk-Out Receiver), UTX (USB Bulk-In Receiver), and CTL (USB Control Block). They are represented as the HSIC interface and UDC, collectively, in Figure 1-1.

The UCB generates various clocks, including the system clocks of the device. The URX and UTX implement the Bulk-Out and Bulk-In Endpoints respectively. The CTL manages control and interrupt Endpoints.

The UDC is a USB low-level protocol interpreter. The UDC controls the USB bus protocol, packet generation/extraction, PID/Device ID parsing, and CRC coding/decoding with autonomous error handling. It is capable of operating either in USB 1.1 or 2.0 compliant modes. It has autonomous protocol handling functions such as stall condition clearing on setup packets, suspend/resume/reset conditions, and remote wakeup. It also autonomously handles error conditions such as retry for CRC errors, Data toggle errors, and generation of NYET, STALL, ACK and NACK, depending on the Endpoint buffer status.

The UDC is configured to support one configuration, one interface, one alternate setting, and four Endpoints.

4.3.1 SUPPORTED ENDPOINTS

Table 4-1 lists the supported Endpoints. The following subsections discuss these Endpoints in detail.

TABLE 4-1: SUPPORTED ENDPOINTS

Endpoint Number	Description
0	Control Endpoint
1	Bulk-In Endpoint
2	Bulk-Out Endpoint
3	Interrupt Endpoint

The URX and UTX implement the Bulk-Out and Bulk-In Endpoints, respectively. The CTL manages the Control and Interrupt Endpoints.

4.3.1.1 Endpoint 1 (Bulk-In)

The Bulk-In Endpoint is controlled by the UTX (USB Bulk-In Transmitter). The UTX is responsible for encapsulating Ethernet data into a USB Bulk-In packet. Ethernet frames are retrieved from the FCT's RX FIFO.

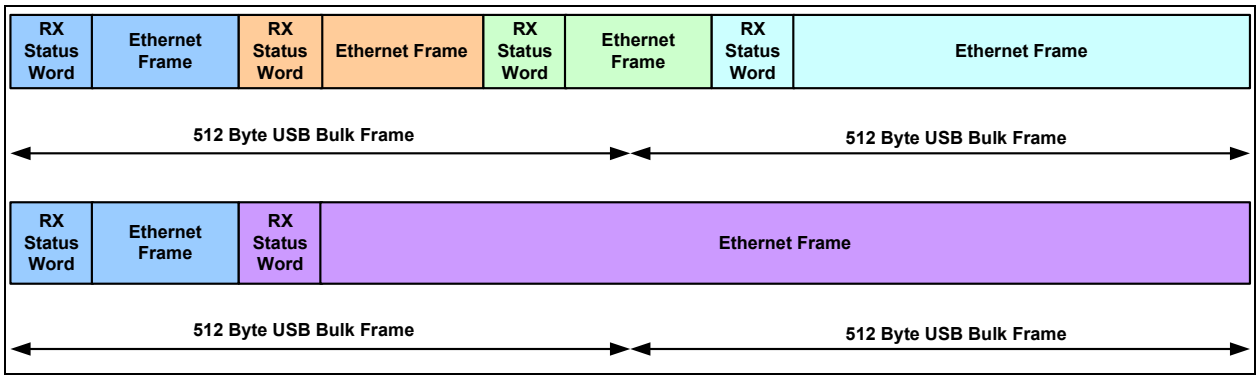
The UTX supports the following two modes of operation: MEF and SEF, selected via the Multiple Ethernet Frames per USB Packet (MEF) bit of the Hardware Configuration Register (HW_CFG).

- MEF: Multiple Ethernet frames per Bulk-In packet. This mode will maximize USB bus utilization by allowing multiple Ethernet frames to be packed into a USB packet. Frames greater than 512 bytes are split across multiple Bulk-In packets.
- SEF: Single Ethernet frame per Bulk-In packet. This mode will not maximize USB bus utilization, but can potentially ease the burden on a low end host processor. Frames greater than 512 bytes are split across multiple Bulk-In packets.

Each Ethernet frame is prepended with an RX Status Word by the FCT. The status word contains the frame length that is used by the UTX to perform the encapsulation functions. The RX Status word is generated by the RX Transaction Layer Interface (RX TLI). The TLI resides between the MAC and the FCT.

Padding may be inserted between the RX Status Word and the Ethernet frame by the FCT. This condition exists when the RXDOFF register has a nonzero value (refer to Hardware Configuration Register (HW_CFG) for details). The padding is implemented by the FCT barrel shifting the Ethernet frame by the specified byte offset.

FIGURE 4-1: MEF USB ENCAPSULATION



In accordance with the USB protocol, the UTX terminates a burst with either a ZLP or a Bulk-In packet with a size of less than the Bulk-In maximum packet size (512 bytes). The ZLP is needed when the total amount of data transmitted is a multiple of a Bulk-In packet size. The UTX monitors the RX FIFO size signal from the FCT to determine when a burst has ended.

Note:	In SEF mode, a ZLP is transmitted if the Ethernet frame is the same size as a Bulk-In packet, or a multiple of the Bulk-In packet size.

An Ethernet frame always begins on a DWORD boundary. In MEF mode, the UTX will not concatenate the end of the current frame and the beginning of the next frame into the same DWORD. Therefore, the last DWORD of an Ethernet frame may have unused bytes added to ensure DWORD alignment of the next frame. The addition of pad bytes depends on whether another frame is available for transmission after the current one. If the current frame is the last frame to be transmitted, no pad bytes will be added, as the USB protocol allows for termination of the packet on a byte boundary. If, however, another frame is available for transmission, the current frame will be padded out so that it ends on the DWORD boundary. This ensures the next frame to be transmitted will start on a DWORD boundary.

If the UTX receives a Bulk-In Token when the RX FIFO is empty, it will transmit a ZLP.

Note:	Any unused bytes that were added to the last DWORD of a frame are not counted in the length field of the RX Status Word.

Note:	The host ignores unused bytes that exist in the first DWORD and last words of an Ethernet frame.

Note: When using SEF mode, there will never be any unused bytes added for end alignment padding. The USB transfer always ends on the last byte of the Ethernet frame.

Note: When RX COE is enabled, the last byte would pertain to the RX COE Word.

Once a decision is made to end a transfer and a short packet or ZLP has been sent, it is possible that an Ethernet frame will arrive prior to the UTX seeing an ACK from the host for the previous Bulk-In packet. In this case, the UTX must continue to repeat the short packet or ZLP until the ACK is received for the end of the previous transfer. The UTX must not start a new transfer, or re-use the previous data toggle, to begin sending the next Ethernet frame until the ACK has been received for the end of the previous transfer.

In order to more efficiently utilize USB bandwidth in MEF mode, the UTX has a mechanism for delaying the transmission of a short packet, or ZLP. This mode entails having the UTX wait a time defined by the Bulk-In Delay Register (BULK_IN_DLY) before terminating the burst. A value of zero in this register disables this feature. By default, a delay of 34 µs is used.

After the UTX transmits the last USB wPacketSize packet in a burst, the UTX will enable an internal timer. When the Bulk-In Delay time expires, any Bulk-In data will be transmitted upon reception of the next Bulk-In Token. If enough data arrives before the timer elapses to build at least one maximum sized packet, then the UTX will transmit this packet when it receives the next Bulk-In Token. After packet transmission, the UTX will then reset its internal timer and delay the short packet, or ZLP, transmission until the Bulk-In Delay time elapses.

In the case where the FIFO is empty and a single Ethernet packet less than the USB wPacketSize has been received, the UTX will enable its internal timer. If enough data arrives before the timer elapses to build at least one maximum sized packet, then the UTX will transmit this packet when it receives the next Bulk-In Token and will reset the internal timer. Otherwise, the short packet, or ZLP, is sent in response to the first Bulk-In Token received after the timer expires.

The UTX will NACK any Bulk-In tokens while waiting for the Bulk-In Delay to elapse. This NACK response is not affected by the Bulk-In Empty Response (BIR). The Bulk-In Empty Response (BIR) setting only applies after the Bulk-In Delay time expires.

The UTX, via the Burst Cap Register (BURST_CAP), is capable or prematurely terminating a burst. When the amount transmitted exceeds the value specified in this register, the UTX transmits a ZLP after the current Bulk-In packet completes. The Burst Cap Register (BURST_CAP) uses units of USB packet size (512 bytes). To enable use of the Burst Cap register, the Burst Cap Enable (BCE) bit in the Hardware Configuration Register (HW_CFG) must be set. For proper operation, the BURST_CAP field should be set to a value greater than 4. Burst Cap enforcement is disabled if BURST_CAP is set to a value less than or equal to 4.

Whenever Burst Cap enforcement is disabled, the UTX will respond with a ZLP (when Bulk-In Empty Response (BIR) =0) or with NACK (when Bulk-In Empty Response (BIR) = 1).

Whenever Burst Cap enforcement is enabled (BURST_CAP value is legal), the following holds:

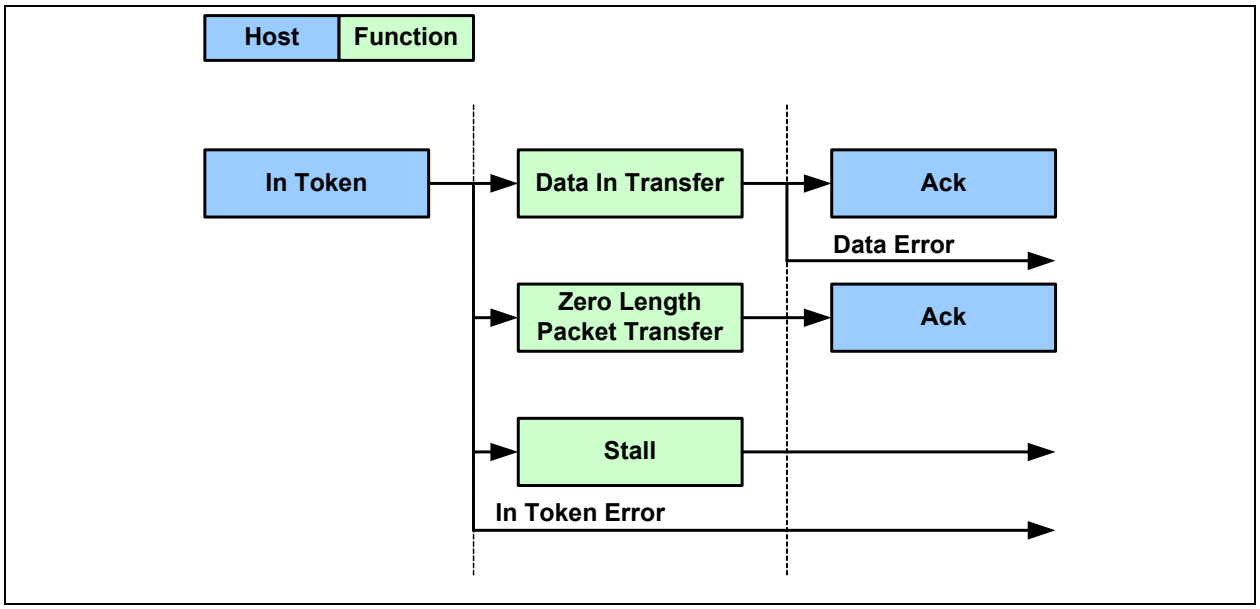
- Let BURST = BURST_CAP * 512
  The burst may terminate at BURST-4, BURST-3, BURST-2, BURST-1, or BURST bytes, or, when the RX FIFO runs out of data. The burst is terminated with either a short USB packet or with a ZLP.

Note: Ethernet frames are not fragmented across bursts when using Burst Cap enforcement.

In the case of an error condition, the UTX will issue a rewind to the FCT. This occurs when the UTX completes transmitting a Bulk-In packet and does not receive an ACK from the host. In this case, the next frame received by the UTX will be another In token and the Bulk-In packet is retransmitted. When the ACK is finally received, the UTX notifies the FCT. The FCT will then advance the read head pointer to the next packet.

Note: The UTX will never stall the Endpoint. The Endpoint can only be stalled by the host.

FIGURE 4-2: USB BULK-IN TRANSACTION SUMMARY



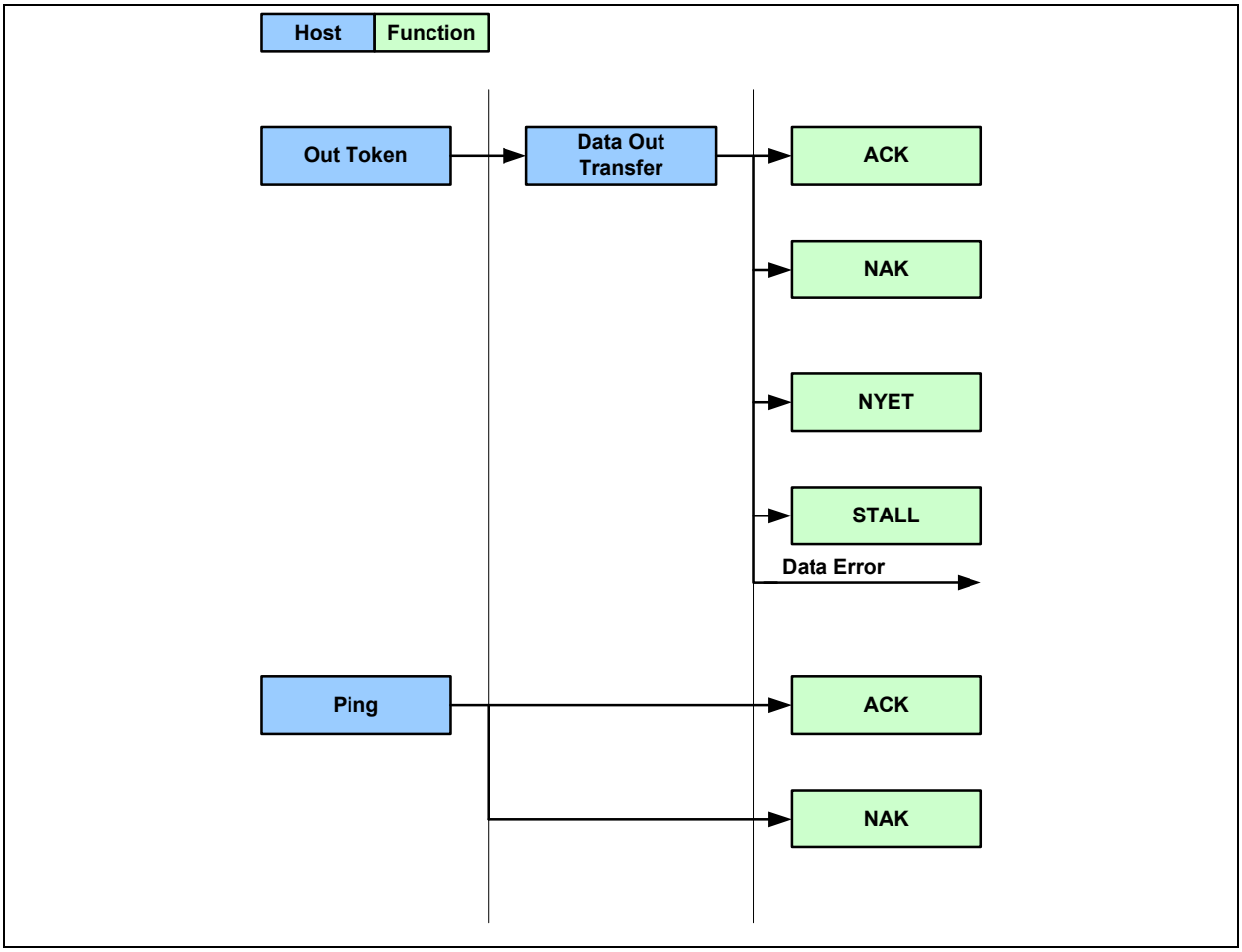
4.3.1.2 Endpoint 2 (Bulk-Out)

The Bulk-Out Endpoint is controlled by the URX (USB Bulk-Out Receiver). The URX is responsible for receiving Ethernet data encapsulated over a USB Bulk-Out packet. Unlike the UTX, the URX does not explicitly track Ethernet frames. It views all received packets as purely USB data. The extraction of Ethernet frames is handled by the FCT and the Transaction Layer Interface (TLI).

The URX always simultaneously supports multiple Ethernet frames per USB packet, as well as a single Ethernet frame per USB packet. No mechanism exists to select between modes.

The URX monitors the amount of free space in the TX FIFO. If at least 512 bytes of space exists, the URX can accept an additional Bulk-In frame and responds to a Bulk-Out Token with an ACK or NYET. The NYET response is used when less than 1024 bytes of free space exist. This means that the current Bulk-Out packet was accepted, but room does not exist for a second packet. If less than 512 bytes exist, the URX responds with a NACK. The URX supports the PING protocol.

FIGURE 4-3: USB BULK-OUT TRANSACTION SUMMARY



In the case where the Bulk-Out packet is errored, the URX does not respond to the host. The URX will request that the FCT rewinds the packet. It is the host's responsibility to retransmit the packet at a later time.

The FCT notifies the URX when it detects loss of sync. When this occurs, the URX stalls the Bulk-Out pipe. This is an appropriate response, as loss of sync is a catastrophic error. This behavior is configurable via the Stall Bulk-Out Pipe Disable (SBP) bit of the Hardware Configuration Register (HW_CFG) on page 125.

4.3.1.3 Endpoint 3 (Interrupt)

The Interrupt Endpoint is responsible for indicating device status at each polling interval. The Interrupt Endpoint is implemented via the CTL module. When the Endpoint is accessed, the Interrupt packet specified in Table 4-2 is presented to the host.

TABLE 4-2: INTERRUPT PACKET FORMAT

Bits	Description
31:20	**RESERVED**
19	**MACRTO_INT**
18	**RX FIFO Has Frame.** The RX FIFO has at least one complete Ethernet frame.
17	**TXSTOP_INT**
16	**RXSTOP_INT**
15	**PHY_INT**
14	**TXE**
13	**TDFU**
12	**TDFO**
11	**RXDF_INT**
10:0	**GPIO_INT**

If there is no interrupt status to report, the device responds with a NACK.

Note:	The polling interval is static and set through the EEPROM. The host can change the polling interval by updating the contents of the EEPROM and resetting the part.

The interrupt status can be cleared by writing to Interrupt Status Register (INT_STS).

4.3.1.4 Endpoint 0 (Control)

The Control Endpoint is handled by the CTL (USB Control) module. The CTL module is responsible for handling USB standard commands, as well as USB vendor commands. In order to support these commands, the CTL must compile a variety of statistics and store the programmable portions of the USB descriptors. The supported USB commands can be found in Section 4.3.2, "USB Standard Commands".

4.3.1.5 USB Command Processing

The UDC is programmed to decode USB commands. After a standard command is decoded by the UDC, it may be passed to the CTL for completion. The CTL is responsible for implementing the Get Descriptor and vendor commands.

In order to implement the Get Descriptor command for String Descriptors, the CTL manages a 128 x 32 register file which stores the string values for Language ID, Manufacturer ID, Product ID, Serial Number, Configuration, and Interface. The RAM's contents is initialized via the EEPROM, after a system reset occurs.

TABLE 4-3: STRING DESCRIPTOR INDEX MAPPINGS

Index	String Name
0	Language ID
1	Manufacturer ID
2	Product ID
3	Serial Number
4	Configuration String
5	Interface String

When the UDC decodes a Get Descriptor command, it will pass a pointer to the CTL. The CTL uses this pointer to determine what the command is and how to fill it.

4.3.1.6 USB Descriptors

The following subsections describe the USB descriptors.

4.3.1.6.1 Device Descriptor

The Device Descriptors are initialized based on values stored in EEPROM. Table 4-4 shows the default Device Descriptor values.

TABLE 4-4: DEVICE DESCRIPTOR

Offset	Field	Size (Bytes)	Default Value	Loaded from EEPROM	Description
00h	bLength	1	12h	Note 4-1	Size of the descriptor in bytes (18 bytes)
01h	bDescriptorType	1	01h	Note 4-1	Device descriptor (0x01)
02h	bcdUSB	2	0200h	Note 4-2	USB Specification Number which device complies to.
04h	bDeviceClass	1	FFh	Yes	Class Code
05h	bDeviceSubClass	1	00h	Yes	Subclass Code
06h	bDeviceProtocol	1	FFh	Yes	Protocol Code
07h	bMaxPacketSize	1	40h	Note 4-2	Maximum Packet Size for Endpoint 0
08h	IdVendor	2	0424h	Yes	Vendor ID
0Ah	IdProduct	2	9730h	Yes	Product ID
0Ch	bcdDevice	2	Note 4-3	Yes	Device Release Number
0Eh	iManufacturer	1	00h	Yes	Index of Manufacturer String Descriptor
0Fh	iProduct	1	00h	Yes	Index of Product String Descriptor
10h	iSerialNumber	1	00h	Yes	Index of Serial Number String Descriptor
11h	bNumConfigurations	1	01h	Note 4-2	Number of possible configurations

Note 4-1 The descriptor length and descriptor type for Device Descriptors specified in EEPROM are "don't cares" and are always overwritten by hardware as 0x12 and 0x01, respectively.

Note 4-2 Value is loaded from EEPROM, but must be equal to the default value in order to comply with the USB 2.0 Specification and provide for normal device operation. Specification of any other value will result in unwanted behavior and untoward operation.

Note 4-3 Default value is dependent on device release. The MSB matches the device release and the LSB is hardcoded to 00h. The initial release value is 01h.

4.3.1.6.2 Configuration Descriptor

The Configuration Descriptor is initialized based on values stored in EEPROM. Table 4-5 shows the default Configuration Descriptor values.

TABLE 4-5: CONFIGURATION DESCRIPTOR

Offset	Field	Size (Bytes)	Default Value	Loaded from EEPROM	Description
00h	bLength	1	09h	Note 4-4	Size of the Configuration Descriptor in bytes (9 bytes)
01h	bDescriptorType	1	02h	Note 4-5	Configuration Descriptor (0x02)
02h	wTotalLength	2	0027h	Note 4-4	Total length in bytes of data returned (39 bytes)
04h	bNumInterfaces	1	01h	Note 4-4	Number of interfaces
05h	bConfigurationValue	1	01h	Note 4-4	Value to use as an argument to select this configuration
06h	iConfiguration	1	00h	Yes	Index of String Descriptor describing this configuration
07h	bmAttributes	1	A0h	Yes	Bus powered and remote wakeup enabled.
08h	bMaxPower	1	Note 4-6	Yes	Maximum power consumption is 500 mA.

Note 4-4 Value is loaded from EEPROM, but must be equal to the default value in order to comply with the USB 2.0 Specification and provide for normal device operation. Specification of any other value will result in unwanted behavior and untoward operation.

Note 4-5 The descriptor type for Configuration Descriptors specified in EEPROM is a “don’t care” and is always overwritten by hardware as 0x02.

Note 4-6 Default value is 01h in Self-Powered mode and FAh in Bus Powered mode.

Note:	The RMT_WKP strap affects the default value of bmAttributes.

4.3.1.6.3 Interface Descriptor 0 Default

Table 4-6 shows the default value for Interface Descriptor 0. This descriptor is initialized based on values stored in EEPROM.

TABLE 4-6: INTERFACE DESCRIPTOR 0

Offset	Field	Size (Bytes)	Default Value	Loaded from EEPROM	Description
00h	bLength	1	09h	Note 4-7	Size of descriptor in bytes (9 bytes)
01h	bDescriptorType	1	04h	Note 4-7	Interface Descriptor (0x04)
02h	bInterfaceNumber	1	00h	Note 4-7	Number identifying this interface
03h	bAlternateSetting	1	00h	Note 4-7	Value used to select alternative setting
04h	bNumEndpoints	1	03h	Note 4-7	Number of Endpoints used for this interface (less Endpoint 0)
05h	bInterfaceClass	1	FFh	Yes	Class Code
06h	bInterfaceSubClass	1	00h	Yes	Subclass Code
07h	bInterfaceProtocol	1	FFh	Yes	Protocol Code
08h	iInterface	1	00h	Yes	Index of String Descriptor describing this interface

Note 4-7 Value is loaded from EEPROM, but must be equal to the default value in order to comply with the USB 2.0 Specification and provide for normal device operation. Specification of any other value will result in unwanted behavior and untoward operation.

4.3.1.6.4 Endpoint 1 (Bulk-In) Descriptor

Table 4-7 shows the default value for Endpoint Descriptor 1. This descriptor is not initialized from values stored in EEPROM.

TABLE 4-7: ENDPOINT 1 DESCRIPTOR

Offset	Field	Size (Bytes)	Default Value	Loaded from EEPROM	Description
00h	bLength	1	07h	No	Size of descriptor in bytes
01h	bDescriptorType	1	05h	No	Endpoint Descriptor
02h	bEndpointAddress	1	81h	No	Endpoint Address
03h	bmAttributes	1	02h	No	Bulk Transfer Type
04h	wMaxPacketSize	2	Note 4-8	No	Maximum packet size this Endpoint is capable of sending.
06h	bInterval	1	00h	No	Interval for polling Endpoint data transfers. Ignored for bulk Endpoints

Note 4-8 512 bytes for Hi-Speed mode.

4.3.1.6.5 Endpoint 2 (Bulk-Out) Descriptor

Table 4-8 shows the default value for Endpoint Descriptor 2. This descriptor is not initialized from values stored in EEPROM.

TABLE 4-8: ENDPOINT 2 DESCRIPTOR

Offset	Field	Size (Bytes)	Default Value	Loaded from EEPROM	Description
00h	bLength	1	07h	No	Size of descriptor in bytes
01h	bDescriptorType	1	05h	No	Endpoint Descriptor
02h	bEndpointAddress	1	02h	No	Endpoint Address
03h	bmAttributes	1	02h	No	Bulk Transfer Type
04h	wMaxPacketSize	2	Note 4-9	No	Maximum packet size this Endpoint is capable of sending.
06h	bInterval	1	00h	No	Interval for polling Endpoint data transfers. Ignored for bulk Endpoints

Note 4-9 512 bytes for Hi-Speed mode.

4.3.1.6.6 Endpoint 3 (Interrupt) Descriptor

Table 4-9 shows the default value for Endpoint Descriptor 3. Only the bInterval field of this descriptor is initialized from EEPROM.

TABLE 4-9: ENDPOINT 3 DESCRIPTOR

Offset	Field	Size (Bytes)	Default Value	Loaded from EEPROM	Description
00h	bLength	1	07h	No	Size of descriptor in bytes
01h	bDescriptorType	1	05h	No	Endpoint Descriptor
02h	bEndpointAddress	1	83h	No	Endpoint Address
03h	bmAttributes	1	03h	No	Interrupt Transfer Type
04h	wMaxPacketSize	2	10h	No	Maximum packet size this Endpoint is capable of sending.
06h	bInterval	1	Note 4-10	Yes	Interval for polling Endpoint data transfers.

Note 4-10 This value is loaded from the EEPROM. If no EEPROM exists then this value defaults to 04h.

4.3.1.6.7 Other Speed Configuration Descriptor

The fields in this descriptor are derived from Configuration Descriptor information that is stored in the EEPROM.

TABLE 4-10: OTHER SPEED CONFIGURATION DESCRIPTOR

Offset	Field	Size (Bytes)	Default Value	Loaded from EEPROM	Description
00h	bLength	1	09h	Note 4-11	Size of descriptor in bytes (9 bytes)
01h	bDescriptorType	1	07h	Note 4-11	Other Speed Configuration Descriptor (0x07)
02h	wTotalLength	2	0027h	Note 4-11	Total length in bytes of data returned (39 bytes)
04h	bNumInterfaces	1	01h	Note 4-11	Number of interfaces
05h	bConfigurationValue	1	01h	Note 4-11	Value to use as an argument to select this configuration
06h	iConfiguration	1	00h	Yes	Index of String Descriptor describing this configuration
07h	bmAttributes	1	A0h	Yes	Bus powered and remote wakeup enabled.
08h	bMaxPower	1	Note 4-12	Yes	Maximum power consumption is 500 mA.

Note: EEPROM values are obtained for the Configuration Descriptor at the other USB speed. I.e., if the current operating speed is FS, then the HS Configuration Descriptor values are used, and vice-versa.

Note 4-11 Value is loaded from EEPROM, but must be equal to the default value in order to comply with the USB 2.0 Specification and provide for normal device operation. Specification of any other value will result in unwanted behavior and untoward operation.

Note 4-12 Default value is 01h in Self-Powered mode and FAh in Bus-Powered mode.

Note: The RMT_WKP strap affects the default value of bmAttributes.

4.3.1.6.8 Device Qualifier Descriptor

The fields in this descriptor are derived from Device Descriptor information that is stored in the EEPROM.

TABLE 4-11: DEVICE QUALIFIER DESCRIPTOR

Offset	Field	Size (Bytes)	Default Value	Loaded from EEPROM	Description
00h	bLength	1	0Ah	No	Size of descriptor in bytes (10 bytes)
01h	bDescriptorType	1	06h	No	Device Qualifier Descriptor (0x06)
02h	bcdUSB	2	0200h	Note 4-13	USB Specification Number which device complies to.
04h	bDeviceClass	1	FFh	Yes	Class Code
05h	bDeviceSubClass	1	00h	Yes	Subclass Code
06h	bDeviceProtocol	1	FFh	Yes	Protocol Code
07h	bMaxPacketSize0	1	40h	Note 4-13	Maximum packet size
08h	bNumConfigurations	1	01h	Note 4-13	Number of Other Speed Configurations
09h	Reserved	1	00h	No	Must be zero

Note:	EEPROM values are from the Device Descriptor (including any EEPROM override) at the opposite HS/FS operating speed. I.e., if the current operating speed is HS, then Device Qualifier data is based on the FS Device Descriptor, and vice-versa.

Note 4-13 Value is loaded from EEPROM, but must be equal to the default value in order to comply with the USB 2.0 Specification and provide for normal device operation. Specification of any other value will result in unwanted behavior and untoward operation.

4.3.1.6.9 String Descriptors

String Index = 0 (LANGID)

TABLE 4-12: LANGID STRING DESCRIPTOR

Offset	Field	Size (Bytes)	Default Value	Loaded from EEPROM	Description
00h	bLength	1	04h	No	Size of LANGID Descriptor in bytes (4 bytes)
01h	bDescriptorType	1	03h	No	String Descriptor (0x03)
02h	LANGID	2	None	Yes	Must be set to 0x0409 (US English).

Note:	If there is no valid/enabled EEPROM, or if all string lengths in the EEPROM are 0, then there are no strings, so any host attempt to read the LANGID string will return stall in the Data Stage of the Control Transfer. If there is a valid/enabled EEPROM, and if at least one of the string lengths in the EEPROM is not 0, then the value contained at EEPROM addresses 0x0A-0x0B will be returned. These must be 0x0409 to allow for proper device operation.

Note:	The device ignores the LANGID field in Control Read's of Strings, and will not return the String (if it exists), regardless of whether the requested LANGID is 0x0409 or not.

String Indices 1-5

TABLE 4-13: STRING DESCRIPTOR (INDICES 1-5)

Offset	Field	Size (Bytes)	Default Value	Loaded from EEPROM	Description
00h	bLength	1	none	Yes	Size of the String Descriptor in bytes (4 bytes)
01h	bDescriptorType	1	none	Yes	String Descriptor (0x03)
02h	Unicode String	2*N	none	Yes	2 bytes per unicode character, no trailing NULL.

Note:	If there is no valid/enabled EEPROM, or if the corresponding String Length and offset in the EEPROM for a given string index are zero, then that string does not exist, so any host attempt to read that string will return stall in the Data Stage of the Control Transfer.

Note:	The device returns whatever bytes are in the designated EEPROM area for each of these strings. It is the responsibility of the EEPROM programmer to correctly set the bLength and bDescriptorType fields in the descriptor consistent with the byte length specified in the corresponding EEPROM locations.

4.3.1.7 Statistics

The CTL tracks the statistics listed in Table 4-14. The statistics are read via the Get Statistics Vendor Command.

Note: The counters are snapshot when fulfilling the command request. The statistics counters rollover.

Error conditions are indicated via the RX Status Word, Table 4-40, or the TX Status Word, Table 4-44.

TABLE 4-14: STATISTIC COUNTERS

Name	Description	Size (Bits)
RX Good Frames	Number of good RX frames received. Includes frames dropped by the FCT.	32
RX CRC Errors	Number of RX frames received with CRC-32 errors. A CRC error is indicated when the CRC error flag is set and the dribbling bit flag is not set.	20
RX Runt Frame Errors	Number of RX frames received with a length of less than 64 bytes and a CRC error.	20
RX Alignment Errors	Number of RX frames received with alignment errors. An alignment error is indicated by the presence of the CRC error flag and the dribbling bit flag is set.	20
RX Frame Too Long Error	Number of RX frames received with a length greater than the programmed maximum Ethernet frame size.	20
RX Later Collision Error	Number of RX frames received where a late collision has occurred.	20
RX Bad Frames	Total number of errored Ethernet frames received. This counter does not include RX FIFO Dropped Frames.	20
RX FIFO Dropped Frames	Number of RX frames dropped by the FCT due to insufficient room in the RX FIFO. If an RX FIFO Dropped Frame has an Ethernet error, i.e., CRC error, it must only be counted by the RX FIFO Dropped Frames counters.	20
TX Good Frames	Number of successfully transmitted TX frames. Does not count pause frames.	32
TX Pause Frames	Number of successfully transmitted pause frames.	20
TX Single Collisions	Number of successfully transmitted frames with one collision.	20
TX Multiple Collisions	Number of successfully transmitted frames with more than one collision.	20
TX Excessive Collision Errors	Number of transmitted frames aborted due to excessive collisions.	20
TX Late Collision Errors	Number of transmitted frames aborted due to late collisions.	20
TX Buffer Underrun Errors	Number of transmitted frames aborted due to Tx buffer under run.	20
TX Excessive Deferral Errors	Number of transmitted frames aborted due to excessive deferrals.	20
TX Carrier Errors	Number of frames transmitted in which the carrier signal was lost or in which the carrier signal was not present.	20
TX Bad Frames	Total number of errored Ethernet frames transmitted.	20

4.3.2 USB STANDARD COMMANDS

This section lists the formats of the supported USB Standard Commands. The Set Descriptor, Set Interface, and Synch Frame commands are not supported.

4.3.2.1 Clear Feature

This command clears the Stall status of the targeted Endpoint or the device remote wakeup.

TABLE 4-15: FORMAT OF CLEAR FEATURE SETUP STAGE

Offset	Field	Value
0h	bmRequestType	Note 4-14
1h	bRequest	01h
2h	wValue	Selects feature to clear.
4h	wIndex	Note 4-15
6h	wLength	00h

Note 4-14 Set to 00h to clear device remote wakeup event. Set to 02h to clear the Endpoint stall status.

Note 4-15 When the bmRequestType field specifies an Endpoint, the wIndex field selects the Endpoint (0, 1, 2, or 3) targeted by the command.

4.3.2.2 Get Configuration

TABLE 4-16: FORMAT OF GET CONFIGURATION SETUP STAGE

Offset	Field	Value
0h	bmRequestType	80h
1h	bRequest	08h
2h	wValue	00h
4h	wIndex	00h
6h	wLength	01h

TABLE 4-17: FORMAT OF GET CONFIGURATION DATA STAGE

Offset	Field
0h	Returns bConfigurationValue

4.3.2.3 Get Descriptor

TABLE 4-18: FORMAT OF GET DESCRIPTOR SETUP STAGE

Offset	Field	Value
0h	bmRequestType	80h
1h	bRequest	06h
2h	wValue	Note 4-16
4h	wIndex	Note 4-17
6h	wLength	Length of descriptor

Note 4-16 Selects descriptor type. The supported descriptors for this command are Device, Configuration, String, Device Qualifier, and Other Speed Configuration.

Note 4-17 Set to zero or Language ID.

Note: The Interface and Endpoint descriptors are not supported by this command. The UDC will stall these requests.

4.3.2.4 Get Interface

TABLE 4-19: FORMAT OF GET INTERFACE SETUP STAGE

Offset	Field	Value
0h	bmRequestType	81h
1h	bRequest	0Ah
2h	wValue	00h
4h	wIndex	00h
6h	wLength	01h

TABLE 4-20: FORMAT OF GET INTERFACE DATA STAGE

Offset	Field
0h	Alternate Setting

Note: The device only supports a single interface.

4.3.2.5 Get Status

4.3.2.5.1 Device Status

TABLE 4-21: FORMAT OF GET STATUS (DEVICE) SETUP STAGE

Offset	Field	Value
0h	bmRequestType	80h
1h	bRequest	00h
2h	wValue	00h
4h	wIndex	00h
6h	wLength	02h

TABLE 4-22: FORMAT OF GET STATUS (DEVICE) DATA STAGE

Offset	Field
0h	{00h, 0h, 00b, Remote Wakeup, Self Powered}

4.3.2.5.2 Endpoint 1 Status (Bulk-In)

TABLE 4-23: FORMAT OF GET STATUS (ENDPOINT 1) SETUP STAGE

Offset	Field	Value
0h	bmRequestType	82h
1h	bRequest	00h
2h	wValue	00h
4h	wIndex	81h
6h	wLength	02h

TABLE 4-24: FORMAT OF GET STATUS (ENDPOINT 1) DATA STAGE

Offset	Field
0h	{00h, 0h, 000b, Stall status}

4.3.2.5.3 Endpoint 2 Status (Bulk-Out)

TABLE 4-25: FORMAT OF GET STATUS (ENDPOINT 2) SETUP STAGE

Offset	Field	Value
0h	bmRequestType	82h
1h	bRequest	00h
2h	wValue	00h
4h	wIndex	02h
6h	wLength	02h

TABLE 4-26: FORMAT OF GET STATUS (ENDPOINT 2) DATA STAGE

Offset	Field
0h	{00h, 0h, 000b, Stall status}

4.3.2.5.4 Endpoint 3 Status (Interrupt)

TABLE 4-27: FORMAT OF GET STATUS (ENDPOINT 3) SETUP STAGE

Offset	Field	Value
0h	bmRequestType	82h
1h	bRequest	00h
2h	wValue	00h
4h	wIndex	83h
6h	wLength	02h

TABLE 4-28: FORMAT OF GET STATUS (ENDPOINT 3) DATA STAGE

Offset	Field
0h	{00h, 0h, 000b, Stall status}

4.3.2.5.5 Set Address

TABLE 4-29: FORMAT OF SET ADDRESS SETUP STAGE

Offset	Field	Value
0h	bmRequestType	00h
1h	bRequest	05h
2h	wValue	Device Address
4h	wIndex	00h
6h	wLength	00h

4.3.2.5.6 Set Feature

This command sets the Stall feature for all supported Endpoints. It also supports the Device Remote Wakeup feature and Test mode.

TABLE 4-30: FORMAT OF SET FEATURE SETUP STAGE

Offset	Field	Value
0h	bmRequestType	• 00h for device • 02h for Endpoint
1h	bRequest	03h
2h	wValue	• 01h for DEVICE_REMOTE_WAKEUP • 00h for ENDPOINT_HALT • 02h for EST_MODE
4h	wIndex	• 00h for device remote wakeup • 00h for TEST_MODE • Interface Endpoint number for halt
6h	wLength	00h

4.3.2.5.7 Set Configuration

The device supports only one configuration. An occurrence of this command places the device into the Configured state.

TABLE 4-31: FORMAT OF SET CONFIGURATION SETUP STAGE

Offset	Field	Value
0h	bmRequestType	00h
1h	bRequest	09h
2h	wValue	Configuration Value
4h	wIndex	00h
6h	wLength	00h

Since only one configuration is supported, 01h is the only supported configuration value.

4.3.2.5.8 Set Interface

Only one interface is supported by the device. Therefore, this command is of marginal use. If the command is issued with an alternative setting of 00h and interface setting of 00h, as shown in Table 4-32, the device responds with an ACK. Otherwise it responds with a STALL handshake.

TABLE 4-32: FORMAT OF SET INTERFACE SETUP STAGE

Offset	Field	Value
0h	bmRequestType	01h
1h	bRequest	0Bh
2h	wValue	00h
4h	wIndex	00h
6h	wLength	00h

4.3.3 USB VENDOR COMMANDS

The device implements several vendor specific commands in order to access CSRs and efficiently gather statistics. The vendor commands allow direct access to Systems CSRs and MAC CSRs.

Note:	When in the Normal state, accesses to the MAC CSRs are stalled.

4.3.3.1 Register Write Command

The commands allows the host to write a single register. Burst writes are not supported. All writes are 32-bits.

TABLE 4-33: FORMAT OF REGISTER WRITE SETUP STAGE

Offset	Field	Value
0h	bmRequestType	40h
1h	bRequest	A0h
2h	wValue	00h
4h	wIndex	{0h, CSR Address[11:0]}
6h	wLength	04h

TABLE 4-34: FORMAT OF REGISTER WRITE DATA STAGE

Offset	Field
0h	Register Write Data [31:0]

4.3.3.2 Register Read Command

The commands allows the host to read a single register. Burst reads are not supported. All reads return 32-bits.

TABLE 4-35: FORMAT OF REGISTER READ SETUP STAGE

Offset	Field	Value
0h	bmRequestType	C0h
1h	bRequest	A1h
2h	wValue	00h
4h	wIndex	{0h, CSR Address[11:0]}
6h	wLength	04h

TABLE 4-36: FORMAT OF REGISTER READ DATA STAGE

Offset	Field
0h	Register Read Data [31:0]

4.3.3.3 Get Statistics Command

The Get Statistics Command returns the entire contents of the statistics RAMs. The wIndex field is used to select the RX or TX statistics.

Note:	The contents of the statistics RAM is snapshot when fulfilling the command request. The statistics counters rollover, hence the RAM is not cleared.

TABLE 4-37: FORMAT OF GET STATISTICS SETUP STAGE

Offset	Field	Value
0h	bmRequestType	C0h
1h	bRequest	A2h
2h	wValue	00h
4h	wIndex	Note 4-18
6h	wLength	Note 4-19

Note 4-18 0b - Retrieves RX statistics. 1b - Retrieves TX statistics.

Note 4-19 28h for RX statistics. 30h for TX statistics.

TABLE 4-38: FORMAT OF GET STATISTICS DATA STAGE (RX)

Offset	Field
00h	RX Good Frames
04h	RX CRC Errors
08h	RX Runt Frame Errors
0Ch	RX Alignment Errors
10h	RX Frame Too Long Error
14h	RX Later Collision Error
18h	RX Bad Frames
1Ch	RX FIFO Dropped Frames
20h	RESERVED
24h	RESERVED

TABLE 4-39: FORMAT OF GET STATISTICS DATA STAGE (TX)

Offset	Field
00h	TX Good Frames
04h	TX Pause Frames
08h	TX Single Collisions
0Ch	TX Multiple Collisions
10h	TX Excessive Collision Errors
14h	TX Late Collision Errors
18h	TX Buffer Underrun Errors
1Ch	TX Excessive Deferral Errors
20h	TX Carrier Errors
24h	TX Bad Frames
28h	RESERVED
2Ch	RESERVED

4.4 FIFO Controller (FCT)

The FIFO controller uses a 28 kB internal SRAM to buffer RX and TX traffic. 20 kB are allocated for received Ethernet-USB traffic (RX buffer), while 8 kB are allocated for USB-Ethernet traffic (TX buffer). Bulk-Out packets from the USB controller are directly stored into the TX buffer. The FCT is responsible for extracting Ethernet frames from the USB packet data and passing the frames to the MAC. Ethernet frames are directly stored into the RX buffer and become the basis for Bulk-In packets. The FCT passes the stored data to the UTX in blocks typically 512 bytes in size.

4.4.1 RX PATH (ETHERNET -> USB)

The 20 kB RX FIFO buffers Ethernet frames received from the TLI. The UTX extracts these frames from the FCT to form USB Bulk-In packets. The host drivers will ultimately reassemble the Ethernet frames from the USB packets.

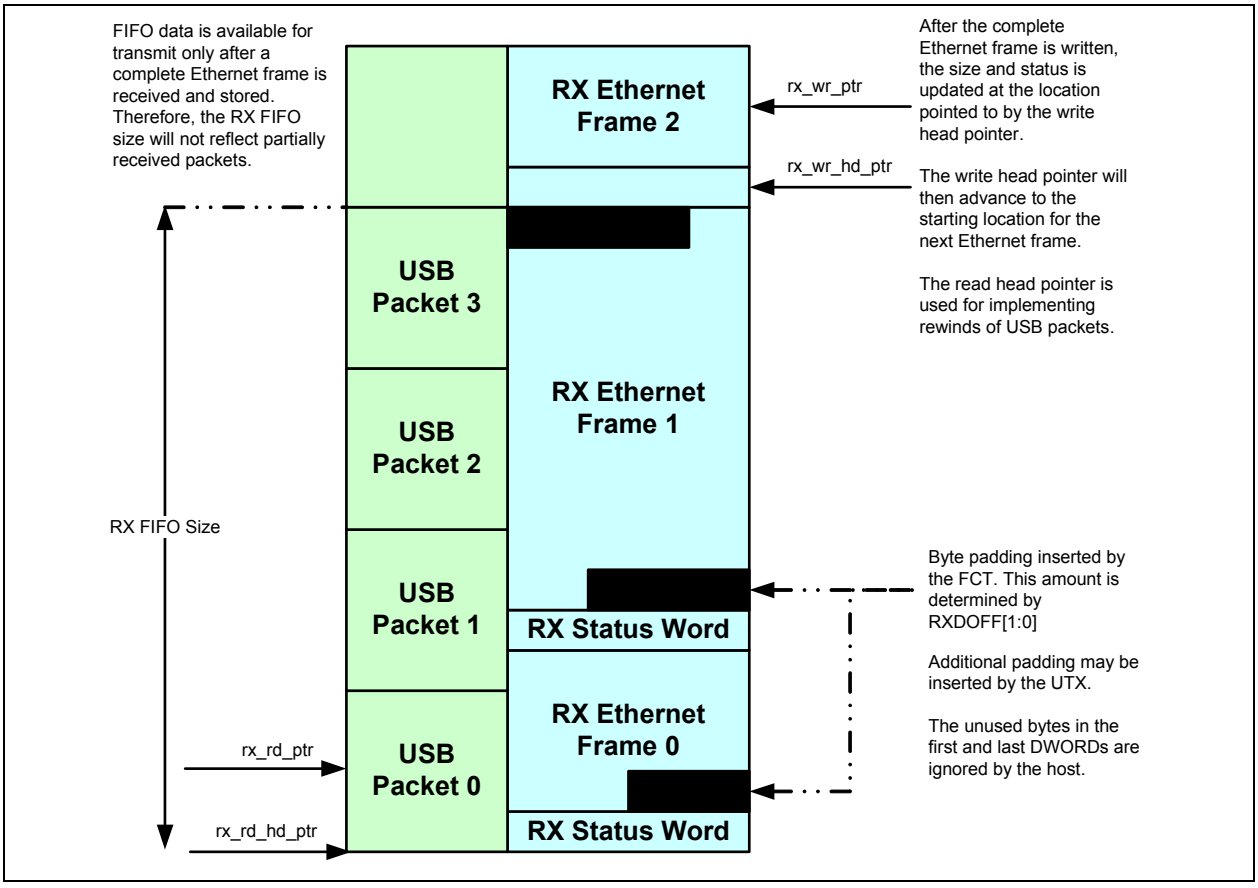
The FCT manages the writing of data into the RX FIFO through the use of two pointers - the rx_wr_ptr and the rx_wr_hd_ptr. The rx_wr_ptr is used to write Ethernet frame data into the FIFO. The rx_wr_hd_ptr points to the location prior to the first DWORD of the frame. It is used to write the RX Status Word received from the TLI, upon completion of a frame transaction. This status word contains status information associated with the frame and the frame transaction. Figure 4-4 illustrates how a frame is stored in the FIFO, along with pointer usage.

When the RX TLI signals that it has data ready, the RX TLI controller starts passing the RX packet data to the FCT. The FCT updates the RX FIFO pointers as the data is written into the FIFO. The last transfer from the TLI is the RX Status Word.

The FCT may insert 0-3 bytes at the start of the Ethernet frame. The value of the RX Data Offset (RXDOFF) field of the Hardware Configuration Register (HW_CFG) determines the number of bytes inserted.

A received Ethernet frame is not visible to the UTX until the complete frame, including the RX Status Word, has been written into the RX FIFO. This is due to the fact that the frame may have to be removed via a rewind (pointer adjustment), in case of an error. Such is the case when a FIFO overflow condition is detected as the frame is being received. The FCT may also be configured to rewind errored frames. Refer to Section 4.4.1.1, "RX Error Detection" for further details.

FIGURE 4-4: RX FIFO STORAGE



4.4.1.1 RX Error Detection

The FCT can be configured to drop Ethernet frames when certain error conditions occur. The setting of the Discard Errored Received Ethernet Frame (DRP) bit of the Hardware Configuration Register (HW_CFG) on page 125 determines if the frame will be retained or dropped. Error conditions are indicated in the Rx Status Word. The following error conditions are tracked by the TLI:

- CRC Error
- Collision Seen
- Frame Too Long
- Runt Frame

Refer to Section 4.3.1.7, "Statistics" for more details on the error conditions tracked by the device.

The FCT also drops frames when it detects a FIFO overflow condition. This occurs when the FIFO full condition occurs while a frame is being received. The FCT also maintains a count of the number of times a FIFO overflow condition has occurred.

Dropping an Ethernet frame is implemented by rewinding the received frame. A write side rewind is implemented by setting the rx_wr_ptr to be equal to the rx_wr_hd_ptr. Similarly, a read side rewind is implemented by setting the rx_rd_ptr to be equal to the rx_rd_hd_ptr.

For the case where the frame is dropped due to overflow, the FCT ignores the remainder of the frame. It will not begin writing into the RX FIFO again until the next frame is received.

In the read direction, the FCT must also support rewinds for the UTX. This is needed for the case where the USB Bulk-Out packet is not successfully received by the host and needs to be retransmitted.

4.4.1.2 RX Status Format

Table 4-40 illustrates the format of the RX Status Word.

TABLE 4-40: RX STATUS WORD FORMAT

Bits	Description
31	RESERVED
30	Filtering Fail When set, this bit indicates that the associated frame failed the address recognizing filtering.
29:16	Frame Length The size, in bytes, of the corresponding received frame.
15	Error Status (ES) When set, this bit indicates that the TLI has reported an error. This bit is the logical OR of bits 11, 7, 6, 1 in this status word.
14	RESERVED
13	Broadcast Frame When set, this bit indicates that the received frame has a Broadcast address.
12	Length Error (LE) When set, this bit indicates that the actual length does not match with the length/type field of the received frame.
11	Runt Frame When set, this bit indicates that frame was prematurely terminated before the collision window (64 bytes). Runt frames are passed on to the host only if the Pass Bad Frames bit MAC_CR Bit [16] is set.
10	Multicast Frame When set, this bit indicates that the received frame has a Multicast address.
9:8	RESERVED
7	Frame Too Long When set, this bit indicates that the frame length exceeds the maximum Ethernet specification of 1518 bytes. This is only a Frame Too Long indication and will not cause the frame reception to be truncated.

TABLE 4-40: RX STATUS WORD FORMAT (CONTINUED)

Bits	Description
6	Collision Seen When set, this bit indicates that the frame has seen a collision after the collision window. This indicates that a late collision has occurred.
5	Frame Type When set, this bit indicates that the frame is an Ethernet-type frame (length/type field in the frame is greater than 1500). When reset, it indicates the incoming frame was an 802.3 type frame. This bit is not set for Runt frames less than 14 bytes.
4	Receive Watchdog Time-Out When set, this bit indicates that the incoming frame is greater than 2048 bytes through 2560 bytes, therefore expiring the Receive Watchdog Timer.
3	MII Error When set, this bit indicates that a receive error (RX_ER asserted) was detected during frame reception.
2	Dribbling Bit When set, this bit indicates that the frame contained a no-integer multiple of 8 bits. This error is reported only if the number of dribbling bits in the last byte is 4 in the MII operating mode, or at least 3 in the 10 Mbps operating mode. This bit will not be set when the Collision Seen bit[6] is set. If set and the CRC error[1] bit is reset, then the frame is considered to be valid.
1	CRC Error When set, this bit indicates that a CRC error was detected. This bit is also set when the RX_ER pin is asserted during the reception of a frame even though the CRC may be correct. This bit is not valid if the received frame is a Runt frame, or a late collision was detected or when the Watchdog time-out occurs.
0	RESERVED

4.4.1.3 Flushing the RX FIFO

The device allows for the host to flush the entire contents of the FCT RX FIFO. When a flush is activated, the read and write pointers of the RX FIFO are returned to their reset state.

Before flushing the RX FIFO, the device's receiver must be stopped, as specified in Section 4.4.1.4. Once the receiver stop completion is confirmed, the Receive FIFO Flush bit can be set in the Receive Configuration Register (RX_CFG) on page 123 to initiate the flush operation. This bit is cleared after the flush is complete.

4.4.1.4 Stopping and Starting the Receiver

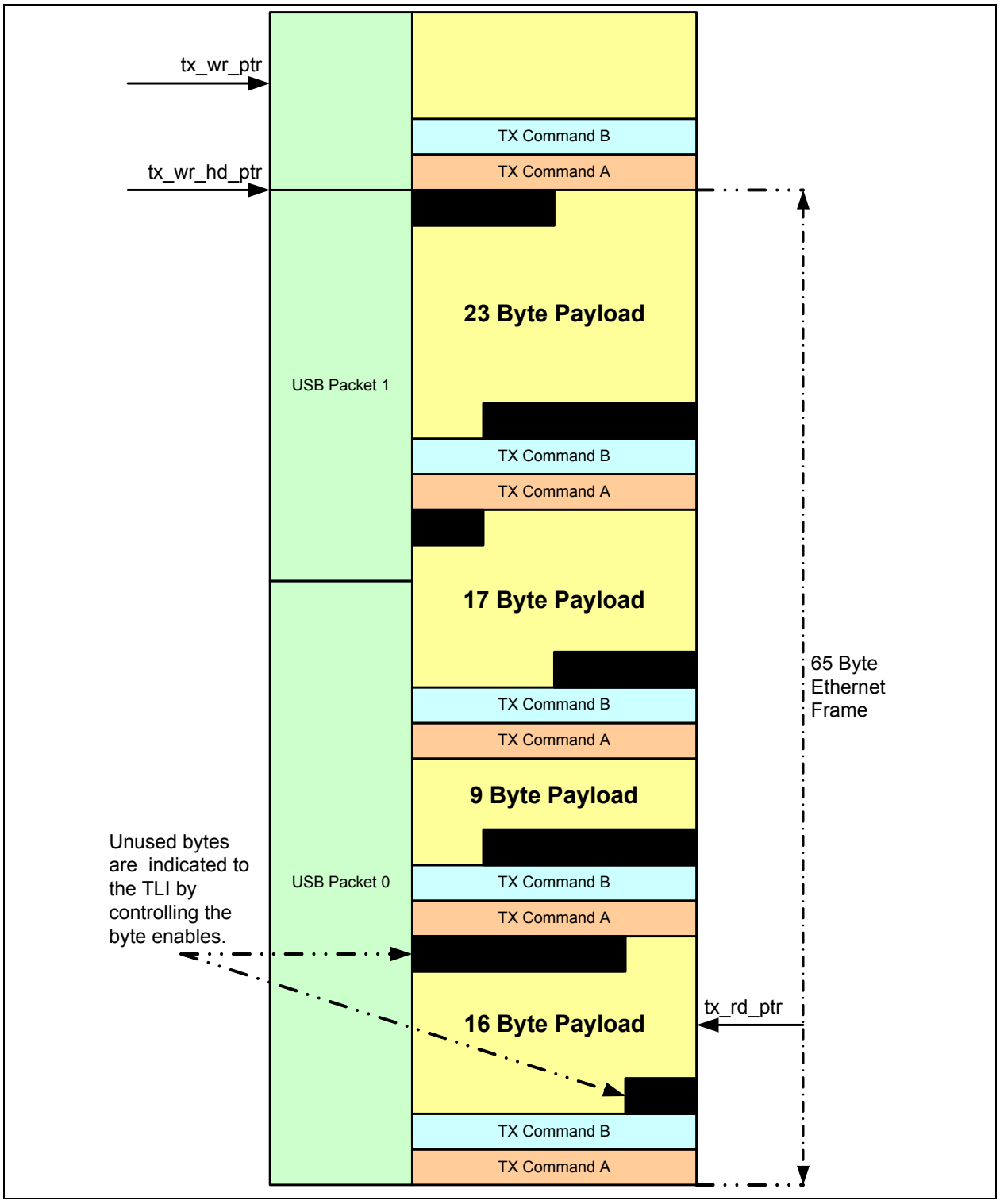
To stop the receiver, the host must clear the Receiver Enable (RXEN) bit in the MAC Control Register (MAC_CR) on page 163. When the receiver is halted, the RXSTOP_INT will be pulsed. Once stopped, the host can optionally clear the RX Status and RX FIFOs. The host must re-enable the receiver by setting the RXEN bit.

4.4.2 TX PATH (USB -> ETHERNET)

The 8 kB TX FIFO buffers USB Bulk-Out packets received by the URX. The FCT is responsible for extracting the Ethernet frames embedded in the USB Bulk-Out packets and passing them to the TLI. The Ethernet frames are segmented across the USB packets by the host drivers.

The FCT manages the writing of data into the TX FIFO through the use of two pointers - the tx_wr_ptr and the tx_wr_hd_ptr. These pointers are used to manage the storing of USB Bulk-Out packets. They support rewinding the stored USB packet, in the event that the Bulk-Out packet is errored and needs to be retransmitted by the host. The write side of the FCT does not perform any processing on the USB packet data. The read side of the TX FIFO is responsible for extracting the Ethernet frames. The Ethernet frames may be split across multiple buffers, as shown in Figure 4-5.

FIGURE 4-5: TX FIFO STORAGE



4.4.2.1 TX Command Format

As shown in Figure 4-5, each buffer starts with a two DWORD TX Command. The TX Command instructs the FCT on the handling of the associated buffer. The command precedes the data to be transmitted. The TX Command is divided into two, 32-bit words; TX Command A and TX Command B.

Both TX Command A and TX Command B are required for each buffer in a given packet. TX Command B must be identical for every buffer in a given packet, with the exception of the TX Checksum Enable (CK) bit. If the TX Command B DWORDs do not match, the FCT will assert the Transmitter Error (TXE) flag.

Frame boundaries are delineated using control bits within the TX Command. The Frame Length field in TX Command B specifies the number of bytes in the associated frame. All Frame Length fields must have the same value for all buffers in a given Frame. Hardware compares the Frame Length field and the actual amount of data received. If the actual frame length count does not match the Frame Length field, an error has occurred.

The formats of TX Command A and TX Command B are shown in Table 4-41 and Table 4-42, respectively.

TABLE 4-41: TX COMMAND A FORMAT

Bits	Description
31:18	RESERVED
17:16	Data Start Offset (bytes) This field specifies the offset of the first byte of TX Data. The offset value ranges between 0 bytes and 3 bytes.
15:14	RESERVED
13	First Segment When set, this bit indicates that the associated buffer is the first segment of the frame.
12	Last Segment When set, this bit indicates that the associated buffer is the last segment of the frame.
11	RESERVED
10:0	Buffer Size (bytes) This field indicates the number of bytes contained in the buffer following the two command DWORDS (TX Command A and TX Command B). This value, along with the Data Start Offset field, is used by the FCT to determine how many extra bytes were added to the end of the Buffer. A running count is also maintained in the FCT of the cumulative buffer sizes for a given frame. This cumulative value is compared against the Frame Length field in the TX Command B Word and if they do not correlate, the TXE flag is set. The buffer size specified does not include bytes added due to the end of buffer alignment padding or the Data Start Offset field.

TABLE 4-42: TX COMMAND B FORMAT

Bits	Description
31:15	RESERVED
14	**TX Checksum Enable (CK)** If this bit is set in conjunction with the first segment bit (FS) in TX Command A and the TX Checksum Offload Engine Enable bit (TXCOE_EN) in the Checksum Offload Engine Control register (COE_CR), the TX checksum offload engine (TXCOE) will calculate an L3 checksum for the associated frame. **Note:** This bit only needs to be set for the first buffer of a frame.
13	Add CRC Disable When set, the automatic addition of the CRC is disabled.
12	Disable Ethernet Frame Padding When set, this bit prevents the automatic addition of padding to an Ethernet frame of less than 64 bytes. The CRC field is also added despite the state of the Add CRC Disable field.
11	RESERVED
10:0	Frame Length (bytes) This field indicates the total number of bytes in the current frame. This length does not include the offset or padding. If the Frame Length field does not match the actual number of bytes in the frame, the Transmitter Error (TXE) flag will be set (in the Interrupt Status Register (INT_STS) and the interrupt Endpoint). This value is read by the TX FIFO controller, and is used to determine the amount of data that must be moved from the TX data FIFO into the TLI block. If the byte count is not aligned to a DWORD boundary, the TX FIFO Controller will issue the correct byte enables to the TLI layer during the last write. Invalid bytes in the last DWORD will not be passed to the TLI for transmission.

4.4.2.2 TX Data Format

The TX data section begins at the third DWORD in the TX buffer (after TX Command A and TX Command B). The location of the first byte of valid buffer data to be transmitted is specified in the Data Start Offset field of TX Command A. Table 4-43, "TX Data Start Offset" shows the correlation between the setting of the LSBs in the Data Start Offset field and the byte location of the first valid data byte.

TABLE 4-43: TX DATA START OFFSET

Data Start Offset[1:0]	11	10	01	00
First TX Data Byte	D[31:24]	D[23:16]	D[15:8]	D[7:0]

TX data is contiguous until the end of the buffer. The buffer may end on a byte boundary. Unused bytes at the end of the packet will not be sent to the TLI for transmission.

4.4.2.3 TX Buffer Fragmentation Rules

Transmit buffers must adhere to the following rules:

- Each buffer may start and end on any arbitrary byte alignment.
- The first buffer of any transmit packet can be any length.
- Middle buffers (i.e., those with First Segment = Last Segment = 0) must be greater than, or equal to 4 bytes in length.
- The final buffer of any transmit packet can be any length.

4.4.2.4 FCT Actions

The FCT performs basic sanity checks on the correctness of the buffer configuration, as described in Section 4.4.2.5, "TX Error Detection". Errors in this regard indicate the TX path is out of sync, which is catastrophic and requires a reinitialization of the TX path.

The FCT performs the following steps when extracting an Ethernet frame:

- Strip out TX Command A
- Strip out TX Command B
- Account for the byte offset at the beginning of the frame. Based upon the buffer size and DataStartOffset[1:0] field of TX Command A, the FCT can numerically determine any unused bytes in the first and last word of the buffer. When transferring these respective DWORDs to the TLI, the FCT adjusts the byte enables accordingly.

Note:	When a packet is split into multiple buffers, each successive buffer's data payload may begin on any arbitrary byte.

Unlike the write side, the read side of the TX FIFO does not need to support rewinds. Errors are reported via the Transmitter Error (TXE) flag, which is visible to the host via the Interrupt Endpoint and is also set in the Interrupt Status Register (INT_STS).

4.4.2.5 TX Error Detection

As previously stated, both TX Command A and TX Command B are required for each buffer in a given frame. TX Command B must be identical for every buffer in a given frame, with the exception of the TX Checksum Enable (CK) bit. If the TX Command B words do not match, then the TX path is out of sync and the FCT asserts the Transmitter Error (TXE) flag.

Similarly, the FCT numerically adds up the size of the frame's buffers. If there is a numerical mismatch, the TX path is out of sync and the FCT asserts the Transmitter Error (TXE) flag. The following error conditions are tracked by the FCT:

- Missing FS - The expected first buffer of a frame does not have the FS bit set.
- Unexpected FS - The FS bit is set when the total size of buffers so far opened is less than the frame size.
- Missing LS - The total size of the buffers opened is equal to or exceeds the size of the frame. The FCT expects this buffer to have the LS bit set and it is not set.
- Unexpected LS - The LS bit is set when the aggregate total size of descriptor buffers so far opened is less than the frame size.
- Buffer Size is Zero Error - The buffer length field is zero.
- Buffer Size Error - The total sum of the buffers received is not equal to the frame length.

Note:	The FCT can be configured to stall the Bulk-Out pipe when a Transmit Error is detected. This is accomplished via the Stall Bulk-Out Pipe Disable (SBP) bit of the Hardware Configuration Register (HW_CFG). Refer to Section 6.3.5, "Hardware Configuration Register (HW_CFG)" for further details.

Note:	A TX Error is a catastrophic condition. The device should be reset in order to recover from it.

4.4.2.6 TX Status Format

After an Ethernet frame is transmitted, the TLI returns the TX Status Word to the FCT, as illustrated in Table 4-44. The contents of the TX Status Word is used for statistics generation and interrupt status creation. Refer to Section 4.3.1.7, "Statistics" and Section 6.3.2, "Interrupt Status Register (INT_STS)" for further details.

TABLE 4-44: TX STATUS WORD FORMAT

Bits	Description
31:16	RESERVED
15	Error Status (ES) When set, this bit indicates that the TLI has reported an error. This bit is the logical OR of bits 11, 10, 9, 8, 2, 1 in this status word.
14:12	RESERVED
11	Loss of Carrier When set, this bit indicates the loss of carrier during transmission.
10	No Carrier When set, this bit indicates that the carrier signal from the transceiver was not present during transmission.
9	Late Collision When set, indicates that the packet transmission was aborted after the collision window of 64 bytes.
8	Excessive Collisions When set, this bit indicates that the transmission was aborted after 16 collisions while attempting to transmit the current packet.
7	RESERVED
6:3	Collision Count This counter indicates the number of collisions that occurred before the packet was transmitted. It is not valid when Excessive Collisions (bit 8) is also set.
2	Excessive Deferral If the deferred bit is set in the Control register, the setting of the Excessive Deferral bit indicates that the transmission has ended because of a deferral of over 24288 bit times during transmission.
1	Underrun Error When set, this bit indicates that the transmitter aborted the associated frame because of an underrun condition on the TX Data FIFO. TX Underrun will cause the assertion of the TDFU flag in the Interrupt Status Register (INT_STS) and the interrupt Endpoint.
0	Deferred When set, this bit indicates that the current packet transmission was deferred.

4.4.2.7 Transmit Examples

4.4.2.7.1 TX Example 1

In this example a single, 1064-byte Ethernet frame will be transmitted. This packet is divided into three buffers. The three buffers are as follows:

Buffer 0:

- 3 bytes Data Start Offset
- 499 bytes of payload data

Buffer 1:

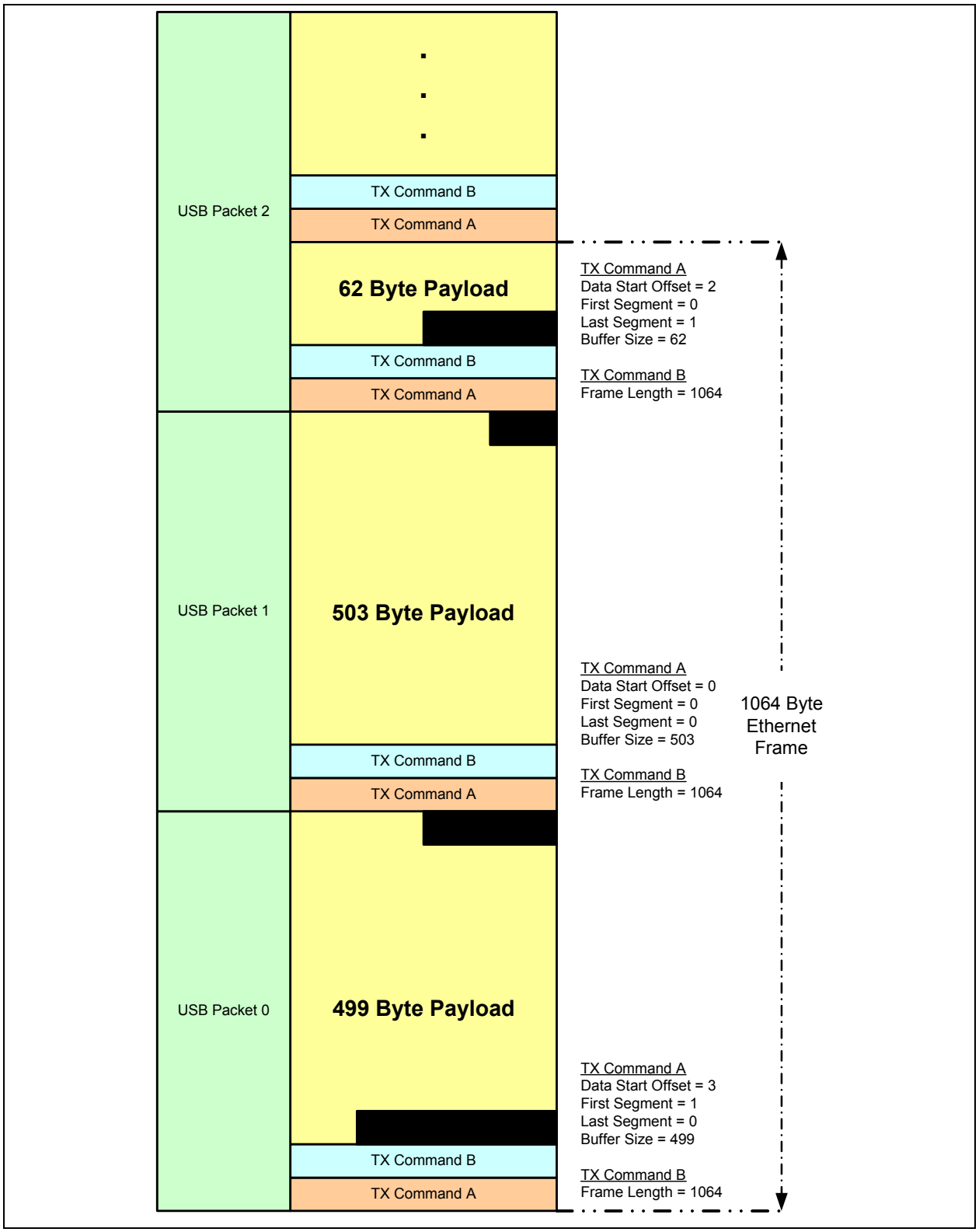
- 0 byte Data Start Offset
- 503 bytes of payload data

Buffer 2:

- 2 bytes Data Start Offset
- 62 bytes of payload data

Table 4-6 illustrates the TX Command structure for this example, and also shows how data is passed to the TX data FIFO.

FIGURE 4-6: TX EXAMPLE 1



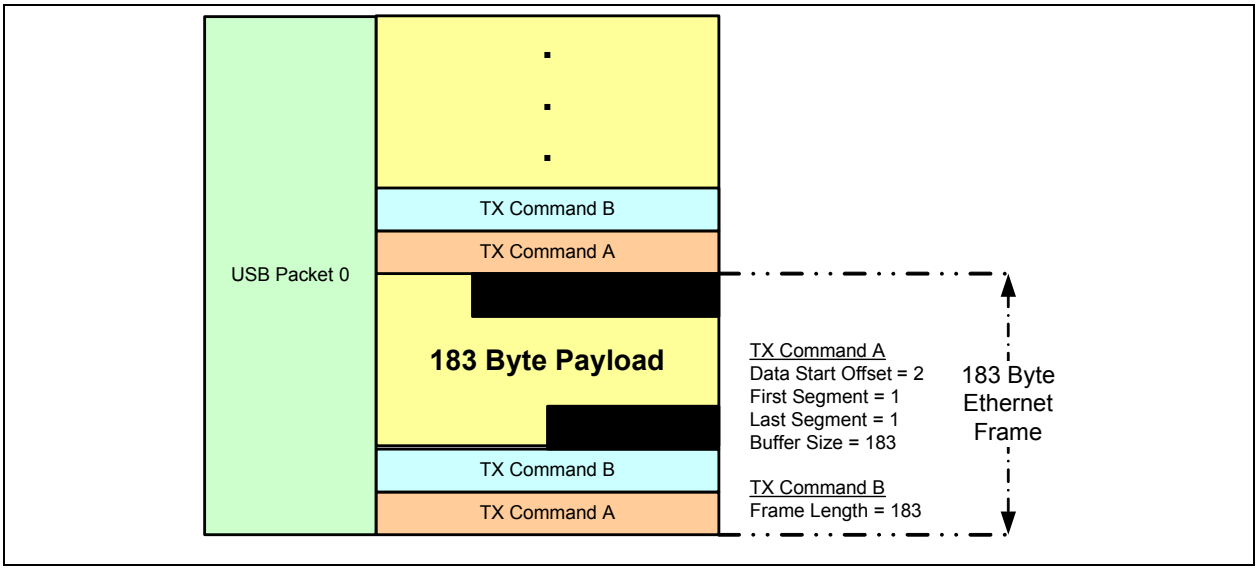
4.4.2.8 TX Example 2

In this example, a single 183-byte Ethernet frame will be transmitted. This packet is in a single buffer as follows:

- 2 bytes Data Start Offset
- 183 bytes of payload data

Figure 4-7 illustrates the TX Command structure for this example, and also shows how data is passed to the TX data FIFO. Note that the packet resides in a single TX Buffer, therefore both the FS and LS bits are set in TX Command A.

FIGURE 4-7: TX EXAMPLE 2



4.4.2.9 TX Example 3

In this example a single, 111-byte Ethernet frame will be transmitted with a TX checksum. This packet is divided into four buffers. The four buffers are as follows:

Buffer 0:

- 0 byte Data Start Offset
- 4 bytes Checksum Preamble

Buffer 1:

- 3 bytes Data Start Offset
- 79 bytes of payload data

Buffer 2:

- 0 byte Data Start Offset
- 15 bytes of payload data

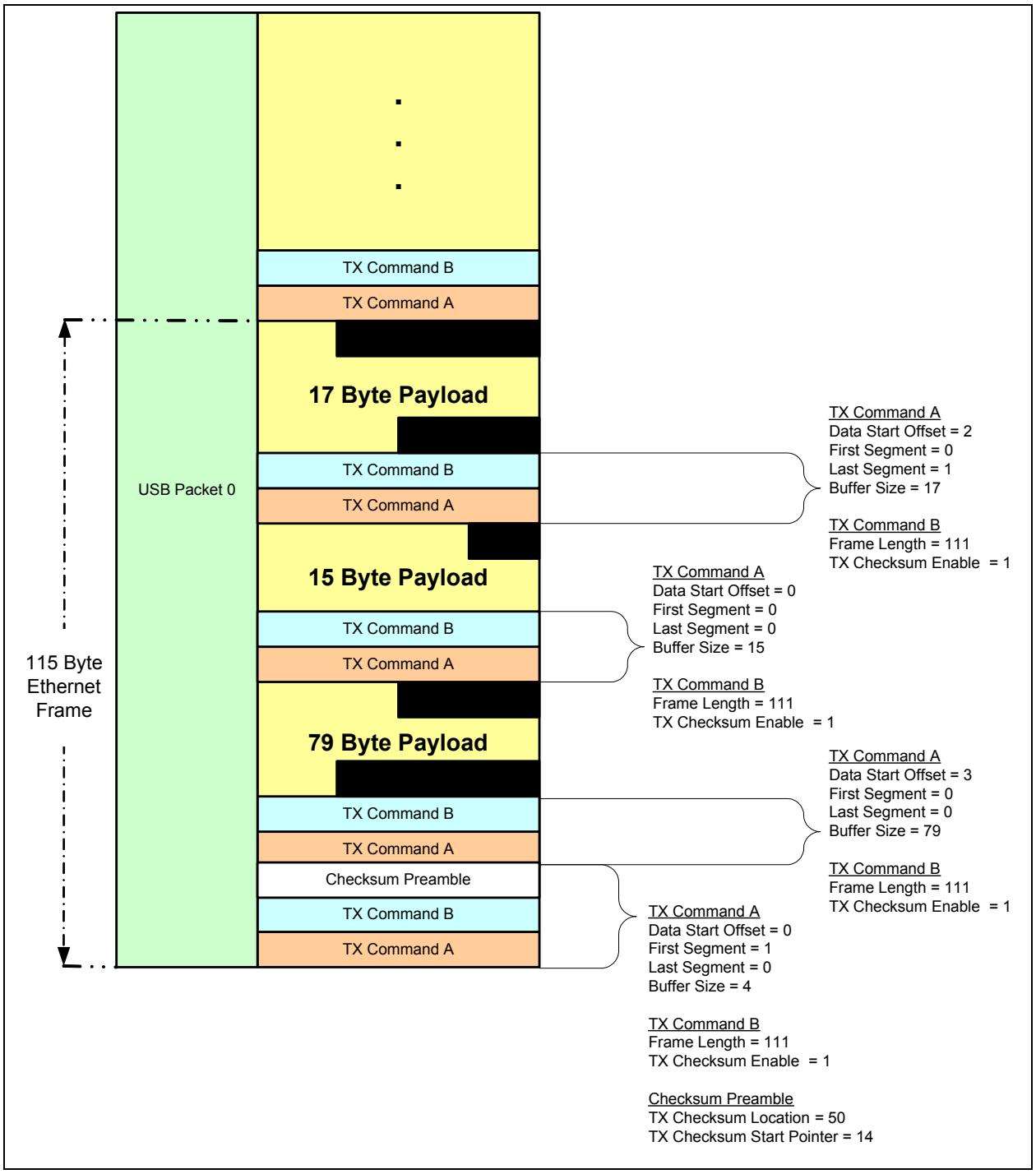
Buffer 3:

- 2 bytes Data Start Offset
- 17 bytes of payload data

Figure 4-8 illustrates the TX Command structure for this example, and also shows how data is passed to the TX data FIFO.

Note:	When enabled, the TX Checksum Preamble is pre-pended to the data to be transmitted. The FS bit in TX Command A, the TX Checksum Enable bit (CK) of TX Command B, and the TXCOE_EN bit of the COE_CR register must all be set for the TX checksum to be generated. FS must not be set for subsequent fragments of the same packet. Refer to Section 4.5.8, "Transmit Checksum Offload Engine (TXCOE)" for further information.

FIGURE 4-8: TX EXAMPLE 3



4.4.2.10 Flushing the TX FIFO

The device allows for the host to flush the entire contents of the FCT TX FIFO. When a flush is activated, the read and write pointers for the TX FIFO are returned to their reset state.

Before flushing the TX FIFO, the device's transmitter must be stopped, as specified in Section 4.4.2.11. Once the transmitter stop completion is confirmed, the Transmit FIFO Flush bit can be set in the Transmit Configuration Register (TX_CFG) on page 124. This bit is cleared after the flush is complete.

4.4.2.11 Stopping and Starting the Transmitter

To halt the transmitter, the host must set the Stop Transmitter (STOP_TX) bit in the TX_CFG register. The transmitter will finish sending the current frame (if there is a frame transmission in progress). When the transmitter has received the TX Status for the current frame, it will clear the STOP_TX and TX_ON bits in the TX_CFG register, and will pulse TXSTOP_INT.

Once stopped, the host can optionally flush the TX FIFO, and can optionally disable the MAC by clearing TXEN. The host must re-enable the transmitter by setting the TX_ON and TXEN bits. If there are frames pending in the TX FIFO (i.e., the TX FIFO was not purged), the transmission will resume with this data.

Note:	The TX Stop mechanism described here assumes that the MAC will return a status for every TX frame.

4.4.3 ARBITRATION

The FCT must arbitrate access to the RX and TX FIFOs to the URX, UTX, TLI RX, and TLI TX. Highest priority is always given to the USB. The TLI RX/TX can be wait stated as frames buffering exists in the TLI (2 kB TX, 128 byte RX).

FCT strict priority order:

1. URX Request (Bulk-Out packet)
2. UTX Request (Bulk-In packet)
3. TLI RX (received Ethernet frame)
4. TLI TX (transmitted Ethernet frame)

Note:	By nature of the USB bus and UDC operation, the URX and UTX do not request bandwidth simultaneously.

4.5 10/100 Ethernet MAC

The Ethernet Media Access Controller (MAC) incorporates the essential protocol requirements for operating an Ethernet/IEEE 802.3-compliant node and provides an interface between the host subsystem and the internal Ethernet PHY. The MAC can operate in either 100 Mbps or 10 Mbps mode.

The MAC operates in both half-duplex and full-duplex modes. When operating in half-duplex mode, the MAC complies fully with Section 4 of ISO/IEC 8802-3 (ANSI/IEEE standard) and ANSI/IEEE 802.3 standards. When operating in full-duplex mode, the MAC complies with IEEE 802.3x full-duplex operation standard.

The MAC provides programmable enhanced features designed to minimize host supervision, bus utilization, and pre- or post-message processing. These features include the ability to disable retries after a collision, dynamic FCS (Frame Check Sequence) generation on a frame-by-frame basis, automatic pad field insertion and deletion to enforce minimum frame size attributes, layer 3 checksum calculation for transmit and receive operations, and automatic retransmission and detection of collision frames.

The MAC can sustain transmission or reception of minimally-sized back-to-back packets at full line speed with an inter-packet gap (IPG) of 9.6 microseconds for 10 Mbps and 0.96 microseconds for 100 Mbps.

The primary attributes of the MAC function are:

- Transmit and receive message data encapsulation
- Framing (frame boundary delimitation, frame synchronization)
- Error detection (physical medium transmission errors)
- Media access management
- Medium allocation (collision detection, except in full-duplex operation)
- Contention resolution (collision handling, except in full-duplex operation)
- Flow control during full-duplex mode
- Decoding of control frames (PAUSE command) and disabling the transmitter
- Generation of control frames
- Interface to the internal PHY and optional external PHY
- Checksum offload engine for calculation of layer 3 transmit and receive checksum

The transmit and receive data paths are separate within the device from the MAC to host interface, allowing the highest performance, especially in full-duplex mode. Payload data as well as transmit and receive status are passed on these buses.

A third internal bus is used to access the MAC's Control and Status Registers (CSR's). This bus is also accessible from the host.

On the backend, the MAC interfaces with the 10/100 PHY through an MII (Media Independent Interface) port which is internal to the device. In addition, there is an external MII interface supporting optional PHY devices. The MAC CSR's also provide a mechanism for accessing the PHY's internal registers through the internal SMI (Serial Management Interface) bus.

The receive and transmit FIFOs allow increased packet buffer storage to the MAC. The FIFOs are a conduit between the host interface and the MAC through which all transmitted and received data and status information is passed. Deep FIFOs allow a high degree of latency tolerance relative to the various transport and OS software stacks reducing and minimizing overrun conditions. Like the MAC, the FIFOs have separate receive and transmit data paths.

4.5.1 FLOW CONTROL

The device's Ethernet MAC supports full-duplex flow control using the pause operation and control frame. It also supports half-duplex flow control using back pressure. In order for flow control to be invoked, the Flow Control Enable (FCEN) bit of the Flow Control Register (FLOW) must be set.

4.5.1.1 Full-Duplex Flow Control

The pause operation inhibits data transmission of data frames for a specified period of time. A Pause operation consists of a frame containing the globally assigned multicast address (01-80-C2-00-00-01), the PAUSE opcode, and a parameter indicating the quantum of slot time (512 bit times) to inhibit data transmissions. The PAUSE parameter may range from 0 to 65,535 slot times. The Ethernet MAC logic, on receiving a frame with the reserved multicast address and PAUSE opcode, inhibits data frame transmissions for the length of time indicated. If a Pause request is received while a transmission is in progress, then the pause will take effect after the transmission is complete. Control frames are received and processed by the MAC and are passed on.

The device will automatically transmit pause frames based on the settings of the Automatic Flow Control Configuration Register (AFC_CFG) and the Flow Control Register (FLOW). When the RX FIFO reaches the level set in the Automatic Flow Control High Level (AFC_HI) field of AFC_CFG, the device will transmit a pause frame. The pause time field that is transmitted is set in the Pause Time (FCPT) field of the FLOW register. When the RX FIFO drops below the level set in the Automatic Flow Control Low Level (AFC_LO) field of AFC_CFG, the device will automatically transmit a pause frame with a pause time of zero. The device will only send another pause frame when the RX FIFO level falls below AFC_LO and then exceeds AFC_HI again.

4.5.1.2 Half-Duplex Flow Control (Backpressure)

In half-duplex mode, back pressure is used for flow control. Whenever the RX FIFO crosses a certain threshold level, the MAC starts sending a jam signal. The MAC transmit logic enters a state at the end of current transmission (if any), where it waits for the beginning of a received frame. Once a new frame starts, the MAC starts sending the jam signal, which will result in a collision. After sensing the collision, the remote station will back off its transmission. The MAC continues sending the jam signal to make other stations defer transmission. The MAC only generates this collision-based back pressure when it receives a new frame, in order to avoid any late collisions.

The device will automatically assert back pressure based on the setting of the Automatic Flow Control Configuration Register (AFC_CFG). When the RX FIFO reaches the level set by Automatic Flow Control High Level (AFC_HI) field of AFC_CFG, the Back Pressure Duration Timer will start. The device will assert back pressure for any received frames, as defined by the values of the FCANY, FCADD, FCMULT and FCBRD control bits of AFC_CFG. This continues until the Back Pressure Duration Timer reaches the time specified by the BACK_DUR field of AFC_CFG. After the BACK_DUR time period has elapsed, the receiver will accept one frame. If, after receiving one RX frame, the RX FIFO is still above the threshold set in the Automatic Flow Control Low Level (AFC_LO) field of AFC_CFG, the device will again start the Back Pressure Duration Timer and will assert back pressure for subsequent frames, repeating the process described here until the RX Data FIFO level drops below the AFC_LO setting. If the RX FIFO drops below AFC_LO before the Back Pressure Duration Timer has expired, the timer will immediately reset and back pressure will not be asserted until the RX FIFO level exceeds AFC_HI.

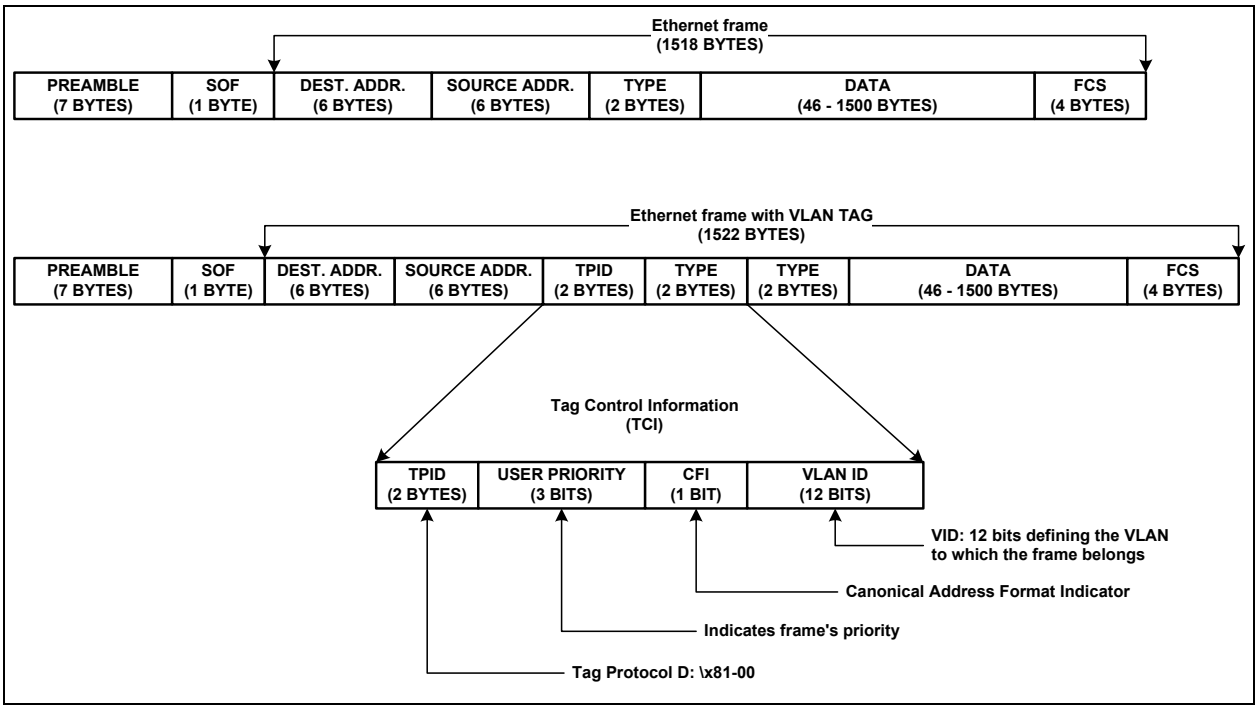
If the AFC_LO value is set to all ones (0xFF) and the AFC_HI value is set to all zeros (0x00), the flow controller will assert back pressure for received frames as if the AFC_HI threshold is always exceeded. This mechanism can be used to generate software-controlled flow control by enabling and disabling the FCANY, FCADD, FCMULT and FCBRD bits.

4.5.2 VIRTUAL LOCAL AREA NETWORK (VLAN) SUPPORT

Virtual Local Area Networks or VLANs, as defined within the IEEE 802.3 standard, provide network administrators one means of grouping nodes within a larger network into broadcast domains. To implement a VLAN, four extra bytes are added to the basic Ethernet packet. As shown in Figure 4-9, the four bytes are inserted after the Source Address Field and before the Type/Length field. The first two bytes of the VLAN tag identify the tag, and by convention are set to the value 0x8100. The last two bytes identify the specific VLAN associated with the packet; they also provide a priority field.

The device supports VLAN-tagged packets. It provides two registers which are used to identify VLAN-tagged packets. One register should normally be set to the conventional VLAN ID of 0x8100. The other register provides a way of identifying VLAN frames tagged with a proprietary (not 0x8100) identifier. If a packet arrives bearing either of these tags in the two bytes succeeding the Source Address field, the controller will recognize the packet as a VLAN-tagged packet. In this case, the controller increases the maximum allowed packet size from 1518 to 1522 bytes (normally the controller filters packets larger than 1518 bytes). This allows the packet to be received, and then processed by host software, or to be transmitted on the network.

FIGURE 4-9: VLAN FRAME



4.5.3 ADDRESS FILTERING FUNCTIONAL DESCRIPTION

The Ethernet address fields of an Ethernet packet, consists of two 6-byte fields: one for the destination address and one for the source address. The first bit of the destination address signifies whether it is a physical address or a multicast address.

The device's address check logic filters the frame based on the Ethernet receive filter mode that has been enabled. Filter modes are specified based on the state of the control bits in Table 4-45, "Address Filtering Modes", which shows the various filtering modes used by the Ethernet MAC function. These bits are defined in more detail in the MAC Control Register. Refer to Section 6.4.1, "MAC Control Register (MAC_CR)" for more information on this register.

If the frame fails the filter, the Ethernet MAC function does not receive the packet. The host has the option of accepting or ignoring the packet.

TABLE 4-45: ADDRESS FILTERING MODES

MCPAS	PRMS	INVFILT	HO	HPFILT	Description
0	0	0	0	0	MAC address Perfect Filtering only for all addresses.
0	0	0	0	1	MAC address Perfect Filtering for physical address and hash filtering for multicast addresses.
0	0	0	1	1	Hash Filtering for physical and multicast addresses.
0	0	1	0	0	Inverse Filtering
X	1	0	X	X	Promiscuous
1	0	0	0	X	Pass all multicast frames. Frames with physical addresses are perfect-filtered.
1	0	0	1	1	Pass all multicast frames. Frames with physical addresses are hash-filtered.

4.5.4 FILTERING MODES

4.5.4.1 Perfect Filtering

This filtering mode passes only incoming frames whose destination address field exactly matches the value programmed into the MAC Address High register and the MAC address low register. The MAC address is formed by the concatenation of the above two registers in the MAC CSR function.

4.5.4.2 Hash Only Filtering Mode

This type of filtering checks for incoming receive packets with either multicast or physical destination addresses, and executes an imperfect address filtering against the hash table.

During imperfect hash filtering, the destination address in the incoming frame is passed through the CRC logic and the upper six bits of the CRC register are used to index the contents of the hash table. The hash table is formed by merging the register's multicast hash table high and multicast hash table low in the MAC CSR function to form a 64-bit hash table. The most significant bit determines the register to be used (High/Low), while the other five bits determine the bit within the register. A value of 00000 selects Bit 0 of the multicast hash table low register and a value of 11111 selects Bit 31 of the multicast hash table high register.

4.5.4.3 Hash Perfect Filtering

In hash Perfect Filtering, if the received frame is a physical address, the device's Packet Filter Block perfect filters the incoming frame's destination field with the value programmed into the MAC Address High register and the MAC Address Low register. If the incoming frame is a multicast frame, however, the device's packet filter function performs an imperfect address filtering against the hash table.

The imperfect filtering against the hash table is the same imperfect filtering process described in Section 4.5.4.2, "Hash Only Filtering Mode".

4.5.4.4 Inverse Filtering

In Inverse Filtering, the Packet Filter Block accepts incoming frames with a destination address not matching the perfect address (i.e., the value programmed into the MAC Address High register and the MAC Address Low register in the CRC block) and rejects frames with destination addresses matching the perfect address.

For all filtering modes, when MCPAS is set, all multicast frames are accepted. When the PRMS bit is set, all frames are accepted regardless of their destination address. This includes all broadcast frames as well.

4.5.5 WAKEUP FRAME DETECTION

Setting the Wakeup Frame Enable (WUEN) bit in the Wakeup Control and Status Register (WUCSR), places the MAC in the Wakeup Frame Detection mode. In this mode, normal data reception is disabled, and detection logic within the MAC examines receive data for the pre-programmed Wakeup Frame patterns. When a wakeup pattern is received, the Remote Wakeup Frame Received (WUFR) bit in the WUCSR is set, the device places itself in a fully operational state, and remote wakeup is issued. The host will then resume the device and read the WUSCR register to determine the condition that caused the remote wakeup. Upon determining that the WUFR bit is set, the host will know a Wakeup Frame detection event was the cause. The host will then clear the WUFR bit, and clear the WUEN bit to resume normal receive operation. Refer to Section 6.4.12, "Wakeup Control and Status Register (WUCSR)" for additional information on this register.

Before putting the MAC into the Wakeup Frame detection state, the host must provide the detection logic with a list of sample frames and their corresponding byte masks. This information is written into the Wakeup Frame Filter register (WUFF). Refer to Section 6.4.11, "Wakeup Frame Filter (WUFF)" for additional information on this register.

Table 4-46 indicates the number of Wakeup Frame Filters contained in the WUFF.The number of writes/reads required to program the WUFF or read its contents, respectively, is also indicated.

TABLE 4-46: WAKEUP FRAME FILTER CAPACITY

Number of Filters	Number of Writes/Reads
8	40

The programmable filters support many different receive packet patterns. If remote wakeup mode is enabled, the remote wakeup function receives all frames addressed to the MAC. It then checks each frame against the enabled filter and recognizes the frame as a remote Wakeup Frame if it passes the WUFF's address filtering and CRC value match.

In order to determine which bytes of the frames should be checked by the CRC module, the MAC uses a programmable byte mask and a programmable pattern offset for each of the supported filters.

The pattern's offset defines the location of the first byte that should be checked in the frame. The byte mask is a 128-bit field that specifies whether or not each of the 128 contiguous bytes within the frame, beginning in the pattern offset, should be checked. If bit j in the byte mask is set, the detection logic checks byte offset +j in the frame.

In order to load the Wakeup Frame Filter register, the host LAN driver software must perform the number of writes indicated in Table 4-46 to the device's Wakeup Frame Filter register (WUFF). The contents of the Wakeup Frame Filter register may be obtained by reading it. The number of reads required to extract the entire contents of the device's WUFF is also indicated in Table 4-46.

Table 4-47 shows the Wakeup Frame Filter register's structure.

TABLE 4-47: WAKEUP FRAME FILTER REGISTER STRUCTURE

Filter 0 Byte Mask 0							
Filter 0 Byte Mask 1							
Filter 0 Byte Mask 2							
Filter 0 Byte Mask 3							
Filter 1 Byte Mask 0							
Filter 1 Byte Mask 1							
Filter 1 Byte Mask 2							
Filter 1 Byte Mask 3							
Filter 2 Byte Mask 0							
Filter 2 Byte Mask 1							
Filter 2 Byte Mask 2							
Filter 2 Byte Mask 3							
Filter 3 Byte Mask 0							
Filter 3 Byte Mask 1							
Filter 3 Byte Mask 2							
Filter 3 Byte Mask 3							
Filter 4 Byte Mask 0							
Filter 4 Byte Mask 1							
Filter 4 Byte Mask 2							
Filter 4 Byte Mask 3							
Filter 5 Byte Mask 0							
Filter 5 Byte Mask 1							
Filter 5 Byte Mask 2							
Filter 5 Byte Mask 3							
Filter 6 Byte Mask 0							
Filter 6 Byte Mask 1							
Filter 6 Byte Mask 2							
Filter 6 Byte Mask 3							
Filter 7 Byte Mask 0							
Filter 7 Byte Mask 1							
Filter 7 Byte Mask 2							
Filter 7 Byte Mask 3							
Reserved	Filter 3 Command	Reserved	Filter 2 Command	Reserved	Filter 1 Command	Reserved	Filter 0 Command
Reserved	Filter 7 Command	Reserved	Filter 6 Command	Reserved	Filter 5 Command	Reserved	Filter 4 Command
Filter 3 Offset		Filter 2 Offset		Filter 1Offset		Filter 0 Offset	
Filter 7 Offset		Filter 6 Offset		Filter 5 Offset		Filter 4 Offset	
Filter 1 CRC-16				Filter 0 CRC-16			
Filter 3 CRC-16				Filter 2 CRC-16			
Filter 5 CRC-16				Filter 4 CRC-16			
Filter 7 CRC-16				Filter 6 CRC-16			

The Filter i Byte Mask defines which incoming frame bytes Filter i will examine to determine whether or not this is a Wakeup Frame. Table 4-48, describes the byte mask's bit fields.

Filter x Mask 0 corresponds to bits [31:0]. Where the lsb corresponds to the first byte on the wire.

Filter x Mask 1 corresponds to bits [63:32]. Where the lsb corresponds to the first byte on the wire.

Filter x Mask 2 corresponds to bits [95:64]. Where the lsb corresponds to the first byte on the wire.

Filter x Mask 3 corresponds to bits [127:96]. Where the lsb corresponds to the first byte on the wire.

The following tables define the WUFF register structures.

TABLE 4-48: FILTER I BYTE MASK BIT DEFINITIONS

Filter i Byte Mask Description	
Bits	**Description**
127:0	**Byte Mask:** If bit j of the byte mask is set, the CRC machine processes byte *pattern-offset + j* of the incoming frame. Otherwise, byte *pattern-offset + j* is ignored.

The Filter i command register controls Filter i operation. Table 4-49 shows the Filter i command register.

TABLE 4-49: FILTER I COMMAND BIT DEFINITIONS

Filter i Commands	
Bits	**Description**
3:2	Address Type: Defines the destination address type of the pattern. 00 = Pattern applies only to unicast frames. 10 = Pattern applies only to multicast frames. X1 = Pattern applies to all frames that have passed the regular receive filter.
1	RESERVED
0	Enable Filter: When bit is set, Filter i is enabled, otherwise, Filter i is disabled.

The Filter i Offset register defines the offset in the frame's destination address field from which the frames are examined by Filter i. Table 4-50 describes the Filter i Offset bit fields.

TABLE 4-50: FILTER I OFFSET BIT DEFINITIONS

Filter i Offset Description	
Bits	**Description**
7:0	**Pattern Offset:** The offset of the first byte in the frame on which CRC is checked for Wakeup Frame recognition. The MAC checks the first offset byte of the frame for CRC and checks to determine whether the frame is a Wakeup frame. Offset 0 is the first byte of the incoming frame's destination address.

The Filter i CRC-16 register contains the CRC-16 result of the frame that should pass Filter i.

Table 4-51 describes the Filter i CRC-16 bit fields.

TABLE 4-51: FILTER I CRC-16 BIT DEFINITIONS

Filter i CRC-16 Description	
Bits	**Description**
15:0	**Pattern CRC-16:** This field contains the 16-bit CRC value from the pattern and the byte mask programmed to the Wakeup Filter register function. This value is compared against the CRC calculated on the incoming frame, and a match indicates the reception of a Wakeup frame.

Table 4-52 indicates the cases that produce a wake when the Wakeup Frame Enable (WUEN) bit of the Wakeup Control and Status Register (WUCSR) is set. All other cases do not generate a wake.

TABLE 4-52: WAKEUP GENERATION CASES

Filter Enabled (Note 4-20)	CRC Match (Note 4-21)	Global Unicast Enabled (Note 4-22)	Pass Regular Receive Filter	Address Type (Note 4-23)	Broadcast Frame (Note 4-24)	Multicast Frame	Unicast Frame
Yes	Yes	x	x	x	Yes	No	No
Yes	Yes	Yes	x	x	No	No	Yes
Yes	Yes	x	Yes	Multicast (=10)	No	Yes	No
Yes	Yes	x	Yes	Unicast (=00)	No	No	Yes
Yes	Yes	x	Yes	Passed Receive Filter (=x1b)	x	x	x

Note 4-20 As determined by bit 0 of Filter i Command.

Note 4-21 CRC matches Filter i CRC-16 field.

Note 4-22 As determined by bit 9 of WUCSR.

Note 4-23 As determined by bits 3:2 of Filter i Command.

Note 4-24 When Wakeup frame detection is enabled via the Wakeup Frame Enable (WUEN) bit of the Wakeup Control and Status Register (WUCSR), a Broadcast Wakeup frame will wake up the device despite the state of the Disable Broadcast Frames (BCAST) bit in the MAC Control Register (MAC_CR).

Note: x indicates “don’t care”.

4.5.6 MAGIC PACKET DETECTION

Setting the Magic Packet Enable (MPEN) bit in the Wakeup Control and Status Register (WUCSR), places the MAC in the Magic Packet Detection mode. In this mode, normal data reception is disabled, and detection logic within the MAC examines receive data for a Magic Packet. When a Magic Packet is received, the Magic Packet Received (MPR) bit in the WUCSR is set, the device places itself in a fully operational state, and remote wakeup is issued. The host will then resume the device and read the WUSCR register to determine the condition that caused the remote wakeup. Upon determining that the MPR bit is set, the host will know reception of a Magic Packet was the cause. The host will then clear the MPR bit, and clear the MPEN bit to resume normal receive operation. Refer to Section 6.4.12, "Wakeup Control and Status Register (WUCSR)" for additional information on this register.

In Magic Packet mode, the Power Management Logic constantly monitors each frame addressed to the node for a specific Magic Packet pattern. It checks only packets with the MAC's address or a broadcast address to meet the Magic Packet requirement. The Power Management Logic checks each received frame for the pattern 48h FF_FF_FF_FF_FF_FF after the destination and source address field.

Then the function looks in the frame for 16 repetitions of the MAC address without any breaks or interruptions. In case of a break in the 16 address repetitions, the PMT function scans for the 48'h FF_FF_FF_FF_FF_FF pattern again in the incoming frame.

The 16 repetitions may be anywhere in the frame but must be preceded by the synchronization stream. The device will also accept a multicast frame, as long as it detects the 16 duplications of the MAC address. If the MAC address of a node is 00h 11h 22h 33h 44h 55h, then the MAC scans for the following data sequence in an Ethernet Frame:

Destination Address Source Address ……FF FF FF FF FF FF

00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55

00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55

00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55

00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55

…CRC

4.5.7 RECEIVE CHECKSUM OFFLOAD ENGINE (RXCOE)

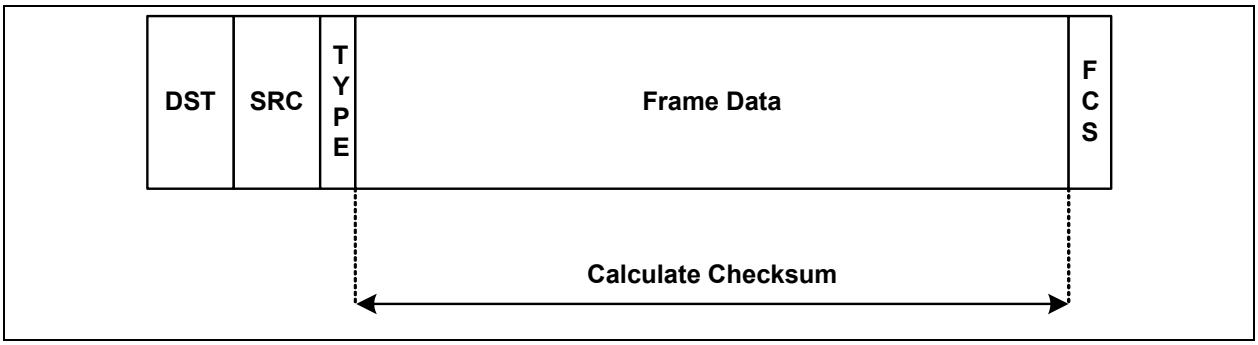
The receive checksum offload engine provides assistance to the host by calculating a 16-bit checksum for a received Ethernet frame. The RXCOE readily supports the following IEEE 802.3 frame formats:

- Type II Ethernet frames
- SNAP encapsulated frames
- Support for up to 2, 802.1q VLAN tags

The resulting checksum value can also be modified by software to support other frame formats.

The RXCOE has two modes of operation. In mode 0, the RXCOE calculates the checksum between the first 14 bytes of the Ethernet frame and the FCS. This is illustrated in Figure 4-10.

FIGURE 4-10: RXCOE CHECKSUM CALCULATION



In mode 1, the RXCOE supports VLAN tags and a SNAP header. In this mode, the RXCOE calculates the checksum at the start of L3 packet. The VLAN1 tag register is used by the RXCOE to indicate what protocol type is to be used to indicate the existence of a VLAN tag. This value is typically 8100h.

Example frame configurations:

FIGURE 4-11: TYPE II ETHERNET FRAME

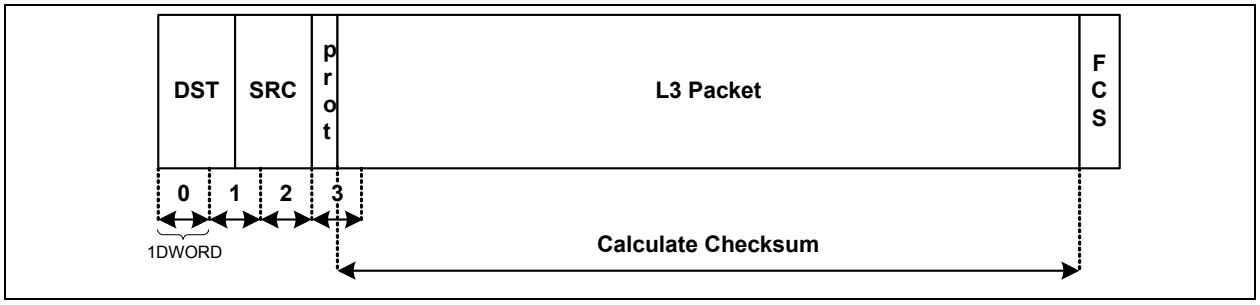


FIGURE 4-12: ETHERNET FRAME WITH VLAN TAG

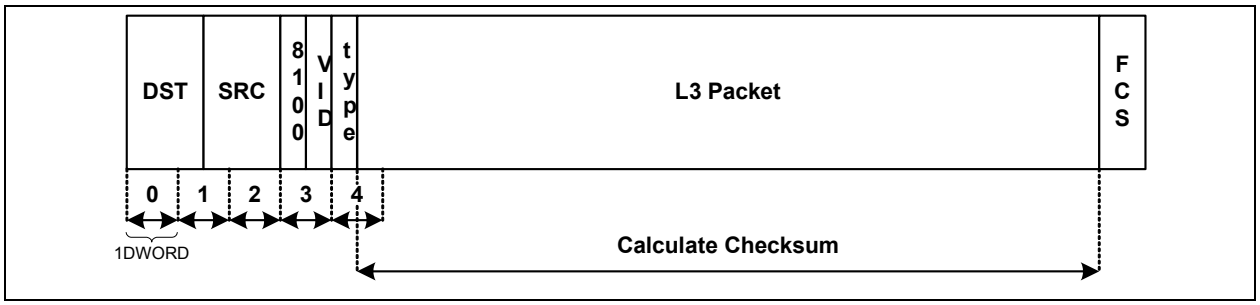


FIGURE 4-13: ETHERNET FRAME WITH LENGTH FIELD AND SNAP HEADER

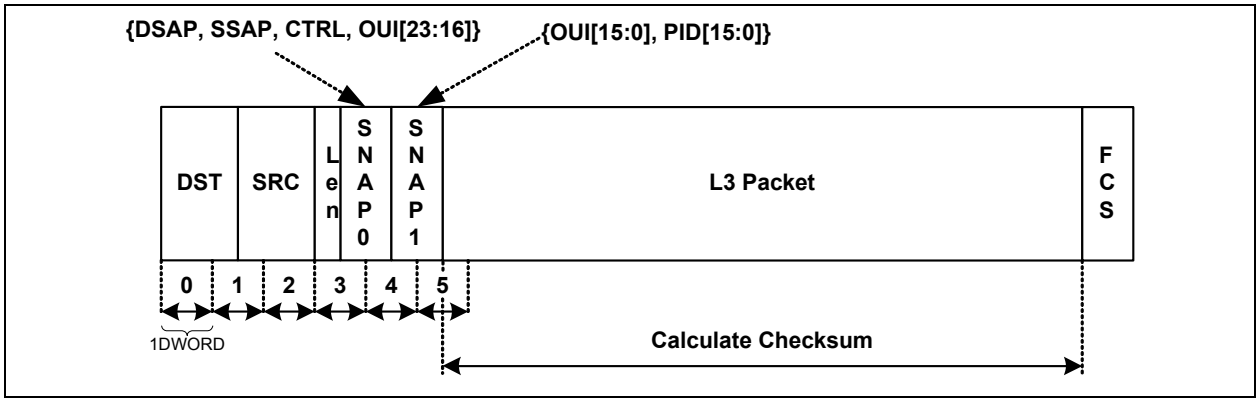


FIGURE 4-14: ETHERNET FRAME WITH VLAN TAG AND SNAP HEADER

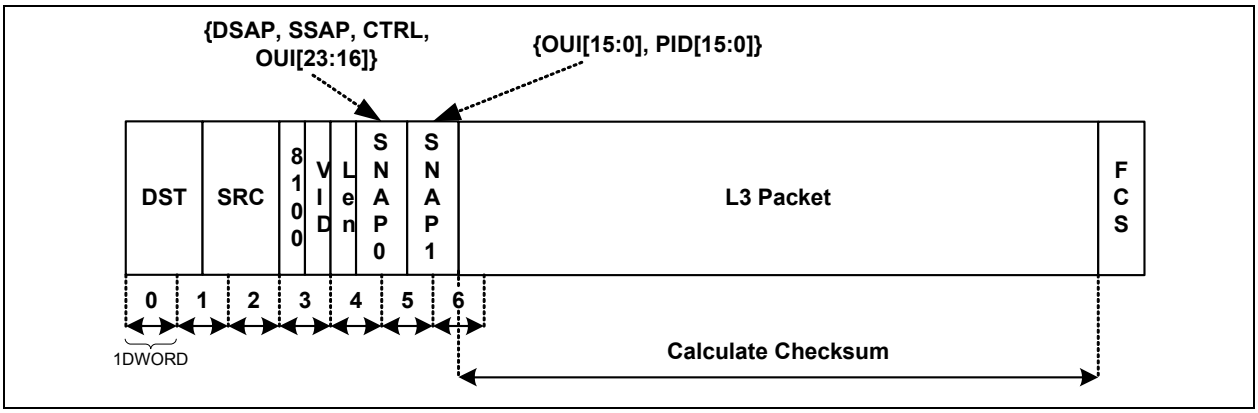
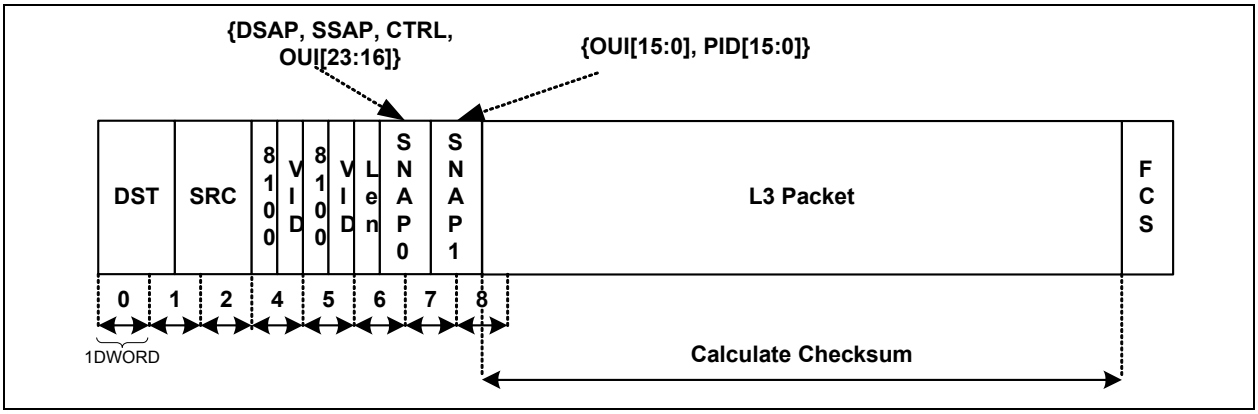


FIGURE 4-15: ETHERNET FRAME WITH MULTIPLE VLAN TAGS AND SNAP HEADER



The RXCOE supports a maximum of two VLAN tags. If there are more than two VLAN tags, the VLAN protocol identifier for the third tag is treated as an Ethernet type field. The checksum calculation will begin immediately after the type field.

The RXCOE resides in the RX path within the MAC. As the RXCOE receives an Ethernet frame, it calculates the 16-bit checksum. The RXCOE passes the Ethernet frame to the RX FIFO with the checksum appended to the end of the frame. The RXCOE inserts the checksum immediately after the last byte of the Ethernet frame and before it transmits the status word. The packet length field in the RX Status Word (refer to Section 4.4.1.2) will indicate that the frame size has increased by two bytes to accommodate the checksum.

Note:	When enabled, the RXCOE calculates a checksum for every received frame.

Setting the RXCOE_EN bit in the Checksum Offload Engine Control Register (COE_CR) enables the RXCOE, while the RXCOE_MODE bit selects the operating mode. When the RXCOE is disabled, the received data is simply passed through the RXCOE unmodified.

Note:	Software applications must stop the receiver and flush the RX data path before changing the state of the RXCOE_EN or RXCOE_MODE bits.

Note:	When the RXCOE is enabled, automatic pad stripping must be disabled (bit 8 (PADSTR) of the MAC Control Register (MAC_CR)) and vice versa. These functions cannot be enabled simultaneously.

4.5.7.1 RX Checksum Calculation

The checksum is calculated 16 bits at a time. In the case of an odd sized frame, an extra byte of zero is used to pad up to 16 bits.

Consider the following packet: DA, SA, Type, B0, B1, B2 … BN, FCS

Let [A, B] = A*256 + B

If the packet has an even number of octets then

checksum = [B1, B0] + C0 + [B3, B2] + C1 + … + [BN, BN-1] + CN-1

Where C0, C1, ... CN-1 are the carry out results of the intermediate sums.

If the packet has an odd number of octets then

checksum = [B1, B0] + C0 + [B3, B2] + C1 + … + [0, BN] + CN-1

4.5.8 TRANSMIT CHECKSUM OFFLOAD ENGINE (TXCOE)

The transmit checksum offload engine provides assistance to the CPU by calculating a 16-bit checksum, typically for TCP, for a transmit Ethernet frame. The TXCOE calculates the checksum and inserts the results back into the data stream as it is transferred to the MAC.

To activate the TXCOE and perform a checksum calculation, the host must first set the TX Checksum Offload Engine Enable (TX_COE_EN) bit in the Checksum Offload Engine Control Register (COE_CR). The host then pre-pends a 3 DWORD buffer to the data that will be transmitted. The pre-pended buffer includes a TX Command A, TX Command B, and a 32-bit TX checksum preamble (refer to Table 4-53). When the CK bit of the TX Command ‘B’ is set in conjunction with the FS bit of TX Command ‘A’ and the TX_COE_EN bit of the COE_CR register, the TXCOE will perform a checksum calculation on the associated packet. The TX checksum preamble instructs the TXCOE on the handling of the associated packet. The TXCSSP - TX Checksum Start Pointer field of the TX checksum preamble defines the byte offset at which the data checksum calculation will begin. The checksum calculation will begin at this offset and will continue until the end of the packet. The data checksum calculation must not begin in the MAC header (first 14 bytes) or in the last 4 bytes of the TX packet. When the calculation is complete, the checksum will be inserted into the packet at the byte offset defined by the TXCSLOC - TX Checksum Location field of the TX checksum preamble. The TX checksum cannot be inserted in the MAC header (first 14 bytes) or in the last 4 bytes of the TX packet. If the CK bit is not set in the first TX Command ‘B’ of a packet, the packet is passed directly through the TXCOE without modification, regardless if the TXCOE_EN is set. An example of a TX packet with a pre-pended TX checksum preamble can be found in Section 4.4.2.9, "TX Example 3". In this example, the host provides the Ethernet frame to the Ethernet controller (via a USB packet) in four fragments, the first containing the TX Checksum Preamble. Figure 4-8 shows how these fragments are loaded into the TX Data FIFO. For more information on the TX Command ‘A’ and TX Command ‘B’, refer to Section 4.4.2.1, "TX Command Format".

If the TX packet already includes a partial checksum calculation (perhaps inserted by an upper layer protocol), this checksum can be included in the hardware checksum calculation by setting the TXCSSP field in the TX checksum preamble to include the partial checksum. The partial checksum can be replaced by the completed checksum calculation by setting the TXCSLOC pointer to point to the location of the partial checksum.

TABLE 4-53: TX CHECKSUM PREAMBLE

Field	Description
31:28	**RESERVED**
27:16	**TXCSLOC - TX Checksum Location** This field specifies the byte offset where the TX checksum will be inserted in the TX packet. The checksum will replace two bytes of data starting at this offset. **Note:** The TX checksum cannot be inserted in the MAC header (first 14 bytes) or in the last 4 bytes of the TX packet.
15:12	RESERVED
11:0	**TXCSSP - TX Checksum Start Pointer** This field indicates start offset, in bytes, where the checksum calculation will begin in the associated TX packet. **Note:** The data checksum calculation must not begin in the MAC header (first 14 bytes) or in the last 4 bytes of the TX packet.

Note: When the TXCOE is enabled, the third DWORD of the pre-pended packet is not transmitted. However, 4 bytes must be added to the packet length field in TX Command B.

Note: Software applications must stop the transmitter and flush the TX data path before changing the state of the TXCOE_EN bit. However, the CK bit of TX Command B can be set or cleared on a per-packet basis.

Note: The TXCOE_MODE may only be changed if the TX path is disabled. If it is desired to change this value during run time, it is safe to do so only after the TX Ethernet path is disabled and the TLI is empty.

Note: The TX checksum preamble must be DWORD-aligned.

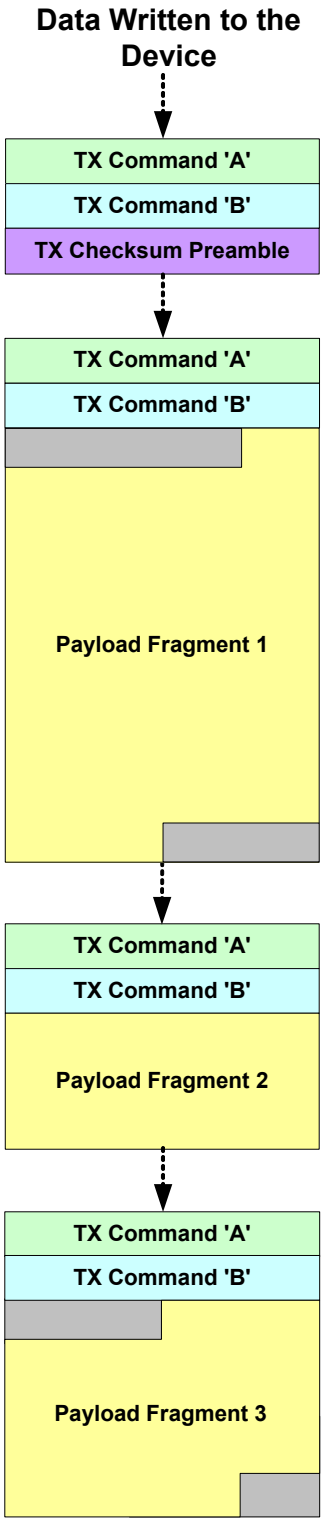
Note: TX preamble size is accounted for in both the buffer length and packet length.

Note: The first buffer, which contains the TX preamble, may not contain any Ethernet frame data.

Figure 4-16 illustrates the use of a pre-pended checksum preamble when transmitting an Ethernet frame consisting of 3 payload buffers.

FIGURE 4-16: TX EXAMPLE ILLUSTRATING A PRE-PENDED TX CHECKSUM PREAMBLE

Note: The TX Checksum Preamble is pre-pended to data to be transmitted. FS is set in TX Command 'A' and CK is set in TX Command 'B'. No start offset may be added. FS must not be set for subsequent fragments of the same packet.



4.5.8.1 TX Checksum Calculation

The TX checksum calculation is performed using the same operation as the RX checksum shown in Section 4.5.7.1, with the exception that the calculation starts as indicated by the preamble, and the transmitted checksum is the one's-compliment of the final calculation.

Note:	When the TX checksum offload feature is invoked, if the calculated checksum is 0000h, it is left unaltered. UDP checksums are optional under IPv4, and a zero checksum calculated by the TX checksum offload feature will erroneously indicate to the receiver that no checksum was calculated, however, the packet will typically not be rejected by the receiver. Under IPv6, however, according to RFC 2460, the UDP checksum is not optional. A calculated checksum that yields a result of zero must be changed to FFFFh for insertion into the UDP header. IPv6 receivers discard UDP packets containing a zero checksum. **Thus, this feature must not be used for UDP checksum calculation under IPv6.**

4.5.9 MAC CONTROL AND STATUS REGISTERS (MCSR)

Refer to Section 6.4, "MAC Control and Status Registers" for a complete description of the MCSR.

4.6 10/100 Internal Ethernet PHY

The device integrates an IEEE 802.3 Physical Layer for Twisted Pair Ethernet applications. The PHY can be configured for either 100 Mbps (100BASE-TX) or 10 Mbps (10BASE-T) Ethernet operation in either full- or half-duplex configurations. The PHY block includes auto-negotiation. Minimal external components are required for the utilization of the internal PHY.

The device provides an option to use an external PHY in place of the internal PHY. The external PHY can be connected via the Media Independent Interface (MII) port. This option is useful for supporting Home PNA operations. When an external PHY is used, the internal PHY must be placed into general power down via a PHY reset (refer to Section 4.6.9, "PHY Resets" for further information).

Functionally, the internal PHY can be divided into the following sections:

- 100BASE-TX transmit and receive
- 10BASE-T transmit and receive
- Internal MII to the Ethernet Media Access Controller
- Auto-negotiation to automatically determine the best speed and duplex possible
- Management Control to read status registers and write control registers

The device's Ethernet interface requires a software sequence to be compliant with the IEEE 802.3 Output V_{OH+}/V_{OH-} voltage specification. Microchip has not experienced any functional limitations if this sequence is not implemented, although IEEE 802.3 compliance may fail by approximately 25 mV if not enabled. The software sequence has been implemented successfully in the driver released by Microchip. Refer to the Microchip driver source code for an example of this implementation.

The following pseudo-code structures detail the required sequence:

```
// Enable access to VOH Compliance registers
REG_ACCESS_ENABLE:
MII_Write: Address 0x14, Data 0x0400
MII_Write: Address 0x14, Data 0x0000
MII_Write: Address 0x14, Data 0x0400

// Enable the VOH Compliance mode
PHY_VOHCOMP_ENABLE:
MII_Write: Address 0x17, Data 0x85E8
(Read MII Address 0x14 until Bit 14 is cleared)
MII_Write: Address 0x14, Data 0x4416
```

// Check that the VOH Compliance mode is active

PHY_VOHCOMP_CHECK:

MII_Write: Address 0x14, Data 0x86C0

(Read MII Address 0x14 until Bit 15 is cleared)

MII_Read: Address 0x15

(if ReadData = 16'bXXXXXX11XXXXXXXX, then VOH Compliance is enabled)

The entire sequence that should occur is:

REG_ACCESS_ENABLE -> PHY_VOHCOMP_ENABLE -> PHY_VOHCOMP_CHECK

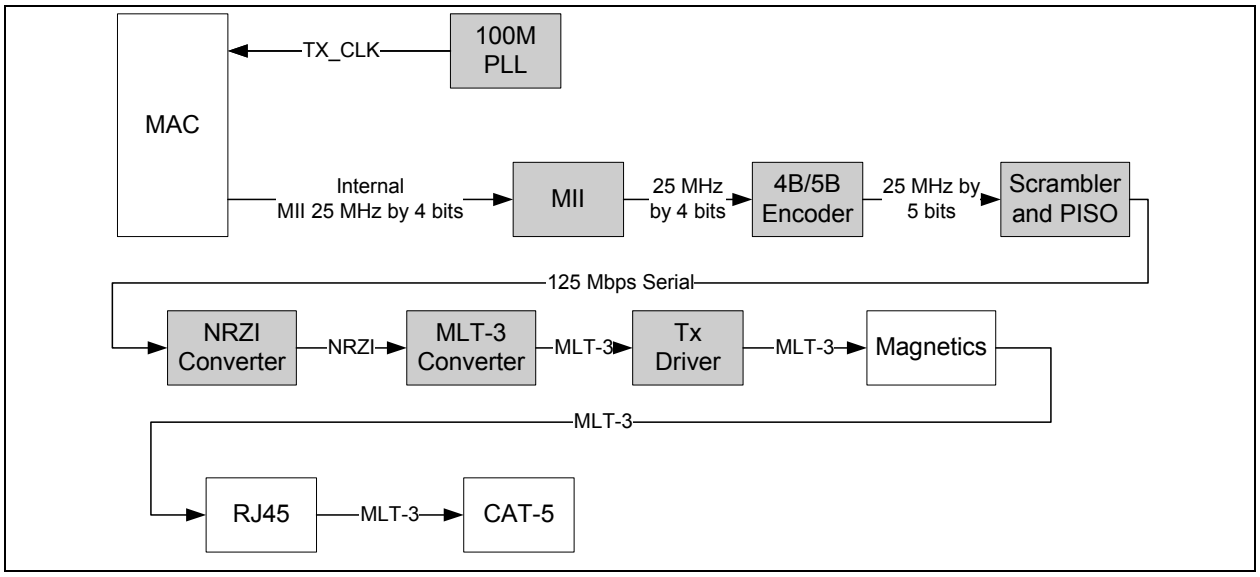
It is recommended to run this sequence upon the following list of events detected by the software driver:

1. Device Hardware Reset:
   a) Assertion of the External Chip Reset (nRESET)
   b) Power on Reset (POR)
2. Device Software Reset:
   a) Setting of the Soft Reset (SRST) bit of the Hardware Configuration Register (HW_CFG)
   b) Setting of the PHY Soft Reset bit of the Basic Control Register
3. Exit from Energy Detect Power-Down (EDPD) mode
4. Auto-Negotiation Enable/Disable via the Auto-Negotiation Enable bit of the Basic Control Register
5. Exit from General Power-Down mode

4.6.1 100BASE-TX TRANSMIT

The data path of the 100Base-TX is shown in Figure 4-17. Each major block is explained in the following sections.

FIGURE 4-17: 100BASE-TX DATA PATH



4.6.1.1 4B/5B Encoding

The transmit data passes from the MII block to the 4B/5B encoder. This block encodes the data from 4-bit nibbles to 5-bit symbols (known as code-groups) according to Table 4-54. Each 4-bit data-nibble is mapped to 16 of the 32 possible code-groups. The remaining 16 code-groups are either used for control information or are not valid.

The first 16 code-groups are referred to by the hexadecimal values of their corresponding data nibbles, 0 through F. The remaining code-groups are given letter designations with slashes on either side. For example, an IDLE code-group is /I/, a transmit error code-group is /H/, etc.

The encoding process may be bypassed by clearing bit 6 of register 31. When the encoding is bypassed, the 5th transmit data bit is equivalent to TX_ER.

TABLE 4-54: 4B/5B CODE TABLE

Code Group	Sym	Receiver Interpretation			Transmitter Interpretation		
11110	0	0	0000	DATA	0	0000	DATA
01001	1	1	0001		1	0001	
10100	2	2	0010		2	0010	
10101	3	3	0011		3	0011	
01010	4	4	0100		4	0100	
01011	5	5	0101		5	0101	
01110	6	6	0110		6	0110	
01111	7	7	0111		7	0111	
10010	8	8	1000		8	1000	
10011	9	9	1001		9	1001	
10110	A	A	1010		A	1010	
10111	B	B	1011		B	1011	
11010	C	C	1100		C	1100	
11011	D	D	1101		D	1101	
11100	E	E	1110		E	1110	
11101	F	F	1111		F	1111	
11111	I	IDLE			Sent after /T/R until TX_EN		
11000	J	First nibble of SSD, translated to “0101” following IDLE, else RX_ER			Sent for rising TX_EN		
10001	K	Second nibble of SSD, translated to “0101” following J, else RX_ER			Sent for rising TX_EN		
01101	T	First nibble of ESD, causes de-assertion of CRS if followed by /R/, else assertion of RX_ER			Sent for falling TX_EN		
00111	R	Second nibble of ESD, causes deassertion of CRS if following /T/, else assertion of RX_ER			Sent for falling TX_EN		
00100	H	Transmit Error Symbol			Sent for rising TX_ER		
00110	V	INVALID, RX_ER if during RX_DV			INVALID		
11001	V	INVALID, RX_ER if during RX_DV			INVALID		
00000	V	INVALID, RX_ER if during RX_DV			INVALID		
00001	V	INVALID, RX_ER if during RX_DV			INVALID		
00010	V	INVALID, RX_ER if during RX_DV			INVALID		
00011	V	INVALID, RX_ER if during RX_DV			INVALID		
00101	V	INVALID, RX_ER if during RX_DV			INVALID		
01000	V	INVALID, RX_ER if during RX_DV			INVALID		
01100	V	INVALID, RX_ER if during RX_DV			INVALID		

TABLE 4-54: 4B/5B CODE TABLE (CONTINUED)

Code Group	Sym	Receiver Interpretation	Transmitter Interpretation
10000	V	INVALID, RX_ER if during RX_DV	INVALID

4.6.1.2 Scrambling

Repeated data patterns (especially the IDLE code-group) can have power spectral densities with large narrow-band peaks. Scrambling the data helps eliminate these peaks and spread the signal power more uniformly over the entire channel bandwidth. This uniform spectral density is required by FCC regulations to prevent excessive EMI from being radiated by the physical wiring.

The scrambler also performs the Parallel In Serial Out conversion (PISO) of the data.

4.6.1.3 NRZI and MLT-3 Encoding

The scrambler block passes the 5-bit wide parallel data to the NRZI converter where it becomes a serial 125 MHz NRZI data stream. The NRZI is encoded to MLT-3. MLT-3 is a tri-level code where a change in the logic level represents a code bit “1” and the logic output remaining at the same level represents a code bit “0”.

4.6.1.4 100M Transmit Driver

The MLT-3 data is then passed to the analog transmitter, which launches the differential MLT-3 signal, on outputs TXP and TXN, to the twisted pair media via a 1:1 ratio isolation transformer. The 10Base-T and 100Base-TX signals pass through the same transformer so that common magnetics can be used for both. The transmitter drives into the 100 Ω impedance of the CAT-5 cable. Cable termination and impedance matching require external components.

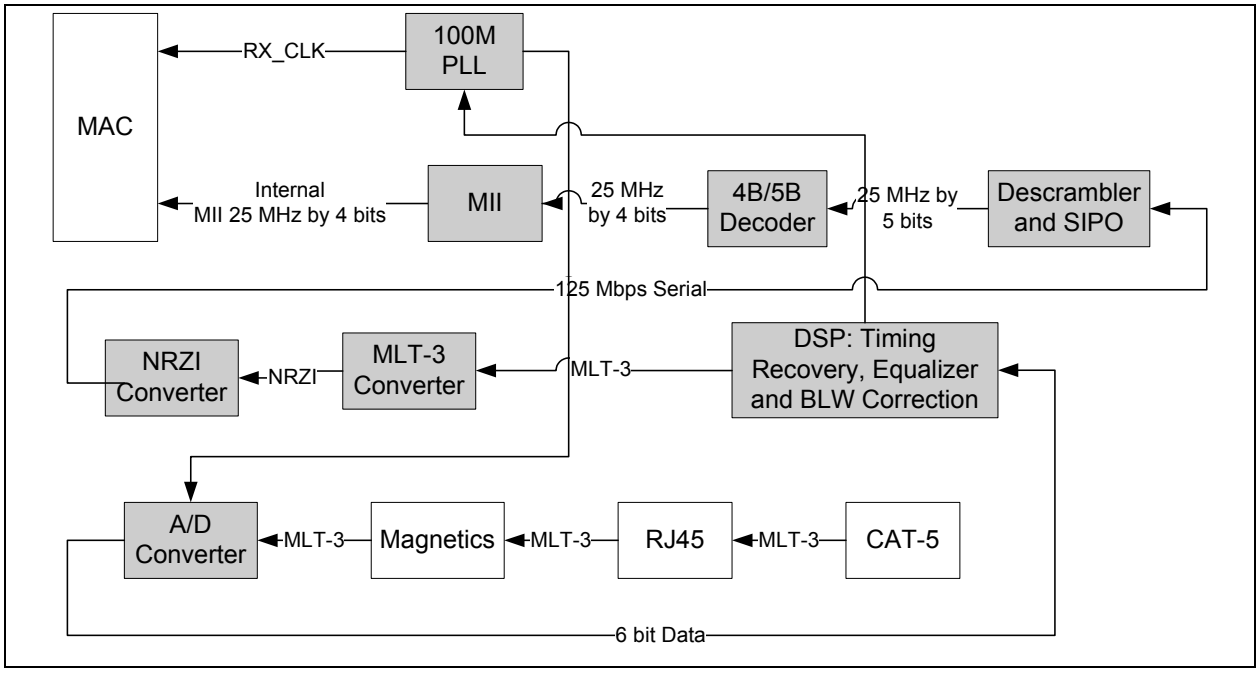
4.6.1.5 100M Phase Lock Loop (PLL)

The 100M PLL locks onto reference clock and generates the 125 MHz clock used to drive the 125 MHz logic and the 100Base-Tx Transmitter.

4.6.2 100BASE-TX RECEIVE

The receive data path is shown in Figure 4-18. Detailed descriptions are given in the following subsections.

FIGURE 4-18: RECEIVE DATA PATH



4.6.2.1 100M Receive Input

The MLT-3 from the cable is fed into the PHY (on inputs RXP and RXN) via a 1:1 ratio transformer. The ADC samples the incoming differential signal at a rate of 125M samples per second. Using a 64-level quantizer, it generates 6 digital bits to represent each sample. The DSP adjusts the gain of the ADC according to the observed signal levels such that the full dynamic range of the ADC can be used.

4.6.2.2 Equalizer, Baseline Wander Correction and Clock and Data Recovery

The 6 bits from the ADC are fed into the DSP block. The equalizer in the DSP section compensates for phase and amplitude distortion caused by the physical channel consisting of magnetics, connectors, and CAT- 5 cable. The equalizer can restore the signal for any good-quality CAT-5 cable between 1 m and 150 m.

If the DC content of the signal is such that the low-frequency components fall below the low frequency pole of the isolation transformer, then the droop characteristics of the transformer will become significant and Baseline Wander (BLW) on the received signal will result. To prevent corruption of the received data, the PHY corrects for BLW and can receive the ANSI X3.263-1995 FDDI TP-PMD defined “killer packet” with no bit errors.

The 100M PLL generates multiple phases of the 125 MHz clock. A multiplexer, controlled by the timing unit of the DSP, selects the optimum phase for sampling the data. This is used as the received recovered clock. This clock is used to extract the serial data from the received signal.

4.6.2.3 NRZI and MLT-3 Decoding

The DSP generates the MLT-3 recovered levels that are fed to the MLT-3 converter. The MLT-3 is then converted to an NRZI data stream.

4.6.2.4 Descrambling

The descrambler performs an inverse function to the scrambler in the transmitter and also performs the Serial In Parallel Out (SIPO) conversion of the data.

During reception of IDLE (/I/) symbols, the descrambler synchronizes its descrambler key to the incoming stream. Once synchronization is achieved, the descrambler locks on this key and is able to descramble incoming data.

Special logic in the descrambler ensures synchronization with the remote PHY by searching for IDLE symbols within a window of 4000 bytes (40 µs). This window ensures that a maximum packet size of 1514 bytes, allowed by the IEEE 802.3 standard, can be received with no interference. If no IDLE-symbols are detected within this time-period, receive operation is aborted and the descrambler re-starts the synchronization process.

The descrambler can be bypassed by setting bit 0 of register 31.

4.6.2.5 Alignment

The de-scrambled signal is then aligned into 5-bit code-groups by recognizing the /J/K/ Start-of-Stream Delimiter (SSD) pair at the start of a packet. Once the code-word alignment is determined, it is stored and utilized until the next start of frame.

4.6.2.6 5B/4B Decoding

The 5-bit code-groups are translated into 4-bit data nibbles according to the 4B/5B table. The SSD, /J/K/, is translated to “0101 0101” as the first 2 nibbles of the MAC preamble. Reception of the SSD causes the PHY to assert the internal RX_DV signal, indicating that valid data is available on the Internal RXD bus. Successive valid code-groups are translated to data nibbles. Reception of either the End of Stream Delimiter (ESD) consisting of the /T/R/ symbols, or at least two /I/ symbols causes the PHY to de-assert the internal carrier sense and RX_DV.

These symbols are not translated into data.

4.6.2.7 Receiver Errors

During a frame, unexpected code-groups are considered receive errors. Expected code groups are the DATA set (0 through F), and the /T/R/ (ESD) symbol pair. When a receive error occurs, the internal MII's RX_ER signal is asserted and arbitrary data is driven onto the internal receive data bus (RXD) to the MAC. Should an error be detected during the time that the /J/K/ delimiter is being decoded (bad SSD error), RX_ER is asserted and the value 1110b is driven onto the internal receive data bus (RXD) to the MAC. Note that the internal MII's data valid signal (RX_DV) is not yet asserted when the bad SSD occurs.

4.6.3 10BASE-T TRANSMIT

Data to be transmitted comes from the MAC layer controller. The 10Base-T transmitter receives 4-bit nibbles from the MII at a rate of 2.5 MHz and converts them to a 10 Mbps serial data stream. The data stream is then Manchester encoded and sent to the analog transmitter, which drives a signal onto the twisted pair via the external magnetics.

The 10M transmitter uses the following blocks:

- MII (digital)
- TX 10M (digital)
- 10M Transmitter (analog)
- 10M PLL (analog)

4.6.3.1 10M Transmit Data Across the Internal MII Bus

The MAC controller drives the transmit data onto the internal TXD BUS. When the controller has driven TX_EN high to indicate valid data, the data is latched by the MII block on the rising edge of TX_CLK. The data is in the form of 4-bit wide 2.5 MHz data.

4.6.3.2 Manchester Encoding

The 4-bit wide data is sent to the TX10M block. The nibbles are converted to a 10 Mbps serial NRZI data stream. The 10M PLL locks onto the external clock or internal oscillator and produces a 20 MHz clock. This is used to Manchester encode the NRZ data stream. When no data is being transmitted (TX_EN is low), the TX10M block outputs Normal Link Pulses (NLPs) to maintain communications with the remote link partner.

4.6.3.3 10M Transmit Drivers

The Manchester encoded data is sent to the analog transmitter where it is shaped and filtered before being driven out as a differential signal across the TXP and TXN outputs.

4.6.4 10BASE-T RECEIVE

The 10BASE-T receiver gets the Manchester encoded analog signal from the cable via the magnetics. It recovers the receive clock from the signal and uses this clock to recover the NRZI data stream. This 10M serial data is converted to 4-bit data nibbles which are passed to the controller across the MII at a rate of 2.5 MHz.

This 10M receiver uses the following blocks:

- Filter and SQUELCH (analog)
- 10M PLL (analog)
- RX 10M (digital)
- MII (digital)

4.6.4.1 10M Receive Input and Squelch

The Manchester signal from the cable is fed into the PHY (on inputs RXP and RXN) via 1:1 ratio magnetics. It is first filtered to reduce any out-of-band noise. It then passes through a SQUELCH circuit. The SQUELCH is a set of amplitude and timing comparators that normally reject differential voltage levels below 300 mV and detect and recognize differential voltages above 585 mV.

4.6.4.2 Manchester Decoding

The output of the SQUELCH goes to the RX10M block where it is validated as Manchester encoded data. The polarity of the signal is also checked. If the polarity is reversed (local RXP is connected to RXN of the remote partner and vice versa), then this is identified and corrected. The reversed condition is indicated by the flag “XPOL“, bit 4 in register 27. The 10M PLL is locked onto the received Manchester signal and from this, generates the received 20 MHz clock. Using this clock, the Manchester encoded data is extracted and converted to a 10 MHz NRZI data stream. It is then converted from serial to 4-bit wide parallel data.

The RX10M block also detects valid 10Base-T IDLE signals - Normal Link Pulses (NLPs) - to maintain the link.

4.6.4.3 Jabber Detection

Jabber is a condition in which a station transmits for a period of time longer than the maximum permissible packet length, usually due to a fault condition, that results in holding the TX_EN input for a long period. Special logic is used to detect the jabber state and abort the transmission to the line, within 45 ms. Once TX_EN is deasserted, the logic resets the jabber condition.

4.6.5 AUTO-NEGOTIATION

The purpose of the auto-negotiation function is to automatically configure the PHY to the optimum link parameters based on the capabilities of its link partner. Auto-negotiation is a mechanism for exchanging configuration information between two link-partners and automatically selecting the highest performance mode of operation supported by both sides. Auto-negotiation is fully defined in clause 28 of the IEEE 802.3 specification.

Once auto-negotiation has completed, information about the resolved link can be passed back to the controller via the internal Serial Management Interface (SMI). The results of the negotiation process are reflected in the Speed Indication bits in register 31, as well as the Link Partner Ability register (Register 5).

The auto-negotiation protocol is a purely physical layer activity and proceeds independently of the MAC controller.

The advertised capabilities of the PHY are stored in register 4 of the SMI registers. The default advertised by the PHY is determined by user-defined on-chip signal options.

The following blocks are activated during an auto-negotiation session:

- Auto-negotiation (digital)
- 100M ADC (analog)
- 100M PLL (analog)
- 100M equalizer/BLW/clock recovery (DSP)
- 10M SQUELCH (analog)
- 10M PLL (analog)
- 10M Transmitter (analog)

When enabled, auto-negotiation is started by the occurrence of one of the following events:

- Hardware reset
- Software reset
- Power-down reset
- Link status down
- Setting register 0, bit 9 high (auto-negotiation restart)

On detection of one of these events, the PHY begins auto-negotiation by transmitting bursts of Fast Link Pulses (FLP). These are bursts of link pulses from the 10M transmitter. They are shaped as Normal Link Pulses and can pass uncorrupted down CAT-3 or CAT-5 cable. A Fast Link Pulse Burst consists of up to 33 pulses. The 17 odd-numbered pulses, which are always present, frame the FLP burst. The 16 even-numbered pulses, which may be present or absent, contain the data word being transmitted. Presence of a data pulse represents a “1”, while absence represents a “0”.

The data transmitted by an FLP burst is known as a Link Code Word. These are defined fully in IEEE 802.3 clause 28. In summary, the PHY advertises 802.3 compliance in its selector field (the first 5 bits of the Link Code Word). It advertises its technology ability according to the bits set in register 4 of the SMI registers.

There are 4 possible matches of the technology abilities. In the order of priority these are:

- 100M full-duplex (highest priority)
- 100M half-duplex
- 10M full-duplex
- 10M half-duplex

If the full capabilities of the PHY are advertised (100M, full-duplex), and if the link partner is capable of 10M and 100M, then auto-negotiation selects 100M as the highest performance mode. If the link partner is capable of half- and full-duplex modes, then auto-negotiation selects full-duplex as the highest performance operation.

Once a capability match has been determined, the link code words are repeated with the acknowledge bit set. Any difference in the main content of the link code words at this time will cause auto-negotiation to re-start. Auto-negotiation will also re-start if not all of the required FLP bursts are received.

Writing register 4 bits [8:5] allows software control of the capabilities advertised by the PHY. Writing register 4 does not automatically re-start auto-negotiation. Register 0, bit 9 must be set before the new abilities will be advertised. Auto-negotiation can also be disabled via software by clearing register 0, bit 12.

The device does not support Next Page capability.

4.6.6 PARALLEL DETECTION

If LAN9730/LAN9730i is connected to a device lacking the ability to auto-negotiate (i.e., no FLPs are detected), it is able to determine the speed of the link based on either 100M MLT-3 symbols or 10M Normal Link Pulses. In this case the link is presumed to be half-duplex per the IEEE standard. This ability is known as “Parallel Detection. This feature ensures interoperability with legacy link partners. If a link is formed via parallel detection, then bit 0 in register 6 is cleared to indicate that the Link Partner is not capable of auto-negotiation. The Ethernet MAC has access to this information via the management interface. If a fault occurs during parallel detection, bit 4 of register 6 is set.

Register 5 is used to store the Link Partner Ability information, which is coded in the received FLPs. If the Link Partner is not auto-negotiation capable, then register 5 is updated after completion of parallel detection to reflect the speed capability of the Link Partner.

4.6.6.1 Re-Starting Auto-Negotiation

Auto-negotiation can be re-started at any time by setting register 0, bit 9. Auto-negotiation will also re-start if the link is broken at any time. A broken link is caused by signal loss. This may occur because of a cable break, or because of an interruption in the signal transmitted by the link partner. Auto-negotiation resumes in an attempt to determine the new link configuration.

If the management entity re-starts auto-negotiation by writing to bit 9 of the control register, the device will respond by stopping all transmission/receiving operations. Once the break_link_timer is done, in the auto-negotiation state-machine (approximately 1250 ms) the auto-negotiation will re-start. The link partner will have also dropped the link due to lack of a received signal, so it too will resume auto-negotiation.

4.6.6.2 Disabling Auto-Negotiation

Auto-negotiation can be disabled by setting register 0, bit 12 to zero. The device will then force its speed of operation to reflect the information in register 0, bit 13 (speed) and register 0, bit 8 (duplex). The speed and duplex bits in register 0 should be ignored when auto-negotiation is enabled.

4.6.6.3 Half vs. Full-Duplex

Half-duplex operation relies on the CSMA/CD (Carrier Sense Multiple Access/Collision Detect) protocol to handle network traffic and collisions. In this mode, the internal carrier sense signal, CRS, responds to both transmit and receive activity. In this mode, if data is received while the PHY is transmitting, a collision results.

In full-duplex mode, the PHY is able to transmit and receive data simultaneously. In this mode, the internal CRS responds only to receive activity. The CSMA/CD protocol does not apply and collision detection is disabled.

Table 4-55 describes the behavior of the internal CRS bit under all receive/transmit conditions.

The internal CRS signal is used to trigger bit 10 (No Carrier) of the TX Status Word (see Section 4.4.2.6, "TX Status Format"). The CRS value, and subsequently the No Carrier value, are invalid during any full-duplex transmission. Therefore, these signals cannot be used as a verification method of transmitted packets when transmitting in 10/100 Mbps full-duplex modes.

TABLE 4-55: CRS BEHAVIOR

Mode	Speed	Duplex	Activity	CRS Behavior Note 4-25
Manual	10 Mbps	Half-Duplex	Transmitting	Active
Manual	10 Mbps	Half-Duplex	Receiving	Active
Manual	10 Mbps	Full-Duplex	Transmitting	Low
Manual	10 Mbps	Full-Duplex	Receiving	Active
Manual	100 Mbps	Half-Duplex	Transmitting	Active
Manual	100 Mbps	Half-Duplex	Receiving	Active
Manual	100 Mbps	Full-Duplex	Transmitting	Low
Manual	100 Mbps	Full-Duplex	Receiving	Active
Auto-Negotiation	10 Mbps	Half-Duplex	Transmitting	Active
Auto-Negotiation	10 Mbps	Half-Duplex	Receiving	Active
Auto-Negotiation	10 Mbps	Full-Duplex	Transmitting	Low
Auto-Negotiation	10 Mbps	Full-Duplex	Receiving	Active
Auto-Negotiation	100 Mbps	Half-Duplex	Transmitting	Active
Auto-Negotiation	100 Mbps	Half-Duplex	Receiving	Active
Auto-Negotiation	100 Mbps	Full-Duplex	Transmitting	Low
Auto-Negotiation	100 Mbps	Full-Duplex	Receiving	Active

Note 4-25 The device's 10/100 PHY internal CRS signal operates in two modes: Active and Low. When in Active mode, the internal CRS will transition high and low upon line activity, where a high value indicates a carrier has been detected. In Low mode, the internal CRS stays low and does not indicate carrier detection. The internal CRS signal and No Carrier (bit 10 of the TX Status Word) cannot be used as a verification method of transmitted packets when transmitting in 10/100 Mbps full-duplex mode.

4.6.7 HP AUTO-MDIX

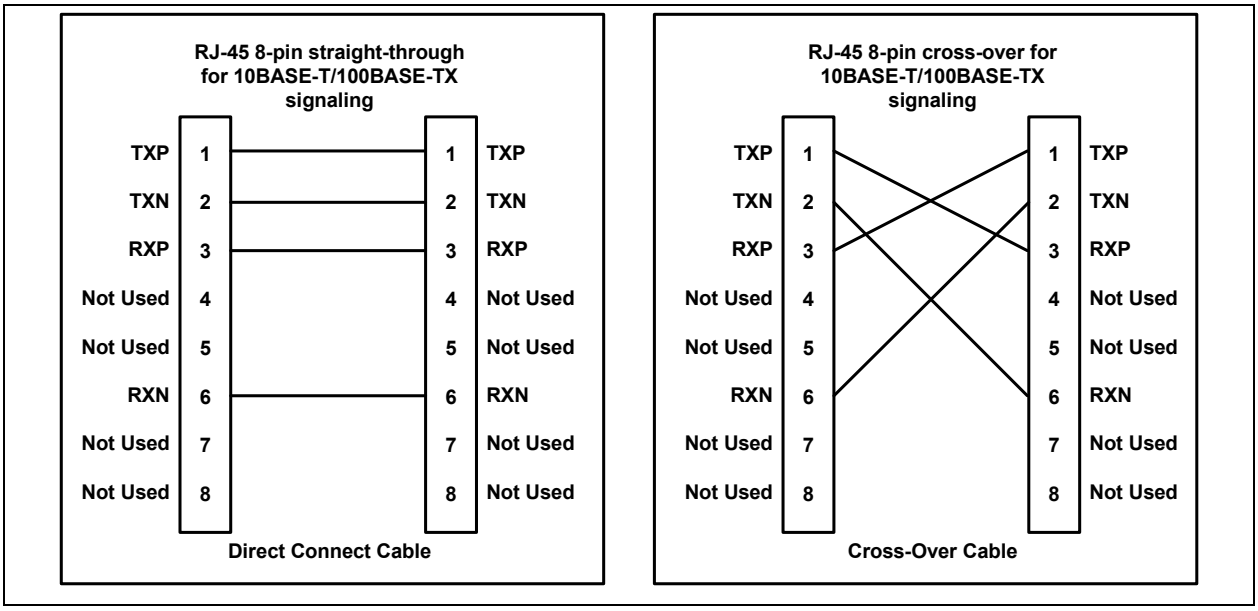
HP Auto-MDIX facilitates the use of CAT-3 (10BASE-T) or CAT-5 (100BASE-TX) media UTP interconnect cable without consideration of interface wiring scheme. If a user plugs in either a direct connect LAN cable, or a cross-over patch cable, as shown in Figure 4-19, the device's Auto-MDIX PHY is capable of configuring the TPO and TPI twisted pair pins for correct transceiver operation.

The internal logic of the device detects the TX and RX pins of the connecting device. Since the RX and TX line pairs are interchangeable, special PCB design considerations are needed to accommodate the symmetrical magnetics and termination of an Auto-MDIX design.

The Auto-MDIX function can be disabled through the Special Control/Status Indications Register or the external AUTOMDIX_EN configuration strap.

Note:	When operating in 10BASE-T or 100BASE-TX manual modes, the Auto-MDIX crossover time can be extended via the Extend Manual 10/100 Auto-MDIX Crossover Time bit of the EDPD NLP/Crossover TimeRegister. Refer to Section 6.5.8, "EDPD NLP/Crossover TimeRegister" for additional information.

FIGURE 4-19: DIRECT CABLE CONNECTION VS. CROSS-OVER CABLE CONNECTION



4.6.8 PHY POWER-DOWN MODES

There are two power-down modes for the PHY as discussed in the following sections.

4.6.8.1 General Power-Down

This power-down is controlled by register 0, bit 11. In this mode the PHY, except the management interface, is powered-down and stays in that condition as long as PHY register bit 0.11 is high. When bit 0.11 is cleared, the PHY powers up and is automatically reset. Refer to Section 6.5.1, "Basic Control Register" for additional information on this register.

Note:	For maximum power savings, auto-negotiation should be disabled before enabling the General Power-Down mode.

4.6.8.2 Energy Detect Power-Down (EDPD)

This power-down mode is activated by setting the EDPWRDOWN bit of the Mode Control/Status Register. In this mode, when no energy is present on the line, the PHY is powered down (except the one for the management interface, the SQUELCH circuit, and the ENERGYON logic). The ENERGYON logic is used to detect the presence of valid energy from 100BASE-TX, 10BASE-T, or auto-negotiation signals.

In this mode, when the ENERGYON bit of the Mode Control/Status Register is low, the PHY is powered-down and nothing is transmitted. When energy is received via link pulses or packets, the ENERGYON bit goes high and the PHY powers-up. The PHY automatically resets itself into the state prior to power-down and asserts the INT7 bit of the PHY Interrupt Source Flag Register register. If the ENERGYON interrupt is enabled, this event will cause a PHY interrupt to the Interrupt Controller and the power management event detection logic. The first and possibly the second packet to activate ENERGYON may be lost.

When the EDPWRDOWN bit of the Mode Control/Status Register is low, Energy Detect Power-Down is disabled.

When in EDPD mode, the device's NLP characteristics may be modified. The device can be configured to transmit NLPs in EDPD via the EDPD TX NLP Enable bit of the EDPD NLP/Crossover TimeRegister. When enabled, the TX NLP time interval is configurable via the EDPD TX NLP Interval Timer Select field of the EDPD NLP/Crossover TimeRegister. When in EDPD mode, the device can also be configured to wake on the reception of one or two NLPs. Setting the EDPD RX Single NLP Wake Enable bit of the EDPD NLP/Crossover TimeRegister will enable the device to wake on reception of a single NLP. If the EDPD RX Single NLP Wake Enable bit is cleared, the maximum interval for detecting reception of two NLPs to wake from EDPD is configurable via the EDPD RX NLP Max Interval Detect Select field of the EDPD NLP/Crossover TimeRegister.

4.6.9 PHY RESETS

In addition to a chip-level reset, the PHY supports two software-initiated resets. These are discussed in the following sections.

4.6.9.1 PHY Soft Reset via PMT_CTL Register PHY Reset (PHY_RST) Bit

The PHY soft reset is initiated by writing a '1' to the PHY Reset (PHY_RST) bit of the Power Management Control Register (PMT_CTL). This self-clearing bit will return to '0' after approximately 2 ms, at which time the PHY reset is complete.

4.6.9.2 PHY Soft Reset via PHY Basic Control Register Bit 15 (PHY Reg. 0.15)

The PHY Reg. 0.15 Soft Reset is initiated by writing a '1' to bit 15 of the PHY's Basic Control Register. This self-clearing bit will return to '0' after approximately 256 µs, at which time the PHY reset is complete. The BCR reset initializes the logic within the PHY, with the exception of register bits marked as NASR (Not Affected by Software Reset).

4.6.10 REQUIRED ETHERNET MAGNETICS

The magnetics selected for use with the device should be an Auto-MDIX style magnetic available from several vendors. The user is urged to review Microchip Application Note 8.13, Suggested Magnetics for the latest qualified and suggested magnetics. Vendors and part numbers are provided in this application note.

4.6.11 PHY REGISTERS

Refer to Section 6.5, "PHY Registers" for a complete description of the PHY registers.

4.7 EEPROM Controller (EPC)

The device may use an external EEPROM to store the default values for the USB descriptors and the MAC address. The EEPROM controller supports most 256/512 byte “93C46” type EEPROMs.

Note:	A 3-wire style 2k/4k EEPROM that is organized for 256/512 x 8-bit operation must be used.

The MAC address is used as the default Ethernet MAC address and is loaded into the MAC’s ADDRH and ADDRL registers. If a properly configured EEPROM is not detected, it is the responsibility of the host LAN Driver to set the IEEE addresses.

After a system-level reset occurs, the device will load the default values from a properly configured EEPROM. The device will not accept USB transactions from the host until this process is completed.

The EEPROM controller also allows the host system to read, write and erase the contents of the serial EEPROM.

4.7.1 EEPROM FORMAT

Table 4-56 illustrates the format in which data is stored inside of the EEPROM.

Note the EEPROM offsets are given in units of 16-bit word offsets. A length field with a value of zero indicates that the field does not exist in the EEPROM. The device will use the field’s HW default value in this case.

Note:	For the Device Descriptor, the only valid values for the length are 0 and 18.

Note:	For the configuration and Interface Descriptor, the only valid values for the length are 0 and 18.

Note:	The EEPROM programmer must ensure that if a String Descriptor does not exist in the EEPROM, the referencing descriptor must contain 00h for the respective string index field.

Note:	If all String Descriptor lengths are zero, then a Language ID will not be supported.

TABLE 4-56: EEPROM FORMAT

EEPROM Address	EEPROM Contents
00h	0xA5
01h	MAC Address [7:0]
02h	MAC Address [15:8]
03h	MAC Address [23:16]
04h	MAC Address [31:24]
05h	MAC Address [39:32]
06h	MAC Address [47:40]
07h	Full-Speed Polling Interval for Interrupt Endpoint
08h	Hi-Speed Polling Interval for Interrupt Endpoint
09h	Configuration Flags
0Ah	Language ID Descriptor [7:0]
0Bh	Language ID Descriptor [15:8]
0Ch	Manufacturer ID String Descriptor Length (bytes)
0Dh	Manufacturer ID String Descriptor EEPROM Word Offset
0Eh	Product Name String Descriptor Length (bytes)
0Fh	Product Name String Descriptor EEPROM Word Offset
10h	Serial Number String Descriptor Length (bytes)
11h	Serial Number String Descriptor EEPROM Word Offset
12h	Configuration String Descriptor Length (bytes)
13h	Configuration String Descriptor Word Offset
14h	Interface String Descriptor Length (bytes)
15h	Interface String Descriptor Word Offset

TABLE 4-56: EEPROM FORMAT (CONTINUED)

EEPROM Address	EEPROM Contents
16h	Hi-Speed Device Descriptor Length (bytes)
17h	Hi-Speed Device Descriptor Word Offset
18h	Hi-Speed Configuration and Interface Descriptor Length (bytes)
19h	Hi-Speed Configuration and Interface Descriptor Word Offset
1Ah	Full-Speed Device Descriptor Length (bytes)
1Bh	Full-Speed Device Descriptor Word Offset
1Ch	Full-Speed Configuration and Interface Descriptor Length (bytes)
1Dh	Full-Speed Configuration and Interface Descriptor Word Offset
1Eh	LSB of GPIO Wake 0-10 (GPIOWKn) field of General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE)
1Fh	MSB of GPIO Wake 0-10 (GPIOWKn) field of General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE)
20h	GPIO PME Flags

Note: The descriptor type for the Device Descriptors specified in the EEPROM is a don't care and always overwritten by HW to 0x1.
The descriptor size for the Device Descriptors specified in the EEPROM is a don't care and always overwritten by HW to 0x12.
The descriptor type for the Configuration Descriptors specified in the EEPROM is a don't care and always overwritten by HW to 0x2.

Note: Descriptors specified in EEPROM having bcdUSB, bMaxPacketSize0, and bNumConfigurations fields defined with values other than 0200h, 40h, and 1, respectively, will result in unwanted behavior and untoward results.

Note: EEPROM byte addresses past 20h can be used to store data for any purpose.

Table 4-57 describes the configuration flags. The configuration flags override the affects of the RMT_WKP strap. If a Configuration Descriptor exists in the EEPROM it will override both the configuration flags and associated straps.

TABLE 4-57: CONFIGURATION FLAGS

Bits	Description
7:4	RESERVED
3	RESERVED
2	**Remote Wakeup Support** 0 = The device does not support remote wakeup. 1 = The device supports remote wakeup.
1	**LED Select** Refer to the LED Select (LED_SEL) bit of the LED General Purpose IO Configuration Register (LED_GPIO_CFG) for bit function definitions.
0	**Power Method** 0 = The device is bus powered. 1 = The device is self powered.

Table 4-58 describes the GPIO PME flags.

TABLE 4-58: GPIO PME FLAGS

Bits	Description
7	**GPIO PME Enable** Setting this bit enables the assertion of the GPIO0 or GPIO8 pin, as a result of a Wakeup (GPIO) pin, Magic Packet, or PHY Link Up. The host processor may use the GPIO0/GPIO8 pin to asynchronously wake up, in a manner analogous to a PCI PME pin. GPIO0 signals the event when operating in Internal PHY mode, while GPIO8 signals the event when operating in External PHY mode. Internal or External PHY mode of operation is dictated by the PHY_SEL pin. 0 = The device does not support GPIO PME signaling. 1 = The device supports GPIO PME signaling. **Note:** When this bit is 0, the remaining GPIO PME parameters in this flag byte are ignored.
6	**GPIO PME Configuration** This bit selects whether the GPIO PME is signaled on the GPIO pin as a level or a pulse. If pulse is selected, the duration of the pulse is determined by the setting of the GPIO PME Length bit of this flag byte. The level of the signal or the polarity of the pulse is determined by the GPIO PME Polarity bit of this flag byte. 0 = GPIO PME is signaled via a level. 1 = GPIO PME is signaled via a pulse. **Note:** If GPIO PME Enable is 0, this bit is ignored.
5	**GPIO PME Length** When the GPIO PME Configuration bit of this flag byte indicates that the GPIO PME is signaled by a pulse on the GPIO pin, this bit determines the duration of the pulse. 0 = GPIO PME pulse length is 1.5 ms. 1 = GPIO PME pulse length is 150 ms. **Note:** If GPIO PME Enable is 0, this bit is ignored.
4	**GPIO PME Polarity** Specifies the level of the signal or the polarity of the pulse used for GPIO PME signaling. 0 = GPIO PME signaling polarity is low. 1 = GPIO PME signaling polarity is high. **Note:** If GPIO PME Enable is 0, this bit is ignored.
3	**GPIO PME Buffer Type** This bit selects the output buffer type for GPIO0/GPIO8. 0 = Open drain driver/open source 1 = Push-pull driver **Note:** Buffer type = 0, polarity = 0 implies open drain Buffer type = 0, polarity = 1 implies open source **Note:** If GPIO PME Enable is 0, this bit is ignored.
2	**GPIO PME WOL Select** Three types of wakeup events are supported: Magic Packet, PHY Link Up, and Wakeup Pin(s) assertion. Wakeup Pin(s) are selected via the GPIO Wake 0-10 (GPIOWKn) field of the General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE). The Wakeup Enables are specified in bytes 1Eh and 1Fh of the EEPROM. This bit selects whether Magic Packet or Link Up wakeup events are supported. 0 = Magic Packet wakeup supported. 1 = PHY Link Up wakeup supported (not supported in External PHY mode). **Note:** If GPIO PME Enable is 0, this bit is ignored.

TABLE 4-58: GPIO PME FLAGS (CONTINUED)

Bits	Description
1	**GPIO10 Detection Select** This bit selects the detection mode for GPIO10 when operating in PME mode. In PME mode, GPIO10 is usable in both Internal and External PHY mode as a wakeup pin. This parameter defines whether the wakeup should occur on an active high or active low signal. 0 = Active-low detection for GPIO10 1 = Active-high detection for GPIO10 **Note:** If GPIO PME Enable is 0, this bit is ignored.
0	**RESERVED**

4.7.2 EEPROM DEFAULTS

The signature value of 0xA5 is stored at address 0. A different signature value indicates to the EEPROM controller that no EEPROM or an un-programmed EEPROM is attached to the device. In this case, the hardware default values are used, as shown in Table 4-59. Refer to Section 4.3.1.6, "USB Descriptors" for further information about the default USB values.

TABLE 4-59: EEPROM DEFAULTS

Field	Default Value
MAC Address	FFFFFFFFFFFFh
Full-Speed Polling Interval (ms)	01h
Hi-Speed Polling Interval (ms)	04h
Configuration Flags	04h
Maximum Power (mA)	FAh
Vendor ID	0424h
Product ID	9730h

Note: The Configuration Flags are affected by the RMT_WKP strap.

4.7.3 EEPROM AUTO-LOAD

Certain system level resets (USB reset, POR, nRESET, and SRST) cause the EEPROM contents to be loaded into the device. After a reset, the EEPROM controller attempts to read the first byte of data from the EEPROM. If the value 0xA5 is read from the first address, then the EEPROM controller will assume that an external serial EEPROM is present.

Note: The USB reset only loads the MAC address.

The EEPROM controller will then load the entire contents of the EEPROM into an internal 512 byte SRAM. The contents of the SRAM are accessed by the CTL (USB Control Block) as needed (i.e., to fill Get Descriptor commands). A detailed explanation of the EEPROM byte ordering with respect to the MAC address is given in Section 6.4.3, "MAC Address Low Register (ADDRL)".

If an 0xA5h is not read from the first address, the EEPROM controller will end initialization. The default values, as specified in Table 4-59, will then be assumed by the associated registers. It is then the responsibility of the host LAN driver software to set the IEEE address by writing to the MAC's ADDRH and ADDRL registers.

The device may not respond to the USB host until the EEPROM loading sequence has completed. Therefore, after reset, the USB PHY is kept in the disconnect state until the EEPROM load has completed.

4.7.4 EEPROM HOST OPERATIONS

After the EEPROM controller has finished reading (or attempting to read) the EEPROM after a system-level reset, the host is free to perform other EEPROM operations. EEPROM operations are performed using the EEPROM Command (E2P_CMD) and EEPROM Data (E2P_DATA) registers. Section 6.3.12, "EEPROM Command Register (E2P_CMD)" provides an explanation of the supported EEPROM operations.

If the EEPROM operation is the “write location” (WRITE) or “write all” (WRAL) commands, the host must first write the desired data into the E2P_DATA register. The host must then issue the WRITE or WRAL command using the E2P_CMD register by setting the EPC_CMD field appropriately. If the operation is a WRITE, the EPC_ADDR field in E2P_CMD must also be set to the desired location. The command is executed when the host sets the EPC_BSY bit high. The completion of the operation is indicated when the EPC_BSY bit is cleared.

If the EEPROM operation is the “read location” (READ) operation, the host must issue the READ command using the E2P_CMD register with the EPC_ADDR set to the desired location. The command is executed when the host sets the EPC_BSY bit high. The completion of the operation is indicated when the EPC_BSY bit is cleared, at which time the data from the EEPROM may be read from the E2P_DATA register.

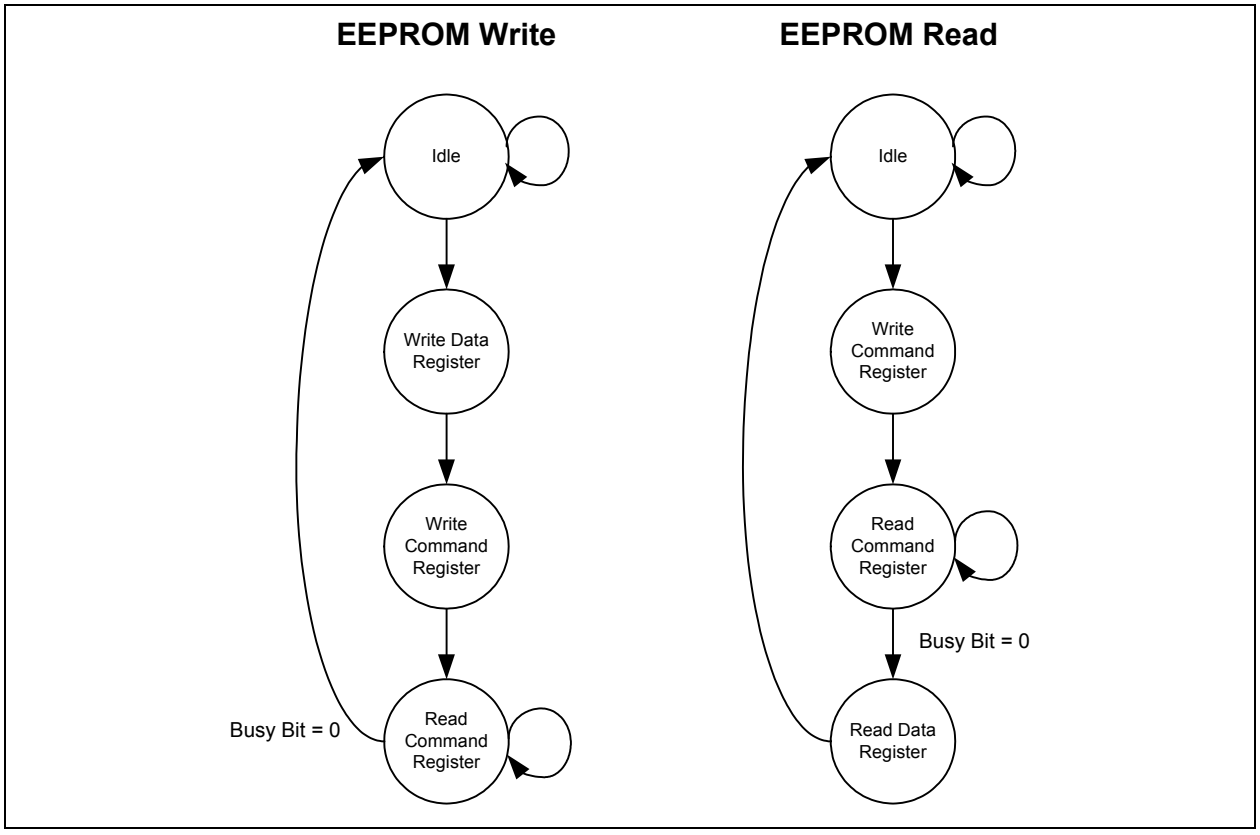
Other EEPROM operations are performed by writing the appropriate command to the E2P_CMD register. The command is executed when the host sets the EPC_BSY bit high. The completion of the operation is indicated when the EPC_BSY bit is cleared. In all cases, the host must wait for EPC_BSY to clear before modifying the E2P_CMD register.

Note:	The EEPROM device powers-up in the erase/write disabled state. To modify the contents of the EEPROM, the host must first issue the EWEN command.

If an operation is attempted, and an EEPROM device does not respond within 30 ms, the device will time-out, and the EPC time-out bit (EPC_TO) in the E2P_CMD register will be set.

Figure 4-20 illustrates the host accesses required to perform an EEPROM Read or Write operation.

FIGURE 4-20: EEPROM ACCESS FLOW DIAGRAM

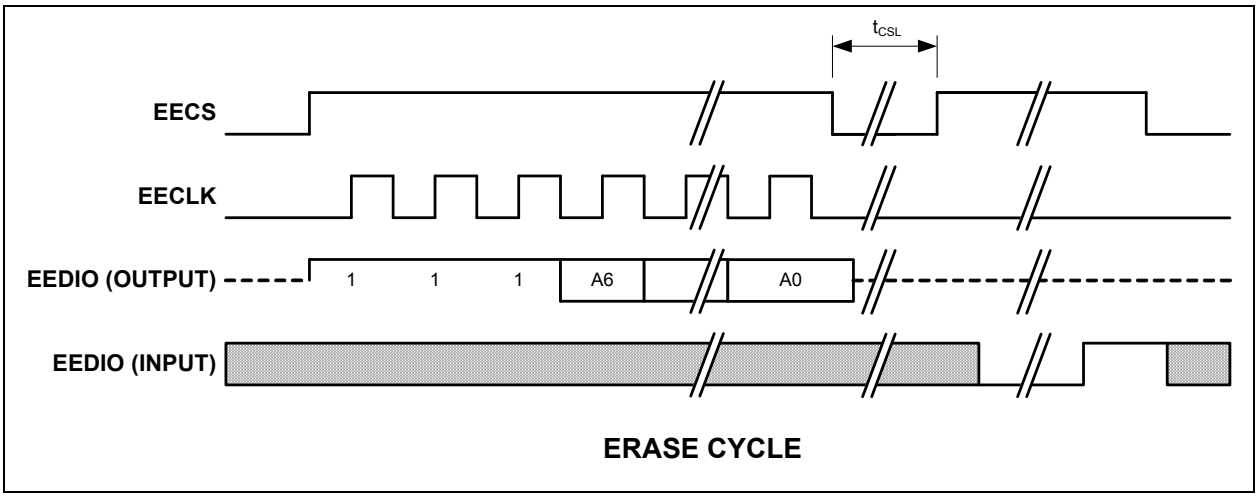


4.7.4.1 Supported EEPROM Operations

The EEPROM controller supports the following EEPROM operations under host control via the E2P_CMD register. The operations are commonly supported by “93C46” EEPROM devices. A description and functional timing diagram is provided below for each operation. Refer to the E2P_CMD register description in Section 6.3.12, "EEPROM Command Register (E2P_CMD)" for E2P_CMD field settings for each command.

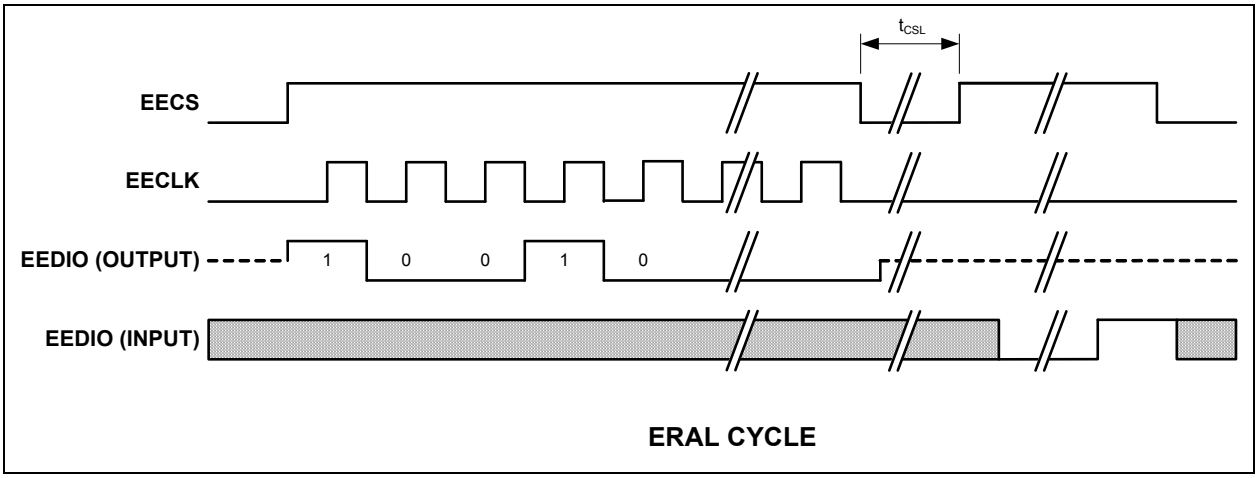
ERASE (Erase Location): If erase/write operations are enabled in the EEPROM, this command will erase the location selected by the EPC Address field (EPC_ADDR). The EPC_TO bit is set if the EEPROM does not respond within 30 ms.

FIGURE 4-21: EEPROM ERASE CYCLE



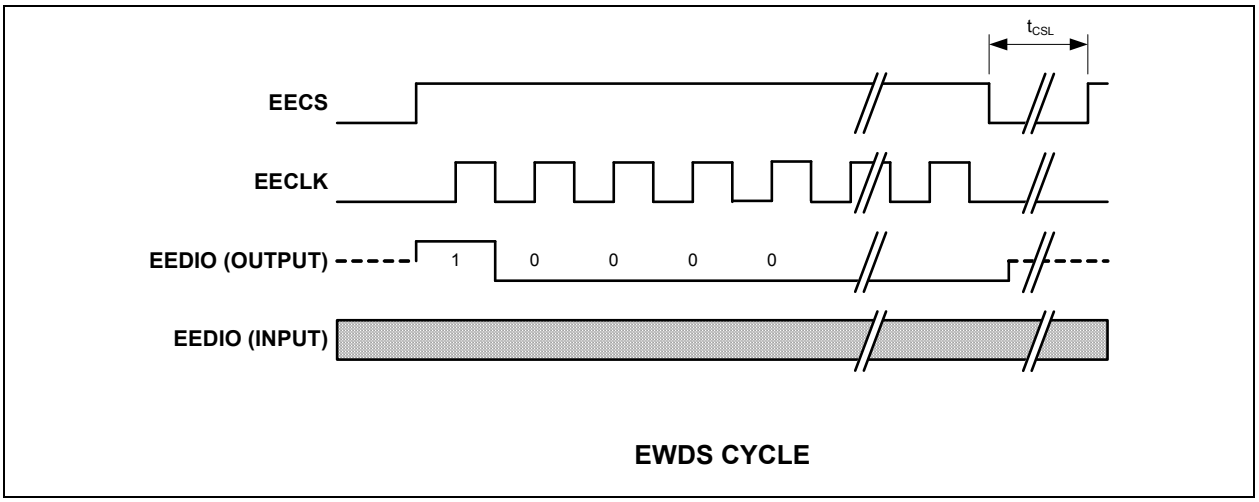
ERAL (Erase All): If erase/write operations are enabled in the EEPROM, this command will initiate a bulk erase of the entire EEPROM. The EPC_TO bit is set if the EEPROM does not respond within 30 ms.

FIGURE 4-22: EEPROM ERAL CYCLE



EWDS (Erase/Write Disable): After issued, the EEPROM will ignore erase and write commands. To re-enable erase/write operations issue the EWEN command.

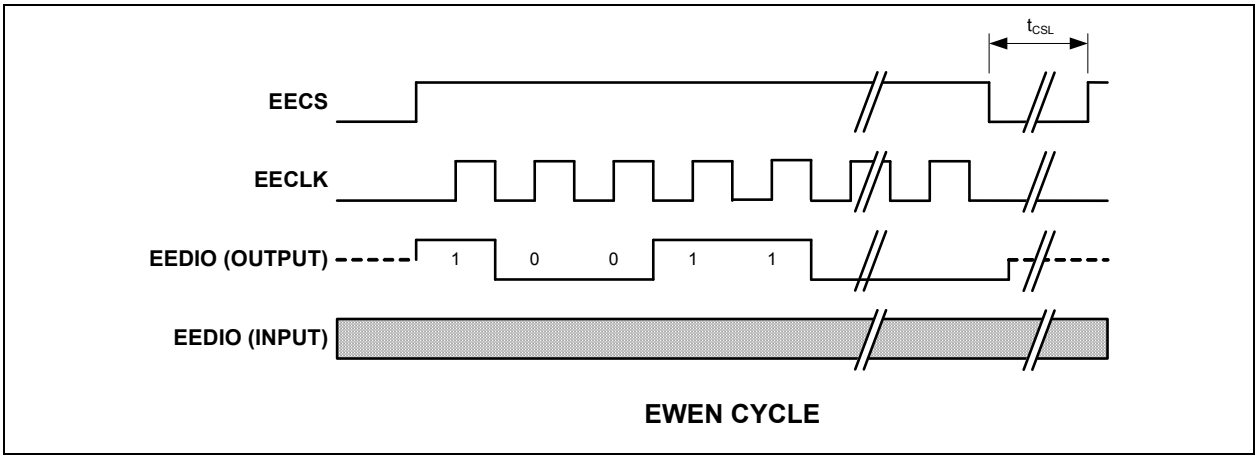
FIGURE 4-23: EEPROM EWDS CYCLE



EWEN (Erase/Write Enable): Enables the EEPROM for erase and write operations. The EEPROM will allow erase and write operations until the “Erase/Write Disable” command is sent, or until power is cycled.

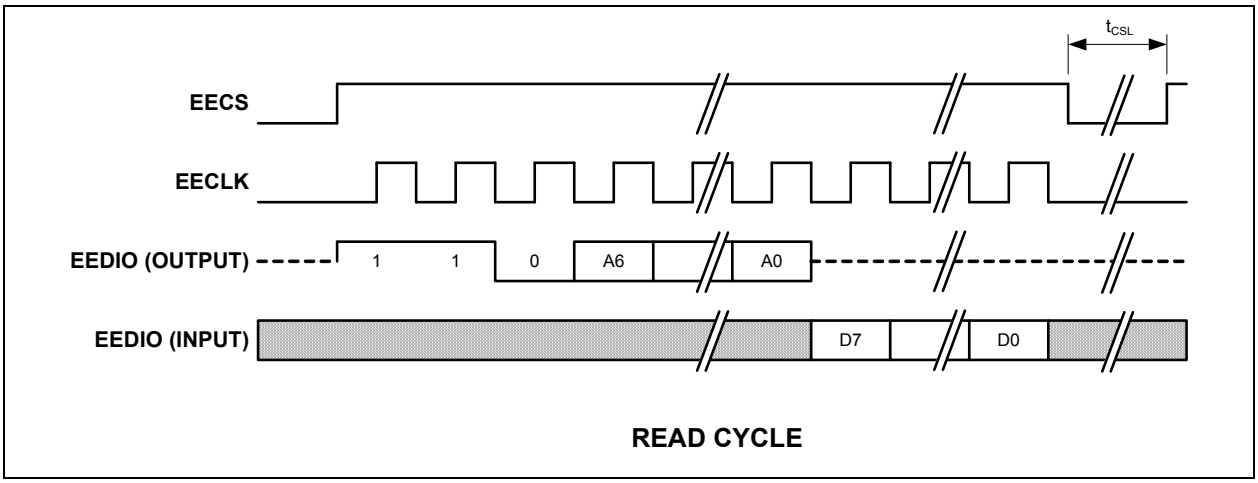
Note:	The EEPROM device will power-up in the erase/write-disabled state. Any erase or write operations will fail until an Erase/Write Enable command is issued.

FIGURE 4-24: EEPROM EWEN CYCLE



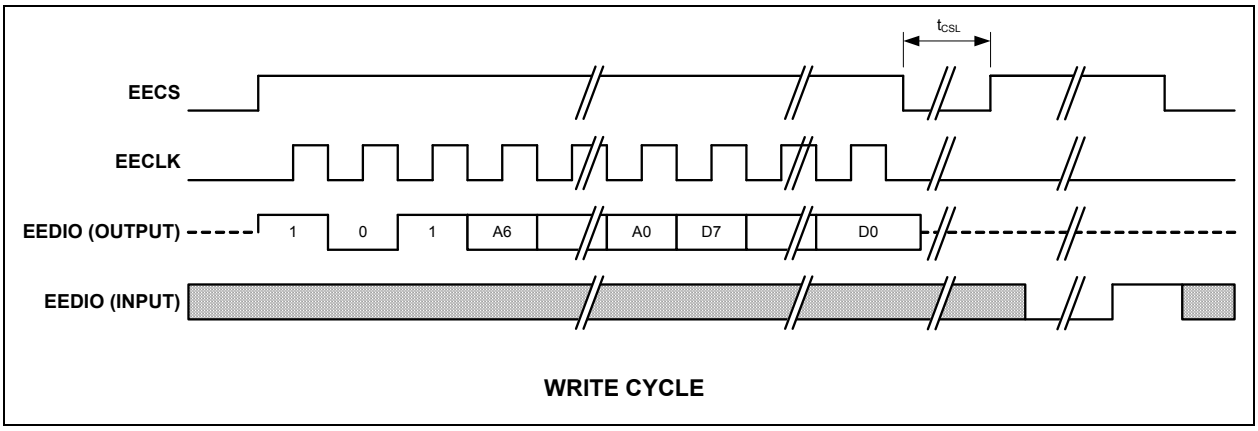
READ (Read Location): This command will cause a read of the EEPROM location pointed to by EPC Address (EPC_ADDR). The result of the read is available in the E2P_DATA register.

FIGURE 4-25: EEPROM READ CYCLE



WRITE (Write Location): If erase/write operations are enabled in the EEPROM, this command will cause the contents of the E2P_DATA register to be written to the EEPROM location selected by the EPC Address field (EPC_ADDR). The EPC_TO bit is set if the EEPROM does not respond within 30 ms.

FIGURE 4-26: EEPROM WRITE CYCLE



WRAL (Write All): If erase/write operations are enabled in the EEPROM, this command will cause the contents of the E2P_DATA register to be written to every EEPROM memory location. The EPC_TO bit is set if the EEPROM does not respond within 30 ms.

FIGURE 4-27: EEPROM WRAL CYCLE

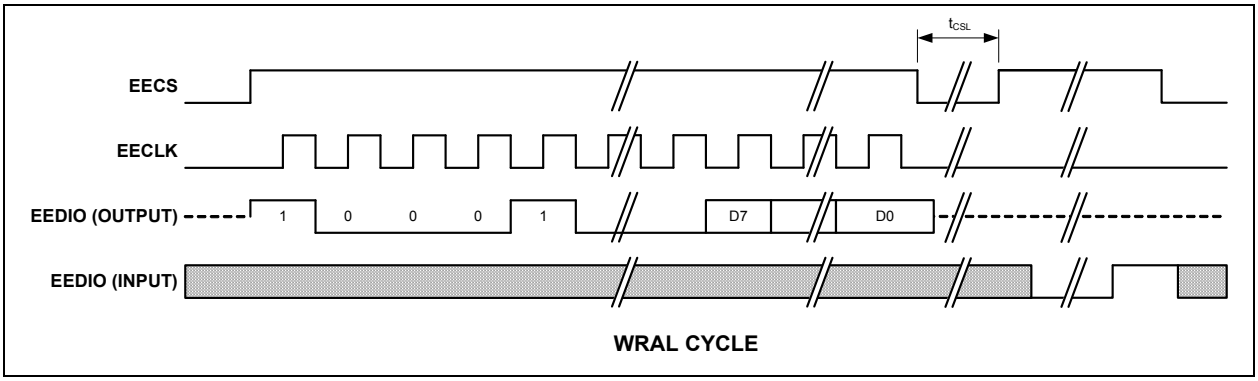


Table 4-60, "Required EECLK Cycles", shown below, shows the number of EECLK cycles required for each EEPROM operation.

TABLE 4-60: REQUIRED EECLK CYCLES

Operation	Required EECLK Cycles
ERASE	10
ERAL	10
EWDS	10
EWEN	10
READ	18
WRITE	18
WRAL	18

4.7.4.2 Host Initiated EEPROM Reload

The host can initiate a reload of the EEPROM by issuing the RELOAD command via the E2P Command (E2P_CMD) register. If the first byte read from the EEPROM is not 0xA5, it is assumed that the EEPROM is not present or not programmed, and the reload will fail. The Data Loaded bit of the E2P_CMD register indicates a successful reload of the EEPROM.

Note:	It is not recommended to use the RELOAD command as part of normal operation, as race conditions can occur with USB Commands that access descriptor data. It is best for the host to issue an SRST to reload the EEPROM data.

4.7.4.3 EEPROM Command and Data Registers

Refer to Section 6.3.12, "EEPROM Command Register (E2P_CMD)" and Section 6.3.13, "EEPROM Data Register (E2P_DATA)" for a detailed description of these registers. Supported EEPROM operations are described in these sections.

4.7.4.4 EEPROM Timing

Refer to Section 7.5.5, "EEPROM Timing" for detailed EEPROM timing specifications.

4.7.5 EXAMPLE OF EEPROM FORMAT INTERPRETATION

Table 4-61 and Table 4-62 provide an example of how the contents of an EEPROM are formatted. Table 4-61 is a dump of the EEPROM memory (256-byte EEPROM), while Table 4-62 illustrates, byte by byte, how the EEPROM is formatted.

TABLE 4-61: DUMP OF EEPROM MEMORY

Offset Byte	Value
0000h	A5 12 34 56 78 9A BC 01
0008h	04 04 09 04 0A 11 12 16
0010h	10 1F 00 00 00 00 12 27
0018h	12 30 12 39 12 42 00 04
0020h	8A 00 0A 03 53 00 4D 00
0028h	53 00 43 00 12 03 4C 00
0030h	41 00 4E 00 39 00 37 00
0038h	33 00 30 00 10 03 30 00
0040h	30 00 30 00 35 00 31 00
0048h	32 00 33 00 12 01 00 02
0050h	FF 00 FF 40 24 04 30 97
0058h	00 01 01 02 03 01 09 02
0060h	27 00 01 01 00 A0 FA 09
0068h	04 00 00 03 FF 00 FF 00
0070h	12 01 00 02 FF 00 FF 40
0078h	24 04 30 97 00 01 01 02
0080h	03 01 09 02 27 00 01 01
0088h	00 A0 FA 09 04 00 00 03
0090h - 00FFh	FF 00 FF 00 .....

TABLE 4-62: EEPROM EXAMPLE - 256 BYTE EEPROM

EEPROM Address	EEPROM Contents (Hex)	Description
00h	A5	EEPROM programmed indicator
01h - 06h	12 34 56 78 9A BC	MAC Address 12 34 56 78 9A BC
07h	01	Full-Speed polling interval for Interrupt Endpoint (1 ms)
08h	04	Hi-Speed polling interval for Interrupt Endpoint (4 ms)
09h	04	Configuration Flags - The device is bus powered and supports remote wakeup, nSPD_LED = Speed Indicator, nLNKA_LED = Link and Activity Indicator, nFDX_LED = full duplex Link Indicator.
0Ah - 0Bh	09 04	Language ID Descriptor 0409h, English
0Ch	0A	Manufacturer ID String Descriptor Length (10 bytes)
0Dh	11	Manufacturer ID String Descriptor EEPROM Word Offset (11h); corresponds to EEPROM Byte Offset 22h
0Eh	10	Product Name String Descriptor Length (16 bytes)
0Fh	16	Product Name String Descriptor EEPROM Word Offset (16h); corresponds to EEPROM Byte Offset 2Ch
10h	10	Serial Number String Descriptor Length (16 bytes)
11h	1E	Serial Number String Descriptor EEPROM Word Offset (1Eh); corresponds to EEPROM Byte Offset 3Ch
12h	00	Configuration String Descriptor Length (0 bytes - NA)

TABLE 4-62: EEPROM EXAMPLE - 256 BYTE EEPROM (CONTINUED)

EEPROM Address	EEPROM Contents (Hex)	Description
13h	00	Configuration String Descriptor Word Offset (don't care)
14h	00	Interface String Descriptor Length (0 bytes - NA)
15h	00	Interface String Descriptor Word Offset (don't care)
16h	12	Hi-Speed Device Descriptor Length (18 bytes)
17h	26	Hi-Speed Device Descriptor Word Offset (26h); corresponds to EEPROM Byte Offset 4Ch
18h	12	Hi-Speed Configuration and Interface Descriptor Length (18 bytes)
19h	2F	Hi-Speed Configuration and Interface Descriptor Word Offset (2Fh); corresponds to EEPROM Byte Offset 5Eh
1Ah	12	Full-Speed Device Descriptor Length (18 bytes)
1Bh	38	Full-Speed Device Descriptor Word Offset (38h); corresponds to EEPROM Byte Offset 70h
1Ch	12	Full-Speed Configuration and Interface Descriptor Length (18 bytes)
1Dh	41	Full-Speed Configuration and Interface Descriptor Word Offset (41h); corresponds to EEPROM Byte Offset 82h
1Eh	00	GPIO7:0 Wake Enables - GPIO7:0 not used for wakeup signaling
1Fh	04	GPIO10:8 Wake Enables - GPIO10 used for wakeup signaling
20h	8A	GPIO PME Flags - PME Signaling Enabled via Low Level, Push-Pull Driver, GPIO10 Active High Detection.
21h	00	PAD BYTE - Used to align following descriptor on Word boundary
22h	0A	Size of Manufacturer ID String Descriptor (10 bytes)
23h	03	Descriptor Type (String Descriptor - 03h)
24h - 2Bh	53 00 4D 00 53 00 43 00	Manufacturer ID String ("SMSC" in UNICODE)
2Ch	10	Size of Product Name String Descriptor (16 bytes)
2Dh	03	Descriptor Type (String Descriptor - 03h)
2Eh - 3Bh	4C 00 41 00 4E 00 39 00 37 00 33 00 30 00	Product Name String ("LAN9730" in UNICODE)
3Ch	10	Size of Serial Number String Descriptor (16 bytes)
3Dh	03	Descriptor Type (String Descriptor - 03h)
3Eh - 4Bh	30 00 30 00 30 00 35 00 31 00 32 00 33 00	Serial Number String ("0005123" in UNICODE)
4Ch	12	Size of Hi-Speed Device Descriptor in bytes (18 bytes)
4Dh	01	Descriptor Type (Device Descriptor - 01h)
4Eh - 4Fh	00 02	USB Specification Number that the device complies with (0200h)
50h	FF	Class Code
51h	00	Subclass Code
52h	FF	Protocol Code
53h	40	Maximum Packet Size for Endpoint 0
54h - 55h	24 04	Vendor ID (0424h)
56h - 57h	30 97	Product ID (9730h)
58h - 59h	00 01	Device Release Number (0100h)
5Ah	01	Index of Manufacturer String Descriptor
5Bh	02	Index of Product String Descriptor
5Ch	03	Index of Serial Number String Descriptor
5Dh	01	Number of possible configurations

TABLE 4-62: EEPROM EXAMPLE - 256 BYTE EEPROM (CONTINUED)

EEPROM Address	EEPROM Contents (Hex)	Description
5Eh	09	Size of Hi-Speed Configuration Descriptor in bytes (9 bytes)
5Fh	02	Descriptor Type (Configuration Descriptor - 02h)
60h - 61h	27 00	Total length in bytes of data returned (0027h = 39 bytes)
62h	01	Number of interfaces
63h	01	Value to use as an argument to select this configuration
64h	00	Index of String Descriptor describing this configuration
65h	A0	Bus powered and remote wakeup enabled
66h	FA	Maximum power consumption is 500 mA
67h	09	Size of Hi-Speed Interface Descriptor in bytes (9 bytes)
68h	04	Descriptor Type (Interface Descriptor - 04h)
69h	00	Number identifying this Interface
6Ah	00	Value used to select alternative setting
6Bh	03	Number of Endpoints used for this interface (Less Endpoint 0)
6Ch	FF	Class Code
6Dh	00	Subclass Code
6Eh	FF	Protocol Code
6Fh	00	Index of String Descriptor Describing this interface
70h	12	Size of Full-Speed Device Descriptor in bytes (18 bytes)
71h	01	Descriptor Type (Device Descriptor - 01h)
72h - 73h	00 02	USB Specification Number that the device complies with (0200h)
74h	FF	Class Code
75h	00	Subclass Code
76h	FF	Protocol Code
77h	40	Maximum Packet Size for Endpoint 0
78h - 79h	24 04	Vendor ID (0424h)
7Ah - 7Bh	30 97	Product ID (9730h)
7Ch - 7Dh	00 01	Device Release Number (0100h)
7Eh	01	Index of Manufacturer String Descriptor
7Fh	02	Index of Product String Descriptor
80h	03	Index of Serial Number String Descriptor
81h	01	Number of possible configurations
82h	09	Size of Full-Speed Configuration Descriptor in bytes (9 bytes)
83h	02	Descriptor Type (Configuration Descriptor - 02h)
84h - 85h	27 00	Total length in bytes of data returned (0027h = 39 bytes)
86h	01	Number of interfaces
87h	01	Value to use as an argument to select this configuration
88h	00	Index of String Descriptor describing this configuration
89h	A0	Bus powered and remote wakeup enabled
8Ah	FA	Maximum power consumption is 500 mA
8Bh	09	Size of Full-Speed Interface Descriptor in bytes (9 bytes)
8Ch	04	Descriptor Type (Interface Descriptor - 04h)
8Dh	00	Number identifying this interface
8Eh	00	Value used to select alternative setting

TABLE 4-62: EEPROM EXAMPLE - 256 BYTE EEPROM (CONTINUED)

EEPROM Address	EEPROM Contents (Hex)	Description
8Fh	03	Number of Endpoints used for this interface (Less Endpoint 0)
90h	FF	Class Code
91h	00	Subclass Code
92h	FF	Protocol Code
93h	00	Index of String Descriptor Describing this interface
94h - FFh	-	Data storage for use by host as desired

4.8 Customized Operation Without EEPROM

The device provides the capability to customize operation without the use of an EEPROM. Descriptor information and initialization quantities normally fetched from EEPROM and used to initialize descriptors and elements of the System Control and Status Registers may be specified via an alternate mechanism. This alternate mechanism involves the use of the Descriptor RAM in conjunction with the Attributes registers and select elements of the System Control and Status Registers. The software device driver orchestrates the process by performing the following actions in the order indicated:

- Initialization of SCSR Elements in Lieu of EEPROM Load
- Attribute Register Initialization
- Descriptor RAM Initialization
- Enable Descriptor RAM and Flag Attribute Registers as Source
- Inhibit Reset of Select SCSR Elements

The following subsections explain these actions. The Attributes registers must be written prior to initializing the Descriptor RAM. Failure to do this will prevent the RMT_WKUP flag from being overwritten by the bmAttributes of the Configuration Descriptor.

4.8.1 INITIALIZATION OF SCSR ELEMENTS IN LIEU OF EEPROM LOAD

During EEPROM operation, the following register fields are initialized by the hardware using the values contained in the EEPROM. In the absence of an EEPROM, the software device driver must initialize these quantities:

- MAC Address High Register (ADDRH) and MAC Address Low Register (ADDRL)
- LED Select (LED_SEL) bit of the LED General Purpose IO Configuration Register (LED_GPIO_CFG)
- GPIO Wake 0-10 (GPIOWKn) field of the General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE)

4.8.2 ATTRIBUTE REGISTER INITIALIZATION

The Attributes registers are as follows:

- HS Descriptor Attributes Register (HS_ATTR)
- FS Descriptor Attributes Register (FS_ATTR)
- String Descriptor Attributes Register 0 (STRNG_ATTR0)
- String Descriptor Attributes Register 1 (STRNG_ATTR1)
- Flag Attributes Register (FLAG_ATTR)

All of these registers, with the exception of FLAG_ATTR, contain fields defining the lengths of the descriptors written into the Descriptor RAM. If the descriptor is not written into the Descriptor RAM, the associated entry in the Attributes register must be written as 0. Writing an erroneous or illegal length will result in untoward operation and unexpected results.

The Flag Attributes Register (FLAG_ATTR) provides the mechanism to initialize components of the Configuration Flags and GPIO PME Flags that are stand-alone and not part of any other System Control and Status Register. During EEPROM operation, the analogous fields in this register are read by the hardware from the EEPROM and are not available to the software for read-back or modification.

Note:	The software device driver must initialize these registers prior to initializing the Descriptor RAM.

Note:	The bmAttributes field of the HS and FS descriptors in Descriptor RAM (if present) must be consistent with the contents of the Flag Attributes Register (FLAG_ATTR).

4.8.3 DESCRIPTOR RAM INITIALIZATION

The Descriptor RAM contents are initialized using the Data Port registers. The Data Port registers are used to select the Descriptor RAM and write the descriptor elements into it. The Descriptor RAM is 512 bytes in length. Every descriptor written into the Descriptor RAM must be DWORD aligned. The Attributes registers discussed in Section 4.8.2 must be written with the length of the descriptors written into the Descriptor RAM. If a descriptor is not used, hence not written into Descriptor RAM, its length must be written as 0 into the associated Attribute register.

Note:	The Attributes registers must be initialized before the Descriptor RAM.

Note:	Address 0 of the Descriptor RAM is always reserved for the Language ID descriptor, even if it will not be supported.

The descriptors must be written in the following order, starting at address 0 of the RAM and observing the DWORD alignment rule:

- Language ID Descriptor
- Manufacturing String Descriptor (String Index 1)
- Product Name String Descriptor (String Index 2)
- Serial Number String Descriptor (String Index 3)
- Configuration String Descriptor (String Index 4)
- Interface String Descriptor (String Index 5)
- HS Device Descriptor
- HS Configuration Descriptor
- FS Device Descriptor
- FS Configuration Descriptor

An example of Descriptor RAM use is illustrated in Figure 4-28.

As in the case of descriptors specified in EEPROM, the following restrictions apply to descriptors written into Descriptor RAM:

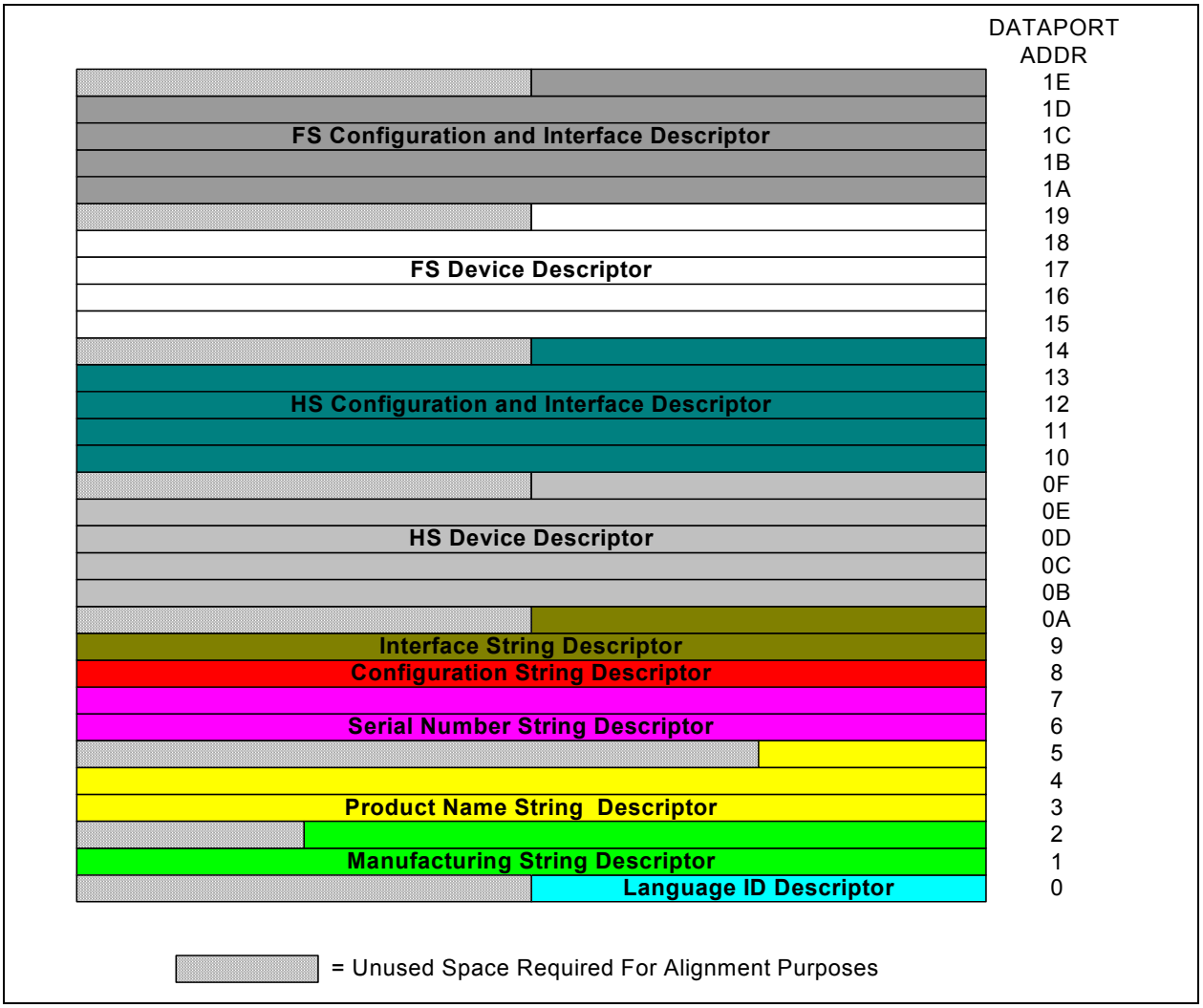
1. For Device Descriptors, the only valid values for the length are 0 and 18. The descriptor size for the Device Descriptors specified in the Descriptor RAM is a don't care and always overwritten by HW to 0x12 when transmitting the descriptor to the host.
2. The descriptor type for Device Descriptors specified in the Descriptor RAM is a don't care and is always overwritten by HW to 0x1 when transmitting the descriptor to the host.
3. For the Configuration and Interface Descriptor, the only valid values for the length are 0 and 18. The descriptor size for the Device Descriptors specified in the Descriptor RAM is a don't care and always overwritten by HW to 0x12 when transmitting the descriptor to the host.
4. The descriptor type for the Configuration Descriptors specified in the Descriptor RAM is a don't care and always overwritten by HW to 0x2 when transmitting the descriptor to the host.
5. If a String Descriptor does not exist in the Descriptor RAM, the referencing descriptor must contain 00h for the respective string index field.
6. If all String Descriptor lengths are zero than a Language ID will not be supported.

Note:	The first entry in the Descriptor RAM is always reserved for the Language ID descriptor, even if it will not be supported.

Note:	Descriptors specified having bcdUSB, bMaxPacketSize0, and bNumConfigurations fields defined with values other than 0200h, 40h, and 1, respectively, will result in unwanted behavior and untoward results.

The RAM Test Mode Enable (TESTEN) bit must be deasserted after programming the Descriptor RAM.

FIGURE 4-28: DESCRIPTOR RAM EXAMPLE



4.8.4 ENABLE DESCRIPTOR RAM AND FLAG ATTRIBUTE REGISTERS AS SOURCE

The EEPROM Emulation Enable (EEM) bit of the Hardware Configuration Register (HW_CFG) must be configured by the software device driver to use the Descriptor RAM and the Attributes registers for custom operation. Upon assertion of EEPROM Emulation Enable (EEM), the hardware will utilize the descriptor information contained in the Descriptor RAM, the Attributes registers, and the values of the items listed in Section 4.8.1 to facilitate custom operation.

4.8.5 INHIBIT RESET OF SELECT SCSR ELEMENTS

The software device driver must take care to ensure that the contents of the Descriptor RAM and SCSR register content critical to custom operation using Descriptor RAM are preserved across reset operations other than POR. The driver must configure the Reset Protection (RST_PROTECT) bit of the Hardware Configuration Register (HW_CFG) in order to accomplish this.

The following registers have contents that can be preserved across all resets other than POR. Consult the register's description for additional details.

- Descriptor RAM
- Attributes registers
- MAC Address High Register (ADDRH) and MAC Address Low Register (ADDRL)
- Hardware Configuration Register (HW_CFG)
- LED General Purpose IO Configuration Register (LED_GPIO_CFG)
- General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE)

4.9 Device Clocking

The device requires a fixed-frequency 25 MHz clock source. This is typically provided by attaching a 25 MHz crystal to the XI and XO pins. The clock can optionally be provided by driving the XI input pin with a single-ended 25 MHz clock source. If a single-ended source is selected, the clock input must run continuously for normal device operation.

Internally, the device generates its required clocks with a phase-locked loop (PLL). It reduces its power consumption in several of its operating states by disabling its internal PLL and derivative clocks. The 25 MHz clock remains operational in all states where power is applied.

4.10 Device Power Sources

The device may be soft powered by the USB bus or self powered via external power supplies. The following external 3.3 V power supplies are required when power is not being furnished by the USB bus:

- VDD33IO, VDD33A

The device includes an internal 1.2 V regulator which provides power to the internal core logic. This regulator may be optionally disabled if an external 1.2 V power source is available. The internal core regulator is controlled via the CORE_REG_EN pin. When disabled, +1.2 V must be supplied to the device by an external source. Refer to Chapter 3.0 for information on proper power connections when enabling or disabling the internal core regulator.

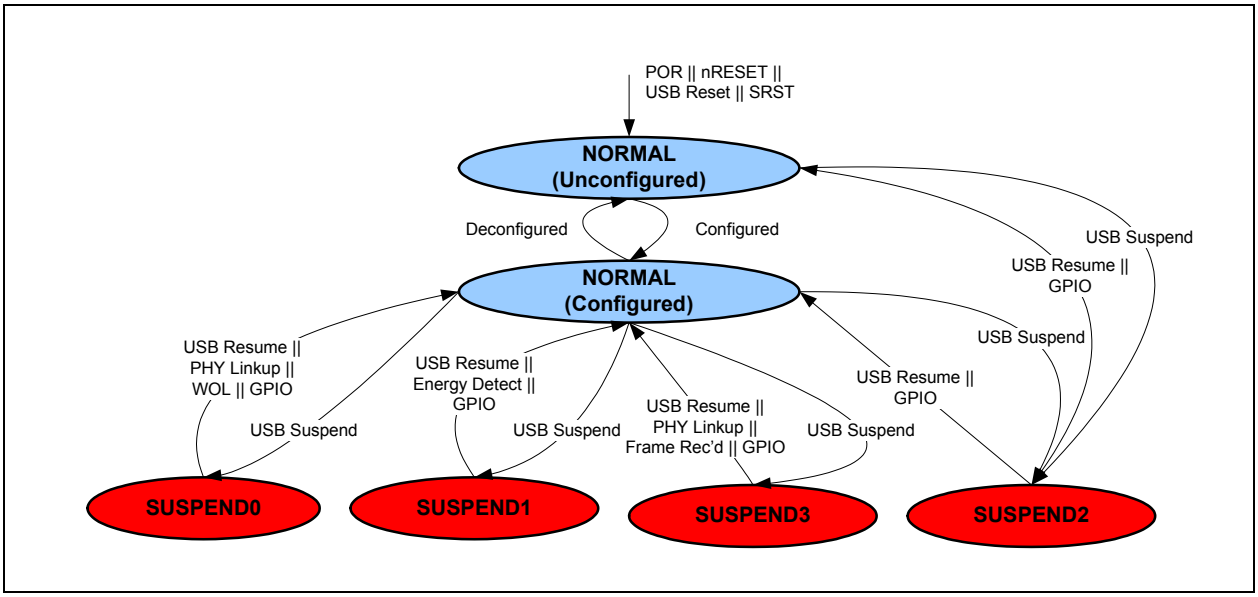
4.11 Power States

The following power states are featured.

- NORMAL (Unconfigured and Configured)
- Suspend (SUSPEND0, SUSPEND1, SUSPEND2, and SUSPEND3)

Figure 4-29 illustrates the power states and allowed state transitions.

FIGURE 4-29: POWER STATES



Note:	It is not possible to transition from SUSPEND2 to NORMAL Configured if SUSPEND2 was entered via a transition from NORMAL Unconfigured.

4.11.1 NORMAL STATE

The NORMAL state is the fully functional state of the device. The are two flavors of the NORMAL state, NORMAL Configured and NORMAL Unconfigured. In the Configured variation, all chip subsystem modules are enabled. The Unconfigured variation has only a subset of the modules enabled. The reduced functionality allows for power savings.

This NORMAL state is entered by any of the following methods:

- A system reset is asserted.
- The device is suspended and the host issues resume signaling.
- The device is suspended and a wake event is detected.

4.11.1.1 Unconfigured

Upon initially entering the NORMAL state, the device is unconfigured. The device transitions to the NORMAL Configured state upon the host completion of the USB configuration.

It is possible for the device to be deconfigured by the host after being placed in the NORMAL Configured state, via a set_configuration command. In this case, the CPM must place the device back into the NORMAL Unconfigured state.

4.11.1.2 Reset Operation

After a system reset, the device is placed into the NORMAL Unconfigured state. When in the NORMAL state, the READY bit in the Power Management Control Register (PMT_CTL) is set. This READY bit is useful to the host after a USB reset occurs. In this case, it indicates that the values in the EEPROM have been completely loaded.

4.11.1.3 Suspend Operation

When returning to the NORMAL state from the SUSPEND state, the USB context is maintained. After entering the NORMAL state, the READY bit in the PMT_CTL register is asserted.

Note:	If the originating SUSPEND state is SUSPEND2, the host is required to reinitialize the Ethernet PHY registers.

4.11.2 SUSPEND STATES

The SUSPEND state is entered after the USB host suspends the device. Four variations of the USB SUSPEND state are available. Each state offers different options in terms of power consumption and wakeup support.

A SUSPEND state is entered via a transition from the NORMAL state. The SUSPEND_MODE field in the Power Management Control Register (PMT_CTL) indicates which SUSPEND state is to be used. The host sets the value of this field to select the desired SUSPEND state, then sends suspend signaling. A transfer back to the NORMAL state occurs when the host sends resume signaling or a wakeup event is detected.

The device can be suspended from the NORMAL Unconfigured state. In this scenario, it is only possible to transition to the SUSPEND2 state. Subsequent resume signaling or a wakeup event will cause the device to transition back to the NORMAL Unconfigured state.

Note:	If the device is deconfigured, the SUSPEND_MODE field in the Power Management Control Register (PMT_CTL) resets to 10b.

4.11.2.1 Reset from Suspend

All suspend states must respond to a USB Reset and pin reset, nRESET. The application of these resets result in the device's hardware being re-initialized and placed into the NORMAL Unconfigured state.

4.11.2.2 SUSPEND0

This state is entered from the NORMAL state when the device is suspended and the SUSPEND_MODE field in the Power Management Control Register (PMT_CTL) is set to 00b.

Refer to Section 4.12.2.1, "Enabling GPIO Wake Events", Section 4.12.2.2, "Enabling WOL Wake Events" and Section 4.12.2.4, "Enabling External PHY Link Up Wake Events" for detailed instructions on how to program events that cause resumption from the SUSPEND0 state.

In this state, the MAC can optionally be programmed to detect a Wake On LAN event or Magic Packet event.

GPIO events can be programmed to cause wakeup in this state. If GPIO7 signals the event, the PHY Link Up Enable (PHY_LINKUP_EN) bit of the General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE) may be examined to determined whether a PHY Link Up event or Pin event occurred.

The host may take the device out of the SUSPEND0 state at any time.

4.11.2.3 SUSPEND1

This state is entered from the NORMAL state when the device is suspended and the SUSPEND_MODE field in the Power Management Control Register (PMT_CTL) is set to 01b.

Refer to Section 4.12.2.1, "Enabling GPIO Wake Events", and Section 4.12.2.3, "Enabling Link Status Change (Energy Detect) Wake Events" for detailed instructions on how to program events that cause resumption from the SUSPEND1 state.

In this state, the Ethernet PHY can be optionally programmed for energy detect. GPIO events can also be programmed to cause wakeup in this state.

The host may take the device out of the SUSPEND1 state at any time.

4.11.2.4 SUSPEND2

This state is entered from the NORMAL state when the device is suspended and the SUSPEND_MODE field in the Power Management Control Register (PMT_CTL) is set to 10b. SUSPEND2 is the default suspend mode.

Refer to Section 4.12.2.1, "Enabling GPIO Wake Events" for detailed instructions on how to program events that cause resumption from the SUSPEND2 state.

This state consumes the least amount of power. In this state, the device may only be awakened by the host or GPIO assertion.

The state of the Ethernet PHY is lost when entering SUSPEND2. Therefore, host must reinitialize the PHY after the device returns to the NORMAL state.

4.11.2.5 SUSPEND3

This state is entered from the NORMAL state when the device is suspended and the SUSPEND_MODE field in the Power Management Control Register (PMT_CTL) is set to 11b.

Refer to Section 4.12.2.1, "Enabling GPIO Wake Events", Section 4.12.2.4, "Enabling External PHY Link Up Wake Events" and Section 4.12.2.5, "Enabling Good Frame Wake Events" for detailed instructions on how to program events that cause resumption from the SUSPEND3 state.

In this SUSPEND state, all clocks in the device are enabled and power consumption is similar to the NORMAL state. However, it allows for power savings in the host CPU, which greatly exceeds that of the device. The driver may place the device in this state after prolonged periods of not receiving any Ethernet traffic.

This state supports wakeup from GPIO assertion, PHY Link Up, and on reception of a frame passing the filtering constraints set by the MAC Control Register (MAC_CR). Due to the limited amount of RX FIFO buffering, it is possible that there will be frames lost when in this state, as the USB resume time greatly exceeds the buffering capacity of the FIFO.

The Wake On LAN bit of the Wakeup Status (WUPS) field of the Power Management Control Register (PMT_CTL) is used to signal wakeup due to reception of a frame passing the aforementioned filtering constraints. This bit, along with the GPIO [10:0] (GPIOx_INT) bits of the Interrupt Status Register (INT_STS), may be examined to determine the event(s) causing the wakeup. If GPIO7 is found to have caused the wakeup, the PHY Link Up Enable (PHY_LINK-UP_EN) bit of the General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE) may be examined to determined whether a PHY Link Up event or pin event occurred.

Note:	Wake On LAN events must not be enabled in the Wakeup Control and Status Register (WUCSR) while operating in the SUSPEND3 state. If any Wake On LAN event is enabled in WUCSR, all received frames will be dropped. The setting of the Wake On LAN Enable (WOL_EN) bit of the Power Management Control Register (PMT_CTL) is a “don’t care”.

Note:	The Wake On LAN bit of the Wakeup Status (WUPS) is used to signal both Wake On LAN events and wakeup from SUSPEND3 state due to reception of frames passing the filtering constraints set by the MAC Control Register (MAC_CR). In order to interpret the Wakeup Status (WUPS) without ambiguity, the software driver may examine the Suspend Mode (SUSPEND_MODE) field of the Power Management Control Register (PMT_CTL) to determine the SUSPEND state it is coming out of.

4.12 Wake Events

The following events can wake up/enable the device, depending on the power state.

- USB host Resume
- Wake On LAN (Wakeup Frame, Magic Packet, Perfect Destination Address Frame, and Broadcast Frame)
- Reception of a Good Frame - a frame received when no Wake On LAN events are enabled in the Wakeup Control and Status Register (WUCSR) that meets the filtering requirements configured in the MAC Control Register (MAC_CR).
- PHY Energy Detect
- PHY Link Up
- GPIO[10:0]

Table 4-63 illustrates the wake events permitted in each of the power states.

TABLE 4-63: POWER STATE/WAKE EVENT MAPPING

Power State	USB Host Resume Signaling	WOL	Good Frame	PHY Energy Detect	PHY Link Up	GPIO[10:0]
SUSPEND0	YES	YES	NO	NO	YES	YES
SUSPEND1	YES	NO	NO	YES	NO	YES
SUSPEND2	YES	NO	NO	NO	NO	YES
SUSPEND3	YES	NO	YES	NO	YES	YES

The occurrence of a GPIO wake event causes the corresponding bit in the Interrupt Status Register (INT_STS) to be set. Before suspending the device, the host must ensure that any pending wake events are cleared. Otherwise, the device will immediately be awakened after being suspended.

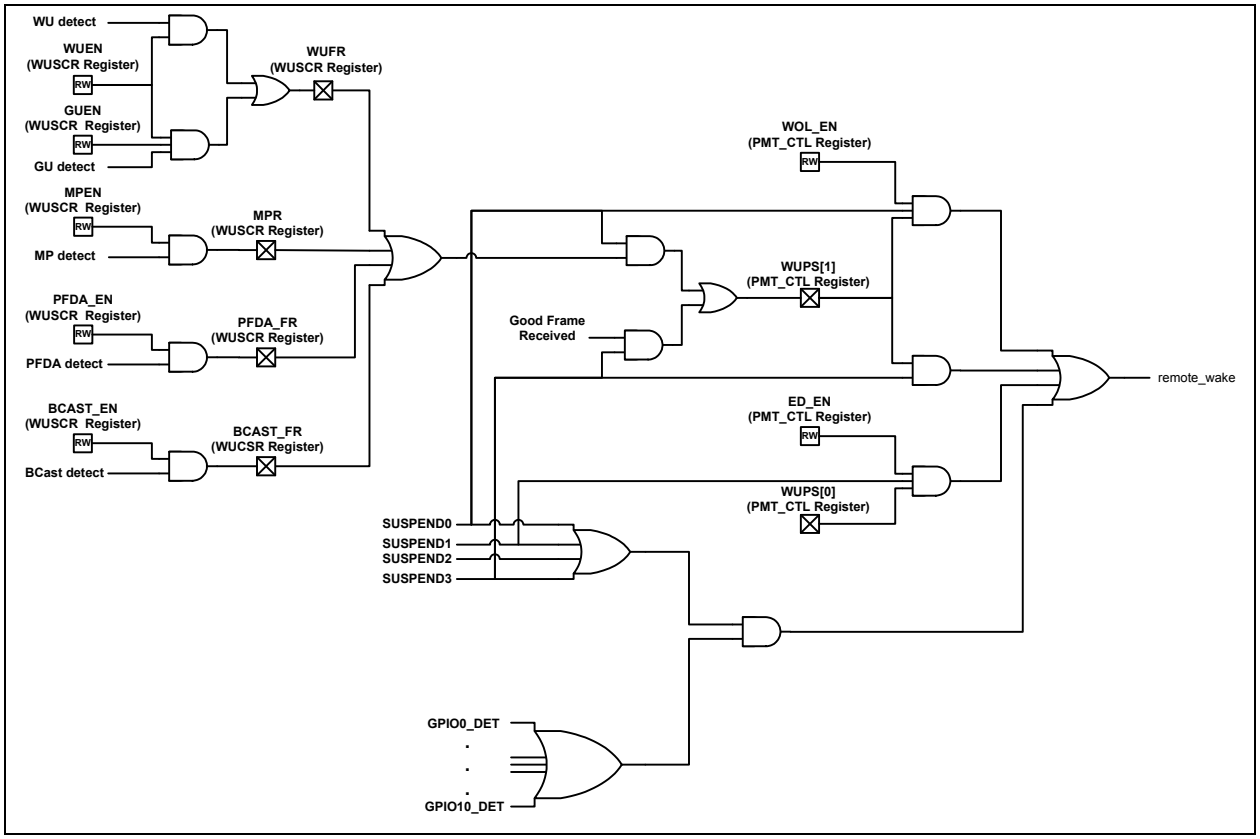
4.12.1 DETECTING WAKEUP EVENTS

The following sections illustrate and discuss the wakeup detection logic.

4.12.1.1 Wake Detection Logic

A simplified diagram of the wake event detection logic is shown in Figure 4-30.

FIGURE 4-30: WAKE EVENT DETECTION BLOCK DIAGRAM



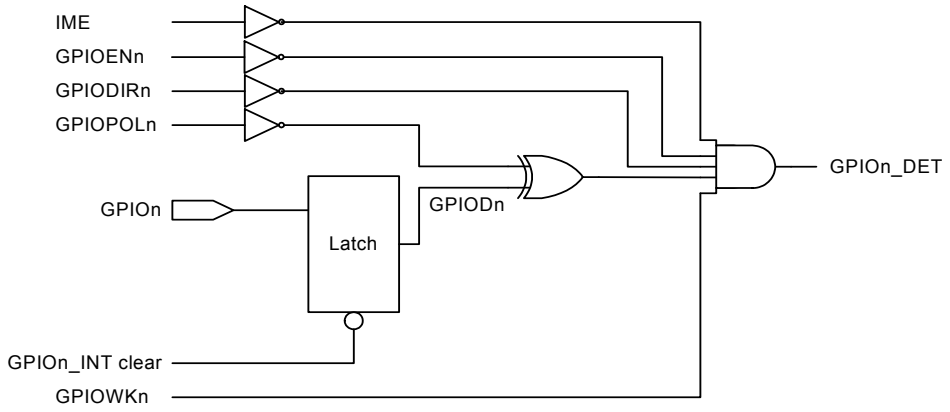
Note:	Diagram does not represent actual hardware implementation.

The functionality of GPIOs 0-6 and GPIOs 8-10 is slightly different. The functionality of GPIO7 is similar to that of GPIOs 0-6, with the additional requirement that it must cause a wakeup event when enabled for use in PHY Link Up detection.

Note:	GPIOs 0-7 are only available for use during internal Ethernet PHY mode of operation. The functionality of GPIOs 0-6 is depicted in Figure 4-31, while that of GPIO7 is shown in Figure 4-32.

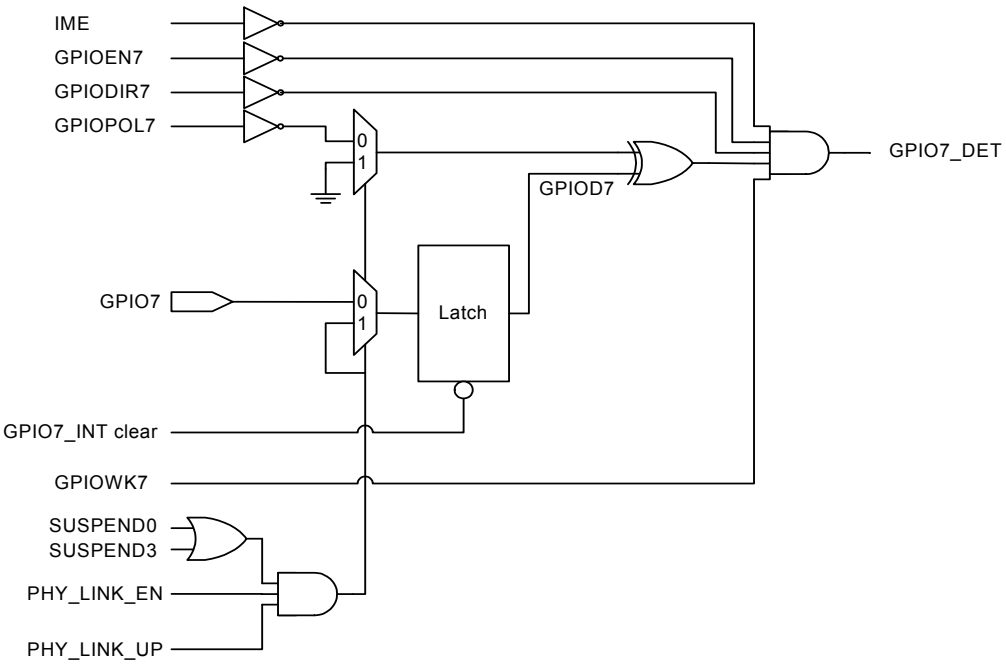
GPIOs 8-10 are available for use in both internal and external Ethernet PHY mode of operation. Their functionality is depicted in Figure 4-33.

FIGURE 4-31: DETAILED GPIOS 0-6 WAKE DETECTION LOGIC



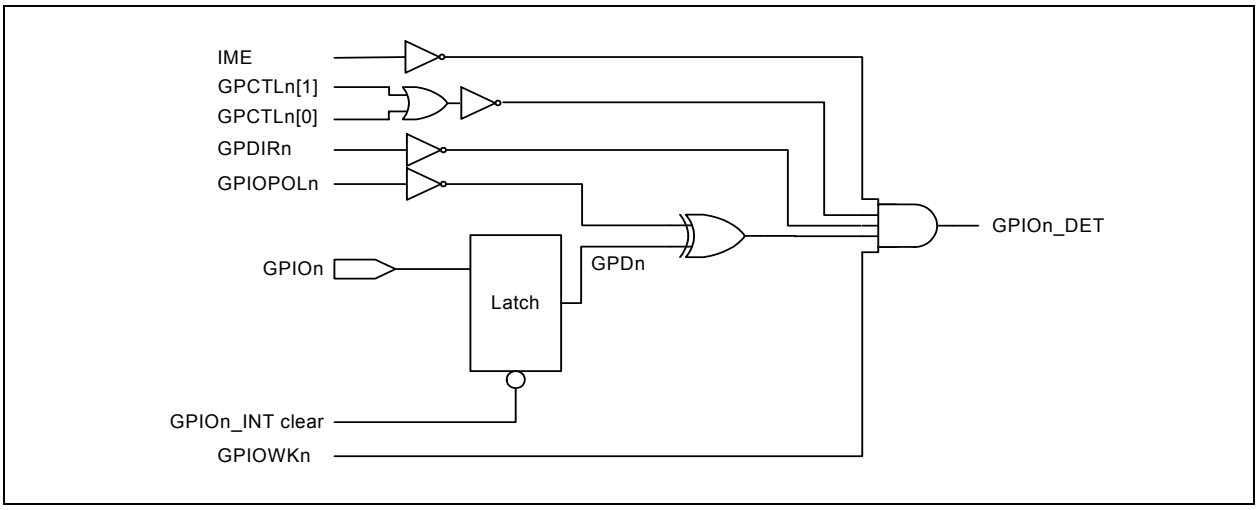
Note: The IME bit is in the Hardware Configuration Register (HW_CFG). General Purpose IO Configuration Register (GPIO_CFG) and General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE) must be set accordingly. Diagram does not represent actual hardware implementation.

FIGURE 4-32: DETAILED GPIO7 WAKE DETECTION LOGIC



Note:	The IME bit is in the Hardware Configuration Register (HW_CFG). General Purpose IO Configuration Register (GPIO_CFG) and General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE) must be set accordingly. PHY Link Up Enable (PHY_LINKUP_EN) bit of the General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE) must be set if PHY Link Up is used to cause a wake event. Diagram does not represent actual hardware implementation.

FIGURE 4-33: DETAILED GPIOS 8-10 WAKE DETECTION LOGIC



Note:	The IME bit is in the Hardware Configuration Register (HW_CFG). LED General Purpose IO Configuration Register (LED_GPIO_CFG) and General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE) must be set accordingly. Diagram does not represent actual hardware implementation.

4.12.1.2 Remote Wake Generation

4.12.1.2.1 Wake On LAN Event or Energy Detect

Control bits are implemented in the MAC's Wakeup Control and Status Register (WUCSR) to control Global Unicast Frame Wakeup, Magic Packet Wakeup, Wake Up Frame Detection Wakeup, Perfect DA Frame Wakeup, and Broadcast Frame Wakeup: GUEN, MPEN, WUEN, PFDA_EN, and BCAST_EN, respectively. A composite signal, depending on the state of these control bits and the associated event, is generated and propagated for further processing, as discussed in the following text.

Two control bits are implemented in the PMT_CTRL SCSR: Wake On LAN Enable (WOL_EN) and Energy Detect Enable (ED_EN). Depending on the state of these control bits, the logic will generate an internal wake event interrupt when the MAC detects a wakeup event (Global Unicast Frame, Wakeup Frame, Magic Packet, Perfect Destination Address Frame, or Broadcast Frame - depending on the state of the aforementioned composite signal), or a PHY interrupt is asserted (energy detect). Two Wakeup Status (WUPS) bits are implemented in the SCSR space. These bits are set depending on the corresponding wake event. (See Section 6.3.8, "Power Management Control Register (PMT_CTL)" for further information). If a Wake On LAN event is detected, then further resolution on the source of the event can be obtained by examining the Remote Wakeup Frame Received (WUFR), Magic Packet Received (MPR), Perfect DA Frame Received (PFDA_FR), and Broadcast Frame Received (BCAST_FR) status bits in the MAC's Wakeup Control and Status Register (WUCSR).

Note:	Wake On LAN events resulting in the generation of a remote-wake event may only occur when in SUSPEND0 state.

Note:	Energy Detect events resulting in the generation of a remote-wake event may only occur when in SUSPEND1 state.

Wakeup frame detection must be enabled in the MAC before detection can occur. Likewise, the energy detect interrupt must be enabled in the PHY before this interrupt can be used as a wake event. If the device is properly configured, the internal wake event interrupt will cause the assertion of the remote_wake signal on detection of a wake event.

4.12.1.2.2 Good Frame Detection

To wakeup on reception of a frame passing the filtering constraints set solely by the MAC Control Register (MAC_CR), the enables for all Wake On LAN events contained in the Wakeup Control and Status Register (WUCSR) must be cleared and the desired constraints must be selected in MAC_CR. The setting of the Wake On LAN Enable (WOL_EN) bit of the Power Management Control Register (PMT_CTL) is a “don’t care”. The logic will generate an internal wake event interrupt when the MAC detects a frame passing the filtering constraints (Good Frame). The Wake On LAN bit of the Wakeup Status (WUPS) field of the Power Management Control Register (PMT_CTL) is used to signal wakeup due to reception of the Good Frame.

Note:	Good Frame reception resulting in the generation of a remote_wake event may only occur when in the SUSPEND3 state.

4.12.1.2.3 GPIO Pin

GPIO pins 0 through 10 may cause the generation of a remote_wake event when properly configured and in any of the SUSPEND states. GPIO pins 0 through 7 each have a control bit (GPIOEN*x*, 0<=*x*<=7) in the General Purpose IO Configuration Register (GPIO_CFG) that is used to enable the GPIO pin to generate a remote_wake event. GPIO pins 8 through 10 have no specific enable bit. The corresponding enable signal for these pins (GPIOEN*y*, 8<=*y*<=10) is derived from the manner in which the pin is programmed. Ten GPIO wakeup status bits (GPIOWK*y*, 8<=*y*<=10) are available to determine the source of the event.

4.12.1.2.4 PHY Link Up

GPIO7 may be programmed to signal a wakeup in the SUSPEND0 or SUSPEND3 state on occurrence of a PHY Link Up. The PHY Link Up Enable (PHY_LINKUP_EN) bit of the General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE) must be set to use GPIO7 for this purpose. When used in this mode, the signal connected to the device’s pin is ignored.

4.12.2 ENABLING WAKE EVENTS

4.12.2.1 Enabling GPIO Wake Events

The host system must perform the following steps to enable the device to assert a remote_wake event on detection of a GPIO wake event.

1. The GPIO pin is programmed to facilitate generation of the wake event. If the pin is one of GPIO0 through GPIO7, the pin must be enabled to generate the event (GPIOENx must be clear in the General Purpose IO Configuration Register (GPIO_CFG)). If the pin is one of GPIO8 through GPIO10, the pin must be programmed as an input GPIO pin (the GPCTL and GPDIR fields for the pin in the LED General Purpose IO Configuration Register (LED_GPIO_CFG) must be set to 00b and 0, respectively). In addition, the pin must be enabled for wakeup and its desired polarity specified in the GPIO Wake 0-10 (GPIOWKn) and GPIO Polarity 0-10 (GPIOPOLn) fields, respectively, of the General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE).
2. The host places the device in the any one of the SUSPEND states by setting the Suspend Mode (SUSPEND_MODE) field of the Power Management Control Register (PMT_CTL) to indicate the desired SUSPEND state, then sends suspend signaling.

On detection of an enabled GPIO wake event, the device will transition back to the NORMAL state and signal a remote_wake event. The host may then examine the GPIO [10:0] (GPIOx_INT) status bits of the Interrupt Status Register (INT_STS) to determine the source of the wakeup.

4.12.2.2 Enabling WOL Wake Events

The host system must perform the following steps to enable the device to assert a remote_wake event on detection of a Wake on LAN event.

1. All transmit and receive operations must be halted:
   a) All pending Ethernet TX and RX operations must be completed.
   b) The MAC must be halted.
2. The MAC must be configured to detect the desired wake event. This process is explained in Section 4.5.5, "Wakeup Frame Detection" for Wakeup Frames and in Section 4.5.6, "Magic Packet Detection" for Magic Packets.
   Configuring Perfect DA and Broadcast Frame wakeup detection is analogous and requires the Perfect DA Wakeup Enable (PFDA_EN) or Broadcast Wakeup Enable (BCAST_EN) bit to be set in the Wakeup Control and Status Register (WUCSR).
3. Bit 1 of the Wakeup Status (WUPS[1]) in the Power Management Control Register (PMT_CTL) must be cleared since a set bit will cause the immediate assertion of wake event when the Wake On LAN Enable (WOL_EN) bit is set. The WUPS[1] bit will not clear if the internal MAC wakeup event is asserted.
4. Set the Wake On LAN Enable (WOL_EN) bit in the Power Management Control Register (PMT_CTL).
5. The host places the device in the SUSPEND0 state by setting the Suspend Mode (SUSPEND_MODE) field in the Power Management Control Register (PMT_CTL) to 00b, to indicate the desired SUSPEND state, then sends suspend signaling.

On detection of an enabled event, the device will transition back to the NORMAL state and signal a remote_wake event. The software will then examine the Suspend Mode (SUSPEND_MODE) field of the Power Management Control Register (PMT_CTL). Upon discovering wakeup occurred from SUSPEND0 state, the status bits of the WUCSR register may be examined to determine the particular event that caused the wakeup.

4.12.2.3 Enabling Link Status Change (Energy Detect) Wake Events

The host system must perform the following steps to enable the device to assert a remote_wake event on detection of an Ethernet link status change.

1. All transmit and receive operations must be halted:
   a) All pending Ethernet TX and RX operations must be completed.
   b) The MAC must be halted.
2. The PHY must be enabled for the Energy Detect Power-Down mode This is done by clearing the EDPWRDOWN bit in the PHY's Mode Control/Status Register. Enabling the Energy Detect Power-Down mode places the PHY in a reduced power state. In this mode of operation the PHY is not capable of receiving or transmitting Ethernet data. In this state, the PHY will assert its internal interrupt if it detects Ethernet activity. Refer to Section 4.6.8.2, "Energy Detect Power-Down (EDPD)" for more information.
3. Bit 0 of the Wakeup Status (WUPS[0]) in the Power Management Control Register (PMT_CTL) must be cleared, since a set bit will cause the immediate assertion of wake event when Energy-Detect Enable (ED_EN) is set. The WUPS[0] bit will not clear if the internal PHY interrupt is asserted.
4. Set the Energy-Detect Enable (ED_EN) bit in the Power Management Control Register (PMT_CTL).
5. The host places the device in the SUSPEND1 state by setting the Suspend Mode (SUSPEND_MODE) field in the Power Management Control Register (PMT_CTL) to 01b, to indicate the desired SUSPEND state, then sends suspend signaling.

On detection of Ethernet activity (energy), the device will transition back to the NORMAL state and signal a remote_wake event.

4.12.2.4 Enabling External PHY Link Up Wake Events

The host system must perform the following steps to enable the device to assert a remote_wake event on detection of PHY Link Up.

1. The system software determines that the link is down by periodically polling the Link Status bit of the Basic Status Register.

   Alternatively, the driver can detect assertion of the PHY_INT bit via the interrupt control Endpoint. The driver may also detect PHY interrupt assertion by polling the Interrupt Status Register (INT_STS). It then reads the Basic Status Register and finds the Link Status bit is deasserted.
2. On finding the link down, the host configures the device to wake up on PHY Link Up and signal the event using GPIO7 as follows:

   a) The PHY Link Up Enable (PHY_LINKUP_EN) bit is set in the General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE) to enable GPIO7 use in signaling the PHY Link Up event. The GPIOWK7 bit is also set in the register to permit its use in wake event generation. The setting of GPIOPOL7 is a “don’t care”.

   b) The following additional parameters for GPIO7 must be configured in the General Purpose IO Configuration Register (GPIO_CFG): GPIOEN7 = 0, GPIODIR7 = 0, GPIOBUF7 = “don’t care”.
3. The GPIO7_INT bit in the Interrupt Status Register (INT_STS) must be cleared, since a set bit will cause the immediate assertion of the wake event.
4. The host places the device in the SUSPEND0 or SUSPEND3 state, as appropriate, by setting the Suspend Mode (SUSPEND_MODE) field in the Power Management Control Register (PMT_CTL) to 00b or 11b, to indicate the desired SUSPEND state. The host then sends suspend signaling.

On detection of PHY Link Up, the device will transition back to the NORMAL state and signal a remote_wake event. The host, in trying to determine the cause of the wake event, may then examine the GPIO [10:0] (GPIOx_INT) status bits of the Interrupt Status Register (INT_STS). On finding GPIO7_INT set, the software will then use the Suspend Mode (SUSPEND_MODE) field of the Power Management Control Register (PMT_CTL) and the value of the PHY Link Up Enable (PHY_LINKUP_EN) bit to determine a PHY Link Up wake event occurred.

4.12.2.5 Enabling Good Frame Wake Events

The host system must perform the following steps to enable the device to assert a remote_wake event on detection of a Good Frame.

1. The MAC filtering is configured by setting the desired constraints in the MAC Control Register (MAC_CR). All Wake On LAN events contained in the Wakeup Control and Status Register (WUCSR) must be disabled. The setting of the Wake On LAN Enable (WOL_EN) bit of the Power Management Control Register (PMT_CTL) is a “don’t care”.
2. Bit 1 of the Wakeup Status (WUPS[1]) in the Power Management Control Register (PMT_CTL) must be cleared since a set bit will cause the immediate assertion of wake event. The WUPS[1] bit will not clear if the internal MAC wakeup event is asserted.
3. The host places the device in the SUSPEND3 state by setting the Suspend Mode (SUSPEND_MODE) field in the Power Management Control Register (PMT_CTL) to 11b, to indicate the desired SUSPEND state, then sends suspend signaling.

On detection of a Good Frame, the device will transition back to the NORMAL state and signal a remote_wake event. The software will then examine the Suspend Mode (SUSPEND_MODE) field of the Power Management Control Register (PMT_CTL). Upon discovering wakeup occurred from SUSPEND3 state, the host may perform desired processing as a result of receiving the Good Frame.

4.13 Resets

The device has the following chip level reset sources:

- Power on Reset (POR)
- External Chip Reset (nRESET)
- Lite Reset (LRST)
- Soft Reset (SRST)
- USB Reset
- PHY Software Reset
- nTRST

4.13.1 POWER ON RESET (POR)

A Power on Reset occurs whenever power is initially applied to the device, or if power is removed and reapplied to the device. A timer within the device will assert the internal reset for approximately 22 ms.

Note:	The EEPROM contents are loaded by this reset.

Note:	After the assertion of the POR, the internal Ethernet PHY is put into general power down mode.

4.13.2 EXTERNAL CHIP RESET (NRESET)

A hardware reset will occur when the nRESET pin is driven low. The READY bit in the PMT_CTRL register can be read by the host, and will read back a ‘0’ until the hardware reset is complete. Upon completion of the hardware reset, the READY bit in PMT_CTRL is set high.

After the READY bit is set, the device can be configured via its control registers. The nRESET pin is pulled-high internally by the device and can be left unconnected if unused. If used, nRESET must be driven low for a minimum period as defined in Section 7.5.4, "Reset and Configuration Strap Timing". If nRESET is unused, the device must be reset following power-up via a soft reset (SRST).

Note:	After the assertion of nRESET, the internal Ethernet PHY is put into General Power-Down mode.

4.13.3 LITE RESET (LRST)

This reset is initiated via the LRST bit in the Section 6.3.5, "Hardware Configuration Register (HW_CFG)". It will reset the entire chip with the exception of the USB Device Controller and the USB PHY (UDC, parts of the CTL, and the HSIC interface). The PLL is not turned off.

Note:	This reset does not cause the USB contents from the EEPROM to be reloaded.

Note:	This reset does not place the device into the Unconfigured state.

Note:	After the LRST, the USB pipes corresponding to the Bulk-In, Bulk-Out, and Interrupt Endpoints must be reset. This process entails clearing the device’s ENDPOINT_HALT feature and resetting the data toggle on the host side.

4.13.4 SOFT RESET (SRST)

A Soft reset is initiated by writing a ‘1’ to bit 0 of the HW_CFG register (SRST). This self-clearing bit will return to ‘0’ after approximately 2 µs, at which time the Soft Reset is complete. Soft reset does not clear control register bits marked as NASR.

Note:	The EEPROM contents are reloaded by this reset.

Note:	After the assertion of the SRST the internal Ethernet PHY is put into General Power-Down mode.

Writing SRST = 1 will cause the device to disconnect from the USB shortly after the first good OUT Data packet during the Data Phase. A brief delay will allow enough time for the device to send the ACK for the Data Stage, but the device will be disconnected (causing a 3-strikes time-out failure) for any next transaction (e.g., the Status Stage, or a repeated Data Stage, if there were any bus errors). To the USB host, the aforementioned behaviors are the same as what happens during any surprise removal of a USB Device. This behavior is completely normal, and a compliant host must be tolerant of it.

4.13.5 USB RESET

A USB reset causes a reset of the entire chip with the exception of the USB Device Controller and the HSIC interface (UDC, parts of the CTL, and the HSIC interface). The PLL is not turned off. It will occur after a POR, nRESET, or SRST (these will all force disconnects of the USB bus). After a USB reset, the READY bit in the PMT_CTRL register can be read by the host and will read back a ‘0’ until the EEPROM contents are loaded (provided one is present). Upon completion of the EEPROM contents load, the READY bit in PMT_CTRL is set high, and the device can be configured via its control registers.

Note:	This reset does not cause the USB contents from the EEPROM to be reloaded. Only the MAC address is reloaded.

Note:	After the assertion of the USB Reset the internal Ethernet PHY is put into General Power-Down mode.

4.13.6 PHY SOFTWARE RESET

The Ethernet PHY can be reset via two software-initiated resets. Refer to Section 4.6.9, "PHY Resets" for details.

4.13.7 NTRST

This active-low reset is used by the TAP controller.

4.14 Configuration Straps

Configuration straps are multi-function pins that are driven as outputs during normal operation. During a Power on Reset (POR) or a External Chip Reset (nRESET), these outputs are tri-stated. The high or low state of the signal is latched following de-assertion of the reset and is used to determine the default configuration of a particular feature. Configuration strap signals are noted in Chapter 2, "Pin Description and Configuration".

Configuration straps are latched as a result of a Power on Reset (POR) or a External Chip Reset (nRESET). Configuration straps include internal resistors in order to prevent the signal from floating when unconnected. If a particular configuration strap is connected to a load, an external pull-up or pull-down should be used to augment the internal resistor to ensure that it reaches the required voltage level prior to latching. The internal resistor can also be overridden by the addition of an external resistor.

Note:	The system designer must guarantee that configuration straps meet the timing requirements specified in Section 7.5.4, "Reset and Configuration Strap Timing" and Section 7.5.3, "Power-On Configuration Strap Valid Timing". If configuration straps are not at the correct voltage level prior to being latched, the device may capture incorrect strap values

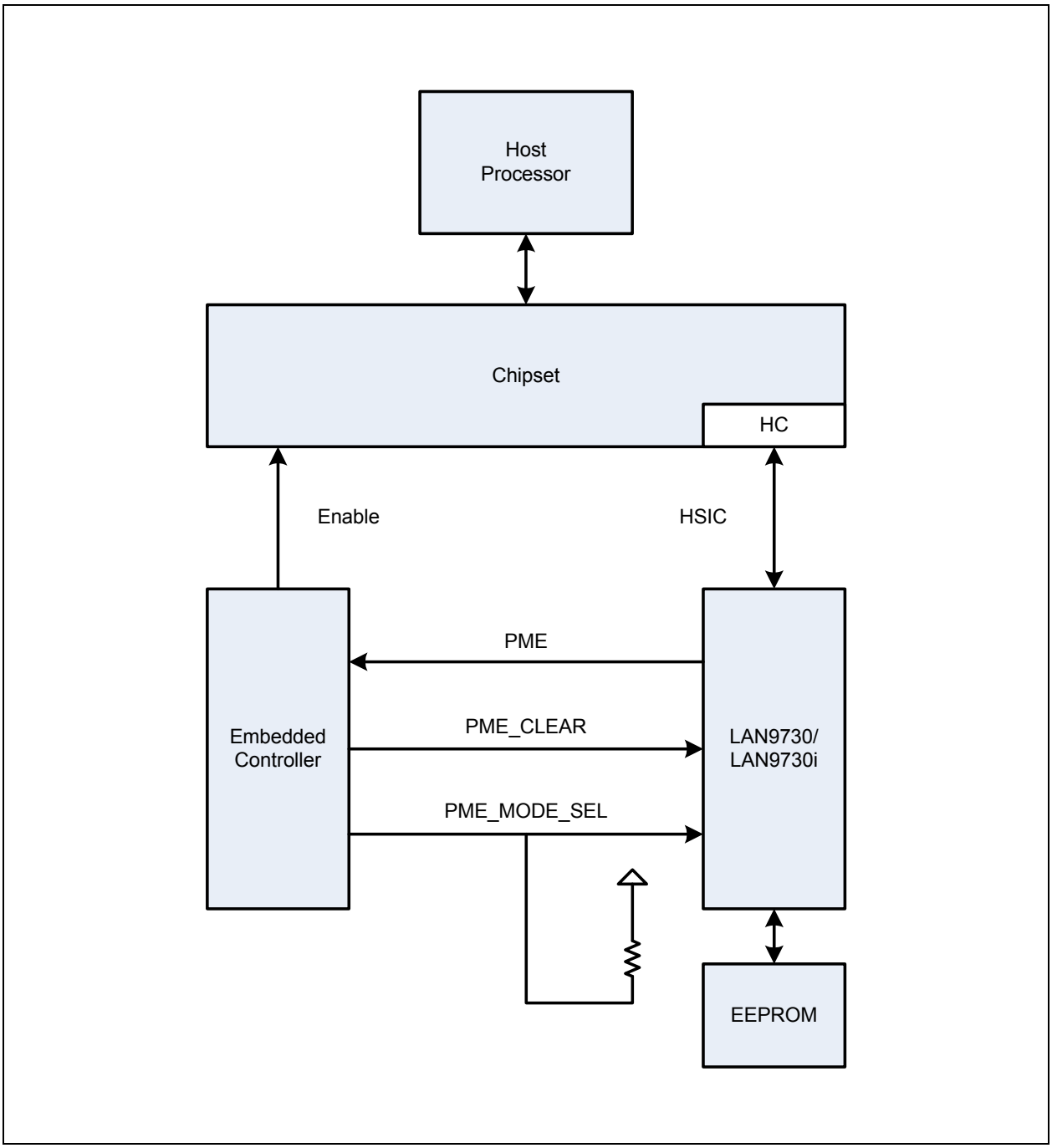
Note:	Configuration straps must never be driven as inputs. If required, configuration straps can be augmented, or overridden with external resistors.

NOTES:

5.0 PME OPERATION

The device provides a mechanism for waking up a host system via PME Mode of operation. PME signaling is only available while the device is operating in the self-powered mode. Figure 5-1 illustrates a typical application.

FIGURE 5-1: TYPICAL APPLICATION



The host processor is connected to a Chipset containing the USB host Controller (HC). The USB host Controller interfaces to the device via the HSIC USB signals. An Embedded Controller (EC) signals the Chipset and the host processor to power up via an Enable signal. The EC interfaces to the device via three signals. The PME signal is an input to the EC from the device that indicates the occurrence of a wakeup event. The PME_CLEAR (nRESET) signal is used to clear the PME. The PME_MODE_SEL signal is sampled by the device when PME_CLEAR (nRESET) is asserted and is used by the device to determine whether it should remain in PME Mode or resume normal operation.

GPIO pins are used for PME handling. The pins used depend on the value of the PHY_SEL pin, which determines PHY Mode of operation. In Internal PHY Mode of operation, GPIO0 is reserved for use as an output to signal the PME. GPIO1 is reserved for use as the PME_MODE_SEL input. GPIO8 and GPIO9 are reserved for analogous use, respectively, in External PHY Mode of operation.

The application scenario in Figure 5-1 assumes that the host processor and the Chipset are powered off, the EC is operational, and the device is in PME Mode, waiting for a wake event to occur. A wake event will result in the device signaling a PME event to the EC, which will then wake up the host processor and Chipset via the Enable signal. The EC sets PME_MODE_SEL to determine whether the device is to begin normal operation or continue in PME Mode, and asserts PME_CLEAR (nRESET) to clear the PME.

The following wake events are supported:

- Wakeup Pin(s)

The GPIO pins not reserved for PME handling have the capability to wake up the device when operating in PME mode. In order for a GPIO to generate a wake event, it must be configured as an input. GPIOs used as wake events must also be enabled by the GPIO_WAKE register, see Section 6.3.20, "General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE)". On POR or nRESET, all GPIOs default to inputs and the default value of the GPIO Wake 0-10 (GPIOWKn) field of the GPIO_WAKE register is set from the contents of the EEPROM. During PME mode of operation, the GPIOs used for signaling (GPIOs 0 and 1 or GPIOs 8 and 9) are not affected by the register defaults.

GPIO10 is available as a wakeup pin in External PHY mode, while GPIOs 2-10 are available in Internal PHY Mode. The GPIO10 Detection Select bit in the GPIO PME Flags byte of the EEPROM sets the detection mode for GPIO10 in both External and Internal PHY mode (if enabled via the GPIO_WAKE register), while GPIOs 2-9 are active low (by default) when operating in Internal PHY mode.

- Magic Packet

Reception of a Magic Packet when in PME mode will result in a PME being asserted.

- PHY Link Up

Detection of a PHY link partner when in PME mode will result in a PME being asserted.

In order to facilitate PME Mode of operation, the GPIO PME Enable bit in the GPIO PME Flags field, must be set and all remaining GPIO PME Flags field bits must be appropriately configured for pulse or level signaling, buffer type and GPIO PME WOL selection. The PME event is signaled on GPIO0 (External PHY Mode) or GPIO8, depending on the PHY Mode of operation.

The PME_MODE_SEL pin (GPIO1 in Internal Mode of operation, GPIO9 in External Mode of operation) must be driven to the value that determines whether or not the device remains in PME Mode of operation (1) or resumes normal operation (0) when the PME is recognized and cleared by the EC via PME_CLEAR (nRESET) assertion.

Note:	The device's software driver is unaware of PME mode. No internal mechanism exists for the driver to examine the internal hardware to determine the setting of the GPIO PME Flags read from the EEPROM on POR or nRESET. PME mode is not visible via the GPIO registers or via the INT_STS register. I.e., if a GPIO pin or reception of a Magic Packet results in a PME, the INT_STS register is not updated to indicate the occurrence of the event. The driver has no mechanism available to clear the PME. The driver can not program any GPIO register associated with the PME until the EC asserts nRESET to clear PME mode.

Note:	When in PME Mode, nRESET or POR will always cause the contents of the EEPROM to be reloaded.

Note:	GPIO10 may be used in PME and External PHY Mode to connect to an external PHY's Link LED, in order to generate a PHY Link Up wake event.

Figure 5-2 flowcharts PME operation while in Internal PHY Mode. The following conditions hold:

EEPROM Configuration:

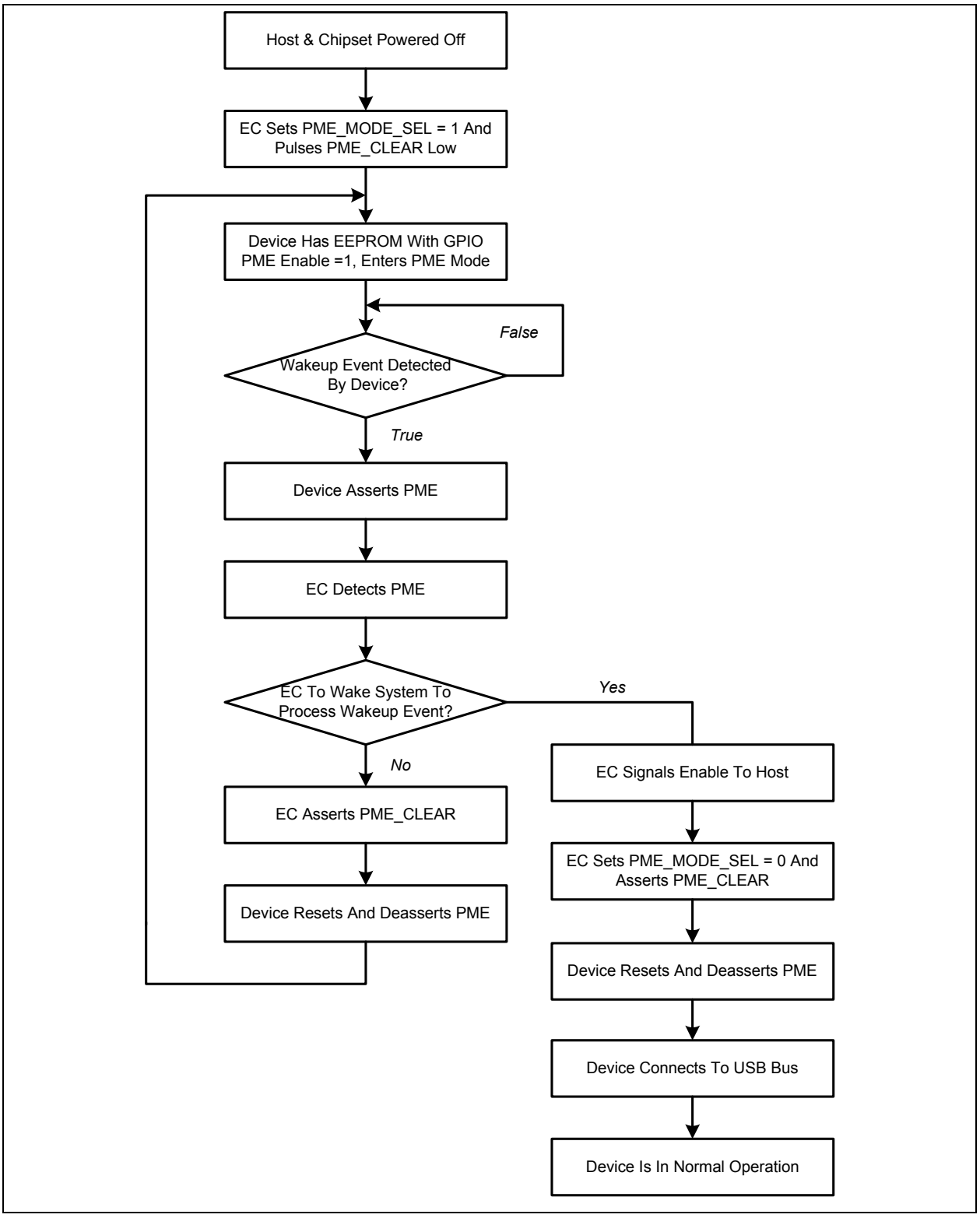
- GPIO PME Enable = 1 (enabled)
- GPIO PME Configuration = 0 (PME signaled via level on GPIO pin)
- GPIO PME Length = 0 (NA)
- GPIO PME Polarity = 1 (high level signals event)
- GPIO PME Buffer Type = 1 (push-pull)
- GPIO PME WOL Select = 0 (Magic Packet wakeup)
- GPIO10 Detection Select = 0 (Active-low detection)
- Power Method = 1 (self powered)
- MAC address for Magic Packet

PME signaling configuration (as determined by PHY Mode):

- GPIO0 signals PME
- GPIO1 is PME_MODE_SEL

Note: A POR occurring when PME_MODE_SEL = 1 and an EEPROM present with the GPIO PME Enable set results in the device entering PME Mode.

FIGURE 5-2: PME OPERATION



6.0 REGISTER DESCRIPTIONS

6.1 Register Nomenclature

Table 6-1 describes the register bit attributes used throughout this document.

TABLE 6-1: REGISTER BIT TYPES

Register Bit Type Notation	Register Bit Description
R	**Read:** A register or bit with this attribute can be read.
W	**Write:** A register or bit with this attribute can be written.
RO	**Read only:** Read only. Writes have no effect.
RS	**Read to Set:** This bit is set on read.
WO	**Write only:** If a register or bit is write-only, reads will return unspecified data.
WC	**Write One to Clear:** Writing a one clears the value. Writing a zero has no effect.
WAC	**Write Anything to Clear:** Writing anything clears the value.
RC	**Read to Clear:** Contents is cleared after the read. Writes have no effect.
LL	**Latch Low:** Clear on read of register.
LH	**Latch High:** Clear on read of register.
SC	**Self-Clearing:** Contents is self-cleared after the being set. Writes of zero have no effect. Contents can be read.
RO/LH	**Read Only, Latch High:** This mode is used by the Ethernet PHY registers. Bits with this attribute will stay high until the bit is read. After a read, the bit will remain high, but will change to low if the condition that caused the bit to go high is removed. If the bit has not been read, the bit will remain high regardless of if its cause has been removed.
NASR	**Not Affected by Software Reset.** The state of NASR bits does not change on assertion of a software reset.
RESERVED	**Reserved Field:** Reserved fields must be written with zeros, unless otherwise indicated, to ensure future compatibility. The value of reserved bits is not guaranteed on a read.

6.2 Register Memory Map

TABLE 6-2: REGISTER MEMORY MAP

Address	Symbol	Register Name
000h - 0FFh	SCSR	System Control and Status Registers
100h - 1FCh	MCSR	MAC Control and Status Registers

6.3 System Control and Status Registers

TABLE 6-3: DEVICE CONTROL AND STATUS REGISTER MAP

Address	Symbol	Register Name
000h	ID_REV	Device ID and Revision Register
004h	RESERVED	Reserved for future expansion
008h	INT_STS	Interrupt Status Register
00Ch	RX_CFG	Receive Configuration Register
010h	TX_CFG	Transmit Configuration Register
014h	HW_CFG	Hardware Configuration Register
018h	RX_FIFO_INF	Receive FIFO Information Register
01Ch	TX_FIFO_INF	Transmit FIFO Information Register
020h	PMT_CTL	Power Management Control Register
024h	LED_GPIO_CFG	LED General Purpose IO Configuration Register
028h	GPIO_CFG	General Purpose IO Configuration Register
02Ch	AFC_CFG	Automatic Flow Control Configuration Register
030h	E2P_CMD	EEPROM Command Register
034h	E2P_DATA	EEPROM Data Register
038h	BURST_CAP	Burst Cap Register
03Ch	RESERVED	Reserved for future expansion
040h	DP_SEL	Data Port Select Register
044h	DP_CMD	Data Port Command Register
048h	DP_ADDR	Data Port Address Register
04Ch	DP_DATA0	Data Port Data 0 Register
050h	DP_DATA1	Data Port Data 1 Register
054h – 060h	RESERVED	Reserved for future expansion
064h	GPIO_WAKE	General Purpose IO Wake Enable and Polarity Register
068h	INT_EP_CTL	Interrupt Endpoint Control Register
06Ch	BULK_IN_DLY	Bulk-In Delay Register
070h	DBG_RX_FIFO_LVL	Receive FIFO Level Debug Register
074h	DBG_RX_FIFO_PTR	Receive FIFO Pointer Debug Register
078h	DBG_TX_FIFO_LVL	Transmit FIFO Level Debug Register
07Ch	DBG_TX_FIFO_PTR	Transmit FIFO Pointer Debug Register
080h – 09Fh	RESERVED	Reserved for future expansion
0A0h	HS_ATTR	HS Descriptor Attributes Register
0A4h	FS_ATTR	FS Descriptor Attributes Register
0A8h	STRNG_ATTR0	String Descriptor Attributes Register 0
0ACh	STRNG_ATTR1	String Descriptor Attributes Register 1
0B0h	FLAG_ATTR	Flag Attributes Register
0B4h – 0FFh	RESERVED	Reserved for future expansion

6.3.1 DEVICE ID AND REVISION REGISTER (ID_REV)

Address: 000h Size: 32 bits

Bits	Description	Type	Default
31:16	**Chip ID** This read-only field identifies the device model.	RO	9730h
15:0	**Chip Revision** This is the revision of the device.	RO	Note 6-1

Note 6-1 Default value is dependent on device revision.

6.3.2 INTERRUPT STATUS REGISTER (INT_STS)

Address: 008h Size: 32 bits

Bits	Description	Type	Default
31:19	**RESERVED**	RO	-
18	**MAC Reset Time Out (MACRTO_INT)** This interrupt signifies that the 8 ms reset watchdog timer has timed out. This means that the Ethernet PHY is not supplying the rx_clk or tx_clk. After the timer times out, the MAC reset is deasserted asynchronously.	R/WC	0b
17	TX Stopped (TXSTOP_INT) This interrupt is asserted when the Stop Transmitter (STOP_TX) bit in Transmit Configuration Register (TX_CFG) is set and the transmitter is halted. **Note:** The source of this interrupt is a pulse.	R/WC	0b
16	**RX Stopped (RXSTOP_INT)** This interrupt is issued when the receiver is halted. **Note:** The source of this interrupt is a pulse.	R/WC	0b
15	**PHY Interrupt (PHY_INT)** Indicates a PHY Interrupt event. **Note:** Depending on configuration, this may report the interrupt status of the internal or the external PHY. **Note:** The source of this interrupt is a level. The interrupt persists until it is cleared in the PHY.	RO	-
14	**Transmitter Error (TXE)** When generated, indicates that the transmitter has encountered an error. Refer to Section 4.4.2.5, "TX Error Detection" for a description of the conditions that will cause a TXE. **Note:** The source of this interrupt is a pulse.	R/WC	0b
13	**TX Data FIFO Underrun Interrupt (TDFU)** Generated when the TX Data FIFO underruns. **Note:** The source of this interrupt is a pulse.	R/WC	0b
12	**TX Data FIFO Overrun Interrupt (TDFO)** Generated when the TX Data FIFO is full, and another write is attempted. **Note:** This interrupt should never occur and indicates a catastrophic hardware error. **Note:** The source of this interrupt is a pulse.	R/WC	0b
11	**RX Dropped Frame Interrupt (RXDF_INT)** This interrupt is issued whenever a receive frame is dropped. **Note:** The source of this interrupt is a pulse.	R/WC	0b
10:0	**GPIO [10:0] (GPIOx_INT)** Interrupts are generated from the GPIOs. These interrupts are configured through the GPIO_CFG and LED_GPIO_CFG registers. **Note:** The sources for these interrupts are a level.	R/WC Note 6-3	Note 6-2

Note 6-2 The default depends on the state of the GPIO pin.

Note 6-3 The clearing of a GPIOx_INT bit also clears the corresponding GPIO wake event.

6.3.3 RECEIVE CONFIGURATION REGISTER (RX_CFG)

Address: 00Ch Size: 32 bits

Bits	Description	Type	Default
31:1	**RESERVED**	RO	-
0	**Receive FIFO Flush** Setting this bit will reset the RX FIFO pointers.	SC	0b

6.3.4 TRANSMIT CONFIGURATION REGISTER (TX_CFG)

Address: 010h Size: 32 bits

Bits	Description	Type	Default
31:3	**RESERVED**	RO	-
2	**Transmitter Enable (TX_ON)** When this bit is set, the transmitter is enabled. Any data in the TX FIFO will be sent. This bit is cleared automatically when STOP_TX is set and the transmitter is halted.	R/W	0b
1	**Stop Transmitter (STOP_TX)** When this bit is set, the transmitter will finish the current frame being read from the TX FIFO, and will then stop transmitting. When the transmitter has stopped, this bit will clear. All writes to this bit are ignored while this bit is high. **Note:** After this bit clears, there will be no TX Ethernet frame data in the TX data path.	SC	0b
0	**Transmit FIFO Flush** Setting this bit will reset the TX FIFO pointers.	SC	0b

6.3.5 HARDWARE CONFIGURATION REGISTER (HW_CFG)

Address: 014h Size: 32 bits

Bits	Description	Type	Default
31:17	**RESERVED**	RO	-
16	**EEPROM Emulation Enable (EEM)** This bit is used to select the source of descriptor information and configuration flags when no EEPROM is present. 0 = Use defaults as specified in Section 4.7.2, "EEPROM Defaults" . 1 = Use Descriptor RAM and Attributes Registers. **Note:** This bit affects operation only when an EEPROM is not present. This bit has no effect when an EEPROM is present. **Note:** This field is protected by Reset Protection (RST_PROTECT).	R/W	0b
15	**Reset Protection (RST_PROTECT)** Setting this bit protects select fields of certain registers from being affected by resets other than POR. **Note:** This field is protected by Reset Protection (RST_PROTECT).	R/W	0b
14:13	**RESERVED**	RO	-
12	**Bulk-In Empty Response (BIR)** This bit controls the response to Bulk-In tokens when the RX FIFO is empty. 0 = Respond to the IN token with a ZLP 1 = Respond to the IN token with a NAK	R/W	0b
11	Activity LED 80 ms Bypass (LEDB) When set, the Activity LED on/off time is reduced to approximately 15 µs/ 15 µs.	R/W	0b
10:9	**RX Data Offset (RXDOFF)** This field controls the amount of offset, in bytes, that is added to the beginning of an RX Data packet. The start of the valid data will be shifted by the amount of bytes specified in this field. An offset of 0-3 bytes is a valid number of offset bytes. **Note:** This register may not be modified after the RX data path has been enabled.	R/W	00b
8	**Stall Bulk-Out Pipe Disable (SBP)** This bit controls the operation of the Bulk-Out pipe when the FCT detects the loss of sync condition. Refer to Section 4.4.2.5, "TX Error Detection" for details. 0 = Stall the Bulk-Out pipe when loss of sync detected. 1 = Do not stall the Bulk-Out pipe when loss of sync detected.	R/W	0b

Bits	Description	Type	Default
7	**Internal MII Visibility Enable (IME)** This register enables a subset of the MII interface to be visible on unused pins when configured for the internal Ethernet PHY mode. The pins controlled by the IME bit are comprised of the pins listed in Table 2-1, "MII Interface Pins" and the nPHY_INT pin. 0 = The MII signals are not visible. The MII pins function as inputs. 1 = The MII signals are visible. The MII pins function as outputs. **Note:** This register has no affect when using an external PHY. **Note:** The IME has priority over the GPIO_CFG register. When IME is asserted, the pins CRS, MDC, MDIO, COL, TXD3, TXD2, TXD1, and TXD0 can not be configured for GPIO operation.	RW	0b
6	**Discard Errored Received Ethernet Frame (DRP)** This bit will cause errored Ethernet frames to be discarded when enabled. 0 = Do not discard errored Ethernet frames. 1 = Discard errored Ethernet frames.	R/W	0b
5	**Multiple Ethernet Frames per USB Packet (MEF)** This bit enables the USB transmit direction to pack multiple Ethernet frames per USB packet whenever possible. 0 = Support no more than one Ethernet frame per USB packet. 1 = Support packing multiple Ethernet frames per USB packet. **Note:** The URX supports this mode by default.	R/W	0b
4	**EEPROM Time-out Control (ETC)** This bit controls the length of time used by the EEPOM controller to detect a time-out. 0 = Time-out occurs if no response received from EEPROM after 30 ms. 1 = Time-out occurs if no response received from EEPROM after 1.28 µs.	R/W	0b
3	**Soft Lite Reset (LRST)** Writing 1 generates the lite software reset of the device. A lite reset will not affect the UDC. Additionally, the contents of the EEPROM will not be reloaded. This reset will not cause the USB PHY to be disconnected. This bit clears after the reset sequence has completed.	SC	0b
2	**PHY Select (PSEL)** This bit indicates whether an internal or external Ethernet PHY is being used. 0 = Internal Ethernet PHY is used. 1 = External Ethernet PHY is used.	RO	Note 6-4
1	**Burst Cap Enable (BCE)** This register enables use of the burst cap register, Section 6.3.14, "Burst Cap Register (BURST_CAP)". 0 = Burst Cap register is not used to limit the TX burst size. 1 = Burst Cap register is used to limit the TX burst size.	R/W	0b

Bits	Description	Type	Default
0	**Soft Reset (SRST)** Writing 1 generates a software initiated reset of the device. If an external Ethernet PHY is used, it will be reset as well. A software reset will result in the contents of the EEPROM being reloaded. While the reset sequence is in progress, the USB PHY will be disconnected. After the device has been reinitialized, it will take the PHY out of the disconnect state and be visible to the host.	SC	0b

Note 6-4 The PHY_SEL pin determines the default value.

6.3.6 RECEIVE FIFO INFORMATION REGISTER (RX_FIFO_INF)

Address: 018h Size: 32 bits

Bits	Description	Type	Default
31:16	**RESERVED**	RO	-
15:0	**RX Data FIFO Used Space (RXDUSED)** Reads the amount of space in bytes used in the RX Data FIFO. For each receive frame, this field is incremented by the length of the receive data rounded up to the nearest DWORD (if the payload does not end on a DWORD boundary).	RO	0000h

6.3.7 TRANSMIT FIFO INFORMATION REGISTER (TX_FIFO_INF)

Address: 01Ch Size: 32 bits

Bits	Description	Type	Default
31:16	**RESERVED**	RO	-
15:0	**TX Data FIFO Free Space (TDFREE)** Reads the amount of space, in bytes, available in the TX Data FIFO.	RO	2000h

6.3.8 POWER MANAGEMENT CONTROL REGISTER (PMT_CTL)

Address: 020h Size: 32 bits

This register controls the power management features.

Bits	Description	Type	Default
31:10	**RESERVED**	RO	-
9	**Resume Clears Remote Wakeup Status (RES_CLR_WKP_STS)** When set, the Remote Wakeup Frame Received (WUFR), Magic Packet Received (MPR), Perfect DA Frame Received (PFDA_FR) and Broadcast Frame Received (BCAST_FR) status signals in the Wakeup Control and Status Register (WUCSR) will clear upon the completion of a resume sequence. When set, this bit also affects the WUPS field. WUPS[1] will clear upon completion of a resume event. When cleared, the wakeup status signals are not cleared after a resume.	R/W	0b
8	**Resume Clears Remote Wakeup Enables (RES_CLR_WKP_EN)** when asserted, all wakeup enable bits are cleared after a resume sequence, initiated from a remote wakeup, completes. Resumes initiated by the host do not clear the wakeup enables. **Note:** Global Unicast Wakeup Enable (GUEN), Perfect DA Wakeup Enable (PFDA_EN), Wakeup Frame Enable (WUEN), Magic Packet Enable (MPEN), and Broadcast Wakeup Enable (BCAST_EN) are not included.	R/W	1b
7	**Device Ready (READY)** When set, this bit indicates that the device is in the NORMAL state and the initial hardware configuration of the device has been completed. **Note:** This bit is useful for events (USB Reset) that do not trigger a soft disconnect. **Note:** In the case where no PHY clocks are present to complete a system reset, this bit will not be set until the watchdog timer expires. This is applicable for a Lite Reset and when transitioning to the Normal Configured state.	RO	0b
6:5	**Suspend Mode (SUSPEND_MODE)** Indicates which suspend power state to use after the host suspends the device. If the device is deconfigured, it transitions to the NORMAL Unconfigured state and this register will reset to the value 10b. SUSPEND_MODE encoding: 00 = SUSPEND0 01 = SUSPEND1 10 = SUSPEND2 11 = SUSPEND3 **Note:** It is not valid to select any suspend variant besides SUSPEND2 when in the NORMAL Unconfigured state.	R/W	10b

<table>
<tr><th>Bits</th><th>Description</th><th>Type</th><th>Default</th></tr>
<tr><td>4</td><td>PHY Reset (PHY_RST)
Writing a '1' to this bit resets the PHY. The internal logic automatically holds the PHY reset for a minimum of 2 ms. When the PHY is released from reset, this bit is automatically cleared. All writes to this bit are ignored while this bit is high.
Note: The device will NAK all USB transfers until the PHY reset completes.</td><td>SC</td><td>0b</td></tr>
<tr><td>3</td><td>Wake On LAN Enable (WOL_EN)
Enables WOL as a wakeup event.

This bit is automatically cleared at the completion of a resume sequence if Resume Clears Remote Wakeup Enables (RES_CLR_WKP_EN) is set.</td><td>R/W</td><td>0b</td></tr>
<tr><td>2</td><td>Energy-Detect Enable (ED_EN)
Enables Energy-Detect as a wakeup event.

This bit is automatically cleared at the completion of a resume sequence if Resume Clears Remote Wakeup Enables (RES_CLR_WKP_EN) is set.</td><td>R/W</td><td>0b</td></tr>
<tr><td>1:0</td><td>Wakeup Status (WUPS)
This field indicates the cause of the current wakeup event. The WUPS[0] and WUPS[1] bits are set and cleared independently. A WUPS bit may be cleared by writing a ‘1’ to the corresponding bit. The encoding of these bits (WUPS[1], WUPS[0]) is as follows:

<table>
<tr><th>WUPS[1:0]</th><th>Event</th></tr>
<tr><td>00</td><td>No wake-up event detected</td></tr>
<tr><td>X1</td><td>Energy-Detect (SUSPEND1)</td></tr>
<tr><td>1X</td><td>Wake On LAN (SUSPEND0) / Good Frame (SUSPEND3)</td></tr>
</table>

More than one bit may be set indicating that multiple events occurred.

The WUPS field will not be set unless the corresponding event is enabled prior to entering the reduced power state.

These bits will set regardless of the values in Wake On LAN Enable (WOL_EN) and Energy-Detect Enable (ED_EN).

If the Resume Clears Remote Wakeup Status (RES_CLR_WKP_STS) bit is set, WUPS[1] will clear upon completion of a resume. See the Resume Clears Remote Wakeup Status (RES_CLR_WKP_STS) bit for further details.
Note: It is not valid to simultaneously clear the WUPS bits and change the contents of the Suspend Mode (SUSPEND_MODE) field.</td><td>R/WC</td><td>00b</td></tr>
</table>

6.3.9 LED GENERAL PURPOSE IO CONFIGURATION REGISTER (LED_GPIO_CFG)

Address: 024h Size: 32 bits

This register configures the external GPIO[10:8] pins.

In order for a GPIO to function as a wake event or interrupt source, it must be configured as an input. GPIO pins used to generate wake events must also be enabled by the GPIO_WAKE register, see Section 6.3.20, "General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE)".

<table>
<tr><th>Bits</th><th>Description</th><th>Type</th><th>Default</th></tr>
<tr><td>31</td><td>LED Select (LED_SEL)
This bit determines the functionality of external LED pins.
<table>
<tr><th>Bit Value</th><th>Pin Name</th><th>Function</th></tr>
<tr><td rowspan="3">0</td><td>nSPD_LED</td><td>Speed Indicator</td></tr>
<tr><td>nLNKA_LED</td><td>Link and Activity Indicator</td></tr>
<tr><td>nFDX_LED</td><td>Full Duplex Link Indicator
Note: Hardware defaults to Activity Indicator. Software must manipulate to provide full-duplex indication.</td></tr>
<tr><td rowspan="3">1</td><td>nSPD_LED</td><td>Speed Indicator</td></tr>
<tr><td>nLNKA_LED</td><td>Link Indicator</td></tr>
<tr><td>nFDX_LED</td><td>Activity Indicator</td></tr>
</table>
Note: This field is protected by Reset Protection (RST_PROTECT).</td><td>R/W</td><td>Note 6-5</td></tr>
<tr><td>30:26</td><td>RESERVED</td><td>RO</td><td>-</td></tr>
<tr><td>25:24</td><td>GPIO 10 Control (GPCTL10)
The value of this field determines the function of the external GPIO10 pin as follows:
00 = GPIO10
01 = nSPD_LED (Ethernet speed indicator LED)
10 = RXD0
11 = RXD3
Note: When enabled as RXD0 or RXD3, the external device pin will reflect the state of the corresponding internal MII signal. This feature is useful as a diagnostic tool.</td><td>R/W</td><td>00b</td></tr>
<tr><td>23:22</td><td>RESERVED</td><td>RO</td><td>-</td></tr>
<tr><td>21:20</td><td>GPIO 9 Control (GPCTL9)
The value of this field determines the function of the external GPIO9 pin as follows:

00 = GPIO9
01 = Note 6-6
10 = RXD1
11 = nPHY_RST
Note: When enabled as RXD1 or nPHY_RST, the external device pin will reflect the state of the corresponding internal MII signal. This feature is useful as a diagnostic tool.</td><td>R/W</td><td>00b</td></tr>
<tr><td>19:18</td><td>RESERVED</td><td>RO</td><td>-</td></tr>
</table>

Bits	Description	Type	Default
17:16	**GPIO 8 Control (GPCTL8)** The value of this field determines the function of the external GPIO8 pin as follows: 00 = GPIO8 01 = Note 6-6 10 = RXD2 11 = CRS **Note:** When enabled as RXD2 or CRS, the external device pin will reflect the state of the corresponding internal MII signal. This feature is useful as a diagnostic tool.	R/W	00b
15:11	**RESERVED**	RO	-
10:8	**GPIO Buffer Type (GPBUF[10:8])** When set, the output buffer for the corresponding GPIO signal is configured as a push/pull driver. When cleared, the corresponding GPIO signal is configured as an open-drain driver. Bits are assigned as follows: GPBUF8 – bit 8 GPBUF9 – bit 9 GPBUF10 – bit 10	R/W	000b
7	**RESERVED**	RO	-
6:4	**GPIO Direction (GPDIR[10:8])** When set, enables the corresponding GPIO as an output. When cleared the GPIO is enabled as an input. Bits are assigned as follows: GPDIR8 – bit 4 GPDIR9 – bit 5 GPDIR10 – bit 6	R/W	000b
3	**RESERVED**	RO	-
2:0	**GPIO Data (GPD[10:8])** When enabled as an output, the value written is reflected on GPIOn. When read, GPIOn reflects the current state of the corresponding GPIO pin. Bits are assigned as follows: GPD8 – bit 0 GPD9 – bit 1 GPD10 – bit 2	R/W	Note 6-7

Note 6-5 The default value for this bit is 0 when no EEPROM is present. If an EEPROM is present, the default value is the value of the LED Select bit in the Configuration Flags of the EEPROM. A USB Reset or Lite Reset (LRST) will cause this bit to be restored to the image value last loaded from EEPROM, or to be set to 0 if no EEPROM is present.

Note 6-6 Determined by the LED Select (LED_SEL) setting.

Note 6-7 The default value depends on the state of the GPIO pin.

6.3.10 GENERAL PURPOSE IO CONFIGURATION REGISTER (GPIO_CFG)

Address: 028h Size: 32 bits

This register configures GPIOs 0-7. These GPIOs are not available when using external MII mode. See the PHY_SEL pin in Table 2-4, "Miscellaneous Pins".

In order for a GPIO to function as a wake event or interrupt source, it must be configured as an input. GPIOs used as wake events must also be enabled by the GPIO_WAKE register, see Section 6.3.20, "General Purpose IO Wake Enable and Polarity Register (GPIO_WAKE)".

Bits	Description	Type	Default
31:24	**GPIO Enable 0-7 (GPIOENn)** A '1' sets the associated pin to use the default function. When cleared low, the pin functions as a GPIO signal. GPIO0 - GPIO7 can be used to mirror internal MII signals when not enabled. See the IME bit in Section 6.3.5, "Hardware Configuration Register (HW_CFG)" GPIOEN0 - bit 24 GPIOEN1 - bit 25 GPIOEN2 - bit 26 GPIOEN3 - bit 27 GPIOEN4 - bit 28 GPIOEN5 - bit 29 GPIOEN6 - bit 30 GPIOEN7 - bit 31 **Note:** These GPIOs are disabled after a reset.	R/W	FFh
23:16	**GPIO Buffer Type 0-7 (GPIOBUFn)** When set, the output buffer for the corresponding GPIO signal is configured as a push/pull driver. When cleared, the corresponding GPIO signal is configured as an open-drain driver. GPIOBUF0 - bit 16 GPIOBUF1 - bit 17 GPIOBUF2 - bit 18 GPIOBUF3 - bit 19 GPIOBUF4 - bit 20 GPIOBUF5 - bit 21 GPIOBUF6 - bit 22 GPIOBUF7 - bit 23	R/W	00h
15:8	**GPIO Direction 0-7 (GPIODIRn)** When set, enables the corresponding GPIO as output. When cleared, the GPIO is enabled as an input. GPIODIR0 - bit 8 GPIODIR1 - bit 9 GPIODIR2 - bit 10 GPIODIR3 - bit 11 GPIODIR4 - bit 12 GPIODIR5 - bit 13 GPIODIR6 - bit 14 GPIODIR7 - bit 15	R/W	00h

Bits	Description	Type	Default
7:0	**GPIO Data 0-7 (GPIODn)** When enabled as an output, the value written is reflected on GPIOn. When read, GPIODn reflects the current state of the corresponding GPIO pin. GPIOD0 - bit 0 GPIOD1 - bit 1 GPIOD2 - bit 2 GPIOD3 - bit 3 GPIOD4 - bit 4 GPIOD5 - bit 5 GPIOD6 - bit 6 GPIOD7 - bit 7	R/W	Note 6-8

Note 6-8 The default value depends on the state of the GPIO pin.

6.3.11 AUTOMATIC FLOW CONTROL CONFIGURATION REGISTER (AFC_CFG)

Address: 02Ch Size: 32 bits

This register configures the mechanism that controls both the automatic- and software-initiated transmission of pause frames and back pressure. Refer to Section 4.5.1, "Flow Control" for more information on flow control operation.

Note: The device will not transmit pause frames or assert back pressure if the transmitter is disabled.

Bits	Description	Type	Default
31:24	**RESERVED**	RO	-
23:16	**Automatic Flow Control High Level (AFC_HI)** Specifies, in multiples of 64 bytes, the level at which flow control will trigger. When this limit is reached, the chip will apply back pressure or will transmit a pause frame, as programmed in bits [3:0] of this register. During full-duplex operation, only a single pause frame is transmitted when this level is reached. The pause time transmitted in this frame is programmed in the Pause Time (FCPT) field of the Flow Control Register (FLOW), contained in the MAC CSR space. During half-duplex operation, each incoming frame that matches the criteria in bits [3:0] of this register will be jammed for the period set in the BACK_DUR field.	R/W	00h
15:8	**Automatic Flow Control Low Level (AFC_LO)** Specifies, in multiples of 64 bytes, the level at which a pause frame is transmitted with a pause time setting of zero. When the amount of data in the RX Data FIFO falls below this level, the pause frame is transmitted. A pause time value of zero instructs the other transmitting device to immediately resume transmission. The zero time pause frame will only be transmitted if the RX Data FIFO had reached the AFC_HI level and a pause frame was sent. A zero pause time frame is sent whenever automatic flow control in enabled in bits [3:0] of this register.	R/W	00h

Bits	Description	Type	Default
7:4	**Back Pressure Duration (BACK_DUR)** This field is used to select the time period for the Back Pressure Duration Timer. This field has no function in full-duplex mode. **Note:** Back Pressure Duration is slightly greater in 10 Mbs mode. Back Pressure Duration 100 Mbps Mode: 0h = 5 µs 1h = 10 µs 2h = 15 µs 3h = 25 µs 4h = 50 µs 5h = 100 µs 6h = 150 µs 7h = 200 µs 8h = 250 µs 9h = 300 µs Ah = 350 µs Bh = 400 µs Ch = 450 µs Dh = 500 µs Eh = 550 µs Fh = 600 µs 10 Mbps Mode: 0h = 7.2 µs 1h = 12.2 µs 2h = 17.2 µs 3h = 27.2 µs 4h = 52.2 µs 5h = 102.2 µs 6h = 152.2 µs 7h = 202.2 µs 8h = 252.2 µs 9h = 302.2 µs Ah = 352.2 µs Bh = 402.2 µs Ch = 452.2 µs Dh = 502.2 µs Eh = 552.2 µs Fh = 602.2 µs	R/W	0h
3	**Flow Control on Multicast Frame (FCMULT)** When this bit is set, the device will assert back pressure when the AFC level is reached and a multicast frame is received. This field has no function in full-duplex mode.	R/W	0b
2	**Flow Control on Broadcast Frame (FCBRD)** When this bit is set, the device will assert back pressure when the AFC level is reached and a broadcast frame is received. This field has no function in full-duplex mode.	R/W	0b
1	**Flow Control on Address Decode (FCADD)** When this bit is set, the device will assert back pressure when the AFC level is reached and a frame addressed to the device is received. This field has no function in full-duplex mode.	R/W	0b

Bits	Description	Type	Default
0	**Flow Control on Any Frame (FCANY)** When this bit is set, the device will assert back pressure or transmit a pause frame when the AFC level is reached and any frame is received. Setting this bit enables full-duplex flow control when the device is operating in full-duplex mode. When this mode is enabled during half-duplex operation, the flow controller does not decode the MAC address and will send a pause frame upon receipt of a valid preamble (i.e., immediately at the beginning of the next frame after the RX Data FIFO level is reached). When this mode is enabled during full-duplex operation, the flow controller will immediately instruct the MAC to send a pause frame when the RX Data FIFO level is reached. The MAC will queue the pause frame transmission for the next available window. Setting this bit overrides bits [3:1] of this register.	R/W	0b

6.3.12 EEPROM COMMAND REGISTER (E2P_CMD)

Address: 030h Size: 32 bits

This register is used to control the read and write operations on the Serial EEPROM.

Bits	Description	Type	Default
31	**EPC Busy** When a '1' is written into this bit, the operation specified in the EPC Command field is performed at the specified EEPROM address. This bit will remain set until the operation is complete. In the case of a read, the host can read valid data from the E2P Data register. The E2P_CMD and E2P_DATA registers should not be modified until this bit is cleared. In the case where a write is attempted and an EEPROM is not present, the EPC Busy remains busy until the EPC time-out occurs. At that time, the busy bit is cleared. **Note:** EPC busy will be high immediately following power-up, chip-level, or USB reset. After the EEPROM controller has finished reading (or attempts to read) the USB Descriptors and Ethernet default register values, the EPC Busy bit is cleared.	SC	0b

Bits	Description	Type	Default
30:28	**EPC Command** This field is used to issue commands to the EEPROM controller. The EPC will execute commands when the EPC Busy bit is set. A new command must not be issued until the previous command completes. This field is encoded as follows: 000 = READ 001 = EWDS 010 = EWEN 011 = WRITE 100 = WRAL 101 = ERASE 110 = ERAL 111 = RELOAD **READ (Read Location):** This command will cause a read of the EEPROM location pointed to by EPC Address. The result of the read is available in the E2P_DATA register. **EWDS (Erase/Write Disable):** After issued, the EEPROM will ignore erase and write commands. To re-enable erase/write operations, issue the EWEN command. **EWEN (Erase/Write Enable):** Enables the EEPROM for erase and write operations. The EEPROM will allow erase and write operations until the Erase/Write Disable command is sent, or until power is cycled. **Note:** The EEPROM device will power-up in the erase/write-disabled state. Any erase or write operations will fail until an Erase/Write Enable command is issued. **WRITE (Write Location):** If erase/write operations are enabled in the EEPROM, this command will cause the contents of the E2P_DATA register to be written to the EEPROM location selected by the EPC Address field. **WRAL (Write All):** If erase/write operations are enabled in the EEPROM, this command will cause the contents of the E2P_DATA register to be written to every EEPROM memory location. **ERASE (Erase Location):** If erase/write operations are enabled in the EEPROM, this command will erase the location selected by the EPC Address field. **ERAL (Erase All):** If erase/write operations are enabled in the EEPROM, this command will initiate a bulk erase of the entire EEPROM. **RELOAD (Data Reload):** Instructs the EEPROM controller to reload the data from the EEPROM. If a value of A5h is not found in the first address of the EEPROM, the EEPROM is assumed to be un-programmed and the Reload operation will fail. The “Data Loaded” bit indicates a successful load of the data. **Note:** A failed reload operation will result in no change to descriptor information or register contents. These items will not be set to default values as a result of the reload failure.	R/W	000b
27:11	**RESERVED**	RO	-

Bits	Description	Type	Default
10	**EPC Time-out** If an EEPROM operation is performed, and there is no response from the EEPROM within 30 ms, the EEPROM controller will time-out and return to its idle state. This bit is set when a time-out occurs, indicating that the last operation was unsuccessful. **Note:** If the EEDI pin is pulled-high (default if left unconnected), EPC commands will not time out if the EEPROM device is missing. In this case, the EPC Busy bit will be cleared as soon as the command sequence is complete. It should also be noted that the ERASE, ERAL, WRITE and WRAL commands are the only EPC commands that will time-out if an EEPROM device is not present and the EEDI signal is pulled low.	R/WC	0
9	**Data Loaded** When set, this bit indicates that a valid EEPROM was found and that the USB and Ethernet data programming has completed normally. This bit is set after a successful load of the data after power-up or after a RELOAD command has been completed.	R/WC	0
8:0	**EPC Address** The 9-bit value in this field is used by the EEPROM Controller to address a specific memory location in the Serial EEPROM. This is a byte-aligned address.	R/W	00h

6.3.13 EEPROM DATA REGISTER (E2P_DATA)

Address: 034h Size: 32 bits

This register is used in conjunction with the E2P_CMD register to perform read and write operations to the Serial EEPROM.

Bits	Description	Type	Default
31:8	**RESERVED**	RO	-
7:0	**EEPROM Data** Value read from or written to the EEPROM.	R/W	00h

6.3.14 BURST CAP REGISTER (BURST_CAP)

Address: 038h Size: 32 bits

This register is used to limit the size of the data burst transmitted by the UTX. When more than the amount specified in the BURST_CAP register is transmitted, the UTX will send a ZLP.

Note: This register must be enabled through the Section 6.3.5, "Hardware Configuration Register (HW_CFG)".

Bits	Description	Type	Default
31:8	**RESERVED**	RO	-
7:0	**BURST_CAP** The maximum amount of contiguous data that may be transmitted by the UTX before a ZLP is sent. This field has units of 512 bytes. **Note:** A value less than or equal to 4 indicates that burst cap enforcement is disabled. In this case, the UTX always responds to In Tokens with a ZLP when the Bulk-In Empty Response (BIR) bit in the Hardware Configuration Register (HW_CFG) is deasserted. It will respond with NACKs if the Bulk-In Empty Response (BIR) bit is set.	R/W	00h

6.3.15 DATA PORT SELECT REGISTER (DP_SEL)

Address: 040h Size: 32 bits

Before accessing the internal RAMs, the TESTEN bit must be set. It is not valid to use the RAM data port during run time.

The RAM Test Mode Select chooses which internal RAM to access.

The Data Port Ready bit indicates when the data port RAM access has been completed. In the case of a read operation, this indicates when the read data has been stored in the DP_DATA register.

Bits	Description	Type	Default
31	**Data Port Ready (DPRDY)** 0 = Data port is processing a transaction. 1 = Data port is ready.	RO	1b
30:3	**RESERVED**	RO	-
2:1	**RAM Test Select (RSEL)** Selects which RAM to access. 00 = FCT Data RAM 01 = EEPROM storage RAM 10 = TX TLI RAM 11 = RX TLI RAM	R/W	0b
0	**RAM Test Mode Enable (TESTEN)** Put all test accessible RAMs in test mode.	R/W	0b

6.3.16 DATA PORT COMMAND REGISTER (DP_CMD)

Address: 044h Size: 32 bits

This register commences the data port access. Writing a one to this register will enable a write access, while writing a zero will do a read access.

The address and data registers need to be configured appropriately for the desired read or write operation before accessing this register.

Bits	Description	Type	Default
31:1	**RESERVED**	RO	-
0	**Data Port Write** Selects operation. Writing to this bit initiates the data port access. 0 = Read operation 1 = Write operation	R/W	0b

6.3.17 DATA PORT ADDRESS REGISTER (DP_ADDR)

Address: 048h Size: 32 bits

Indicates the address to be used for the data port access.

Bits	Description	Type	Default
31:15	**RESERVED**	RO	-
14:0	**Data Port Address[14:0]** **Note:** This quantity specifies a DWORD address.	R/W	0000h

6.3.18 DATA PORT DATA 0 REGISTER (DP_DATA0)

Address: 04Ch Size: 32 bits

The Data Port Data register holds the write data for a write access and the resultant read data for a read access.

Before reading this register for the result of a read operation, the Data Port Ready bit should be checked. The Data Port Ready bit must indicate the data port is ready. Otherwise the read operation is still in progress.

Bits	Description	Type	Default
31:0	**Data Port Data [31:0]**	R/W	0000_0000h

6.3.19 DATA PORT DATA 1 REGISTER (DP_DATA1)

Address: 050h Size: 32 bits

The Data Port Data register holds the write data for a write access and the resultant read data for a read access.

Before reading the this register for the result of a read operation, the Data Port Ready bit should be checked. The Data Port Ready bit must indicate the data port is ready. Otherwise the read operation is still in progress.

This register required when accessing the RX TLI and TX TLI RAMs. These RAMs have a width of 37 bits.

Bits	Description	Type	Default
31:5	**RESERVED**	RO	-
4:0	**Data Port Data [36:32]**	R/W	00h

6.3.20 GENERAL PURPOSE IO WAKE ENABLE AND POLARITY REGISTER (GPIO_WAKE)

Address: 064h Size: 32 bits

This register enables the GPIOs to function as wake events for the device when asserted. It also allows the polarity used for a wake event/interrupt to be configured.

Note: GPIOs must not cause a wake event to the device when not configured as a GPIO.

Bits	Description	Type	Default
31	**PHY Link Up Enable (PHY_LINKUP_EN)** Setting this bit enables the use of GPIO7 to signal a PHY Link Up event when in SUSPEND0 or SUSPEND3 state. In addition to setting this bit, the parameters for GPIO7 must be set as discussed in Section 4.12.2.4, "Enabling External PHY Link Up Wake Events" in order for signaling to occur.	R/W	0b
30:27	**RESERVED**	RO	-
26:16	**GPIO Polarity 0-10 (GPIOPOLn)** 0 = Wakeup/interrupt is triggered when GPIO is driven low. 1 = Wakeup/interrupt is triggered when GPIO is driven high. GPIOPOL0 - bit 16 GPIOPOL1 - bit 17 GPIOPOL2 - bit 18 GPIOPOL3 - bit 19 GPIOPOL4 - bit 20 GPIOPOL5 - bit 21 GPIOPOL6 - bit 22 GPIOPOL7 - bit 23 GPIOPOL8 - bit 24 GPIOPOL9 - bit 25 GPIOPOL10 - bit 26	R/W	000h
15:11	**RESERVED**	RO	-
10:0	**GPIO Wake 0-10 (GPIOWKn)** 0 = The GPIO can not wake up the device. 1 = The GPIO can trigger a wake-up event. GPIOWK0 - bit 0 GPIOWK1 - bit 1 GPIOWK2 - bit 2 GPIOWK3 - bit 3 GPIOWK4 - bit 4 GPIOWK5 - bit 5 GPIOWK6 - bit 6 GPIOWK7 - bit 7 GPIOWK8 - bit 8 GPIOWK9 - bit 9 GPIOWK10 - bit 10 **Note:** This field is protected by Reset Protection (RST_PROTECT).	R/W	Note 6-9

Note 6-9 The default value of this field is loaded from the associated bytes of the EEPROM. The high order unused bits of the EEPROM are ignored. If no EEPROM is present, the default value of each bit in the field is 0. A USB Reset or Lite Reset (LRST) will cause this field to be restored to the image value last loaded from EEPROM, or will cause the value of each bit to be set to 0 if no EEPROM is present.

6.3.21 INTERRUPT ENDPOINT CONTROL REGISTER (INT_EP_CTL)

Address: 068h Size: 32 bits

This register determines which events cause status to be reported by the interrupt Endpoint. See Section 4.3.1.3, "Endpoint 3 (Interrupt)" for more details.

Bits	Description	Type	Default
31	**Interrupt Endpoint Always On (INTEP_ON)** When this bit is set, an interrupt packet will always be sent at the interrupt Endpoint interval. 0 = Only allow the transmission of an interrupt packet when an interrupt source is enabled and occurs. 1 = Always transmit an interrupt packet at the interrupt interval.	R/W	0b
30:20	**RESERVED**	RO	-
19	**MAC Reset Time Out (MACRTO_EN)** 0 = This event can not cause an interrupt packet to be issued. 1 = This event can cause an interrupt packet to be issued.	R/W	0b
18	**RX FIFO Has Frame Enable (RX_FIFO_EN)** 0 = This event can not cause an interrupt packet to be issued. 1 = This event can cause an interrupt packet to be issued.	R/W	0b
17	**TX Stopped Enable (TXSTOP_EN)** 0 = This event can not cause an interrupt packet to be issued. 1 = This event can cause an interrupt packet to be issued.	R/W	0b
16	**RX Stopped Enable (RXSTOP_EN)** 0 = This event can not cause an interrupt packet to be issued. 1 = This event can cause an interrupt packet to be issued.	R/W	0b
15	**PHY Interrupt Enable (PHY_EN)** 0 = This event can not cause an interrupt packet to be issued. 1 = This event can cause an interrupt packet to be issued.	R/W	0b
14	**Transmitter Error Enable (TXE_EN)** 0 = This event can not cause an interrupt packet to be issued. 1 = This event can cause an interrupt packet to be issued.	R/W	0b
13	**TX Data FIFO Underrun Interrupt Enable (TDFU_EN)** 0 = This event can not cause an interrupt packet to be issued. 1 = This event can cause an interrupt packet to be issued.	R/W	0b
12	**TX Data FIFO Overrun Interrupt Enable (TDFO_EN)** 0 = This event can not cause an interrupt packet to be issued. 1 = This event can cause an interrupt packet to be issued.	R/W	0b

Bits	Description	Type	Default
11	**RX Dropped Frame Interrupt Enable (RXDF_EN)** 0 = This event can not cause an interrupt packet to be issued. 1 = This event can cause an interrupt packet to be issued.	R/W	0b
10:0	**GPIOx Interrupt Enable (GPIOx_EN)** 0 = This event can not cause an interrupt packet to be issued. 1 = This event can cause an interrupt packet to be issued.	R/W	0b

6.3.22 BULK-IN DELAY REGISTER (BULK_IN_DLY)

Address: 06Ch Size: 32 bits

Bits	Description	Type	Default
31:16	**RESERVED**	RO	-
15:0	**Bulk-In Delay** Before sending a short packet, the UTX waits the delay specified by this register. This register has units of 16.667 ns and a default interval of 34.133 µs.	R/W	800h

6.3.23 RECEIVE FIFO LEVEL DEBUG REGISTER (DBG_RX_FIFO_LVL)

Address: 070h Size: 32 bits

Bits	Description	Type	Default
31:30	**RESERVED**	RO	-
29:16	**RX FIFO Read Level (RXRDLVL)** This is a DWORD count defined as follows: The count is increased by the number of DWORDs contained in the packet after the ENTIRE packet has been written into the FIFO. As a packet is read from the FIFO, it is decremented each time a DWORD is read. On rewind, it will increase by the number of DWORDS read out of the FIFO. **Note:** Rewind case example: On a USB error, whatever was read will be rewound and the packet will be retransmitted to the host.	RO	0000h
15:14	**RESERVED**	RO	-
13:0	**RX FIFO Write Level (RXWRLVL)** This is a DWORD count defined as follows: As a packet is written into the FIFO, it is incremented each time a DWORD is written. Whenever a COMPLETE packet has been read from the FIFO, it is decreased by the number of DWORDs contained in the packet. On rewind, it is decreased by the number of DWORDs of the packet that has currently been transferred into the FIFO. **Note:** Rewind case example: On an FCS error, whatever was written in the FIFO will be rewound out.	RO	0000h

6.3.24 RECEIVE FIFO POINTER DEBUG REGISTER (DBG_RX_FIFO_PTR)

Address: 074h Size: 32 bits

This register provides information about the RX FIFO read/write pointers.

Bits	Description	Type	Default
31:29	**RESERVED**	RO	-
28:16	**RX FIFO Read Pointer (RXRDPTR)** Current value of RX FIFO read pointer (DWORD address).	RO	0000h
15:13	**RESERVED**	RO	-
12:0	**RX FIFO Write Pointer (RXWRPTR)** Current value of RX FIFO write pointer (DWORD address).	RO	0000h

6.3.25 TRANSMIT FIFO LEVEL DEBUG REGISTER (DBG_TX_FIFO_LVL)

Address: 078h Size: 32 bits

Bits	Description	Type	Default
31:28	**RESERVED**	RO	-
27:16	**TX FIFO Read Level (TXRDLVL)** This is a DWORD count defined as follows: The count is increased by the number of DWORDs contained in the packet after the ENTIRE packet has been written into the FIFO. As a packet is read from the FIFO, it is decremented each time a DWORD is read. **Note:** Rewinds are not supported.	RO	000h
15:12	**RESERVED**	RO	-
11:0	**TX FIFO Write Level (TXWRLVL)** This is a DWORD count defined as follows: As a packet is written into the FIFO, it is incremented each time a DWORD is written. Whenever a COMPLETE packet has been read from the FIFO, it is decreased by the number of DWORDs contained in the packet. On rewind, it is decreased by the number of DWORDs of the packet that has currently been transferred into the FIFO. **Note:** Write side rewinds are supported, i.e., if a USB packet is received with an error, the packet is rewound out and re-received from the host.	RO	000h

6.3.26 TRANSMIT FIFO POINTER DEBUG REGISTER (DBG_TX_FIFO_PTR)

Address: 07Ch Size: 32 bits

This register provides information about the TX FIFO read/write pointers.

Bits	Description	Type	Default
31:27	**RESERVED**	RO	-
26:16	**TX FIFO Read Pointer (TXRDPTR)** Current value of TX FIFO read pointer (DWORD address).	RO	000h
15:11	**RESERVED**	RO	-
10:0	**TX FIFO Write Pointer (TXWRPTR)** Current value of TX FIFO write pointer (DWORD address).	RO	000h

6.3.27 HS DESCRIPTOR ATTRIBUTES REGISTER (HS_ATTR)

Address: 0A0h Size: 32 bits

This register sets the length values for HS descriptors that have been loaded into Descriptor RAM via the Data Port registers. The HS Polling interval is also defined by a field within this register. The Descriptor RAM images may be used, in conjunction with this register, to facilitate customized operation when no EEPROM is present.

Note: If a descriptor does not exist in Descriptor RAM, its size value must be written as 00h.

Note: This register only affects system operation when an EEPROM is not present and the EEPROM Emulation Enable (EEM) bit indicates Descriptor RAM and the Attributes registers are to be used for descriptor processing.

Note: Writing to this register when an EEPROM is present is prohibited and will result in untoward operation and unexpected results.

Note: This register is protected by Reset Protection (RST_PROTECT).

Bits	Description	Type	Default
31:24	**RESERVED**	RO	-
23:16	**HS Polling Interval (HS_POLL_INT)**	R/W	04h
15:8	**HS Device Descriptor Size (HS_DEV_DESC_SIZE)** Note 6-10	R/W	00h
7:0	**HS Configuration Descriptor Size (HS_CFG_DESC_SIZE)** Note 6-10	R/W	00h

Note 6-10 The only legal values are 0 and 0x12h. Writing any other values will result in untoward behavior and unexpected results.

6.3.28 FS DESCRIPTOR ATTRIBUTES REGISTER (FS_ATTR)

Address: 0A4h Size: 32 bits

This register sets the length values for FS descriptors that have been loaded into Descriptor RAM via the Data Port registers. The FS Polling interval is also defined by a field within this register. The Descriptor RAM images may be used, in conjunction with this register, to facilitate customized operation when no EEPROM is present.

Note: If a descriptor does not exist in Descriptor RAM, its size value must be written as 00h.

Note: This register only affects system operation when an EEPROM is not present and the EEPROM Emulation Enable (EEM) bit indicates Descriptor RAM and the Attributes Registers are to be used for descriptor processing.

Note: Writing to this register when an EEPROM is present is prohibited and will result in untoward operation and unexpected results.

Note: This register is protected by Reset Protection (RST_PROTECT).

Bits	Description	Type	Default
31:24	**RESERVED**	RO	-
23:16	**FS Polling Interval (FS_POLL_INT)**	R/W	01h
15:8	**FS Device Descriptor Size (FS_DEV_DESC_SIZE)** Note 6-11	R/W	00h
7:0	**FS Configuration Descriptor Size (FS_CFG_DESC_SIZE)** Note 6-11	R/W	00h

Note 6-11 The only legal values are 0 and 0x12h. Writing any other values will result in untoward behavior and unexpected results.

6.3.29 STRING DESCRIPTOR ATTRIBUTES REGISTER 0 (STRNG_ATTR0)

Address: 0A8h Size: 32 bits

This register sets the length values for the named String Descriptors that have been loaded into Descriptor RAM via the Data Port registers. The Descriptor RAM images may be used, in conjunction with this register, to facilitate customized operation when no EEPROM is present.

Note: If a descriptor does not exist in Descriptor RAM, its size value must be written as 00h.

Note: This register only affects system operation when an EEPROM is not present and the EEPROM Emulation Enable (EEM) bit indicates Descriptor RAM and the Attributes registers are to be used for descriptor processing.

Note: Writing to this register when an EEPROM is present is prohibited and will result in untoward operation and unexpected results.

Note: This register is protected by Reset Protection (RST_PROTECT).

Bits	Description	Type	Default
31:24	**Configuration String Descriptor Size (CFGSTR_DESC_SIZE)**	R/W	00h
23:16	**Serial Number String Descriptor Size (SERSTR_DESC_SIZE)**	R/W	00h
15:8	**Product Name String Descriptor Size (PRODSTR_DESC_SIZE)**	R/W	00h
7:0	**Manufacturing String Descriptor Size (MANUF_DESC_SIZE)**	R/W	00h

6.3.30 STRING DESCRIPTOR ATTRIBUTES REGISTER 1 (STRNG_ATTR1)

Address: 0ACh Size: 32 bits

This register sets the length values for the named String Descriptors that have been loaded into Descriptor RAM via the Data Port registers. The Descriptor RAM images may be used, in conjunction with this register, to facilitate customized operation when no EEPROM is present.

Note: If a descriptor does not exist in Descriptor RAM, its size value must be written as 00h.

Note: This register only affects system operation when an EEPROM is not present and the EEPROM Emulation Enable (EEM) bit indicates Descriptor RAM and the Attributes Registers are to be used for descriptor processing.

Note: Writing to this register when an EEPROM is present is prohibited and will result in untoward operation and unexpected results.

Note: This register is protected by Reset Protection (RST_PROTECT).

Bits	Description	Type	Default
31:8	**RESERVED**	RO	-
7:0	**Interface String Descriptor Size (INTSTR_DESC_SIZE)**	R/W	00h

6.3.31 FLAG ATTRIBUTES REGISTER (FLAG_ATTR)

Address: 0B0h Size: 32 bits

This register sets the values of elements of the Configuration Flags and PME flags when no EEPROM is present and customized operation, using Descriptor RAM images, is to occur. This register does not contain Configuration Flag elements that are components of other registers. Those elements will be programmed by the driver software directly prior to initiating customized operation via Descriptor RAM.

Note: This register only affects system operation when an EEPROM is not present and the EEPROM Emulation Enable (EEM) bit indicates Descriptor RAM and the Attributes registers are to be used for descriptor processing.

Note: Writing to this register when an EEPROM is present is prohibited and will result in untoward operation and unexpected results.

Note: This register is protected by Reset Protection (RST_PROTECT).

Bits	Description	Type	Default
31:18	**RESERVED**	RO	-
17	**Remote Wakeup Support (RMT_WKP)** Refer to Remote Wakeup Support bit in Table 4-57, “Configuration Flags” for definition.	R/W	Note 6-12
16	**Power Method (PWR_SEL)** Refer to Power Method bit in Table 4-57, “Configuration Flags” for definition.	R/W	1b
15:8	**RESERVED**	RO	-
7:0	**GPIO PME Flags (PME_FLAGS)** Refer to Table 4-58, “GPIO PME Flags” for bit definitions.	R/W	00h

Note 6-12 The default value depends on the setting of the RMT_WKP strap.

6.4 MAC Control and Status Registers

Table 6-4 lists the registers contained in this section.

TABLE 6-4: MAC CONTROL AND STATUS REGISTER (MCSR) MAP

Address	Register Name
100h	MAC Control Register (MAC_CR)
104h	MAC Address High Register (ADDRH)
108h	MAC Address Low Register (ADDRL)
10Ch	Multicast Hash Table High Register (HASHH)
110h	Multicast Hash Table Low Register (HASHL)
114h	MII Access Register (MII_ACCESS)
118h	MII Data Register (MII_DATA)
11Ch	Flow Control Register (FLOW)
120h	VLAN1 Tag Register (VLAN1)
124h	VLAN2 Tag Register (VLAN2)
128h	Wakeup Frame Filter (WUFF)
12Ch	Wakeup Control and Status Register (WUCSR)
130h	Checksum Offload Engine Control Register (COE_CR)
134h - 1FCh	Reserved for future use

6.4.1 MAC CONTROL REGISTER (MAC_CR)

Address: 100h Size: 32 bits

This register establishes the RX and TX operating modes and includes controls for address filtering and packet filtering.

Bits	Description	Type	Default
31	**Receive All Mode (RXALL)** When set, all incoming packets will be received and passed on to the address filtering function for processing of the selected filtering mode on the received frame. Address filtering then occurs and is reported in Receive Status. When reset, only frames that pass Destination Address filtering will be sent to the application.	R/W	0b
30:24	**RESERVED**	RO	-
23	**Disable Receive Own (RCVOWN)** When set, the MAC disables the reception of frames when TXEN is asserted. The MAC blocks the transmitted frame on the receive path. When reset, the MAC receives all packets the PHY gives, including those transmitted by the MAC.This bit should be reset when the full-duplex mode bit is set.	R/W	0b
22	**RESERVED**	RO	-
21	**Loopback Operation Mode (LOOPBK)** Selects the loop back operation modes for the MAC. This is only for full-duplex mode. 0 = Normal. No feedback 1 = Internal through MII In internal loopback mode, the TX frame is received by the Internal MII interface and sent back to the MAC without being sent to the PHY. **Note:** When enabling or disabling the loopback mode, it can take up to 10 µs for the mode change to occur. The transmitter and receiver must be stopped and disabled when modifying the LOOPBK bit. The transmitter or receiver should not be enabled within 10 µs of modifying the LOOPBK bit.	R/W	0b
20	**Full Duplex Mode (FDPX)** When set, the MAC operates in full-duplex mode, in which it can transmit and receive simultaneously.	R/W	0b
19	**Pass All Multicast (MCPAS)** When set, indicates that all incoming frames with a Multicast destination address (first bit in the destination address field is ‘1’) are received. Incoming frames with physical address (Individual Address/Unicast) destinations are filtered and received only if the address matches the MAC Address.	R/W	0b
18	**Promiscuous Mode (PRMS)** When set, indicates that any incoming frame is received regardless of its destination address.	R/W	1b
17	**Inverse Filtering (INVFILT)** When set, the address check function operates in Inverse Filtering mode. This is valid only during Perfect Filtering mode.	R/W	0b
16	**Pass Bad Frames (PASSBAD)** When set, all incoming frames that passed address filtering are received, including runt frames, collided frames or truncated frames caused by buffer underrun.	R/W	0b

Bits	Description	Type	Default
15	**Hash Only Filtering mode (HO)** When set, the address check function operates in the imperfect address filtering mode both for physical and multicast addresses.	R/W	0b
14	**RESERVED**	RO	-
13	**Hash/Perfect Filtering Mode (HPFILT)** When reset (‘0’), the device will implement a perfect address filter on incoming frames, according the address specified in the MAC address register. When set (‘1’), the address check function does imperfect address filtering of multicast incoming frames according to the hash table specified in the multicast hash table register. If the Hash Only Filtering mode (HO) bit is set (‘1’), then the physical addresses (IA) are imperfect filtered too. If the Hash Only Filtering mode (HO) bit is reset (‘0’), then the IA addresses are perfect address filtered according to the MAC Address register.	R/W	0b
12	**Late Collision Control (LCOLL)** When set, enables retransmission of the collided frame even after the collision period (late collision). When reset, the MAC disables frame transmission on a late collision. In any case, the Late Collision status is appropriately updated in the Transmit Packet status.	R/W	0b
11	**Disable Broadcast Frames (BCAST)** When set, disables the reception of broadcast frames. When reset, forwards all broadcast frames to the application. **Note:** When Wakeup Frame detection is enabled via the Wakeup Frame Enable (WUEN) bit of the Wakeup Control and Status Register (WUCSR), a broadcast Wakeup Frame will wake up the device despite the state of this bit.	R/W	0b
10	**Disable Retry (DISRTY)** When set, the MAC attempts only one transmission. When a collision is seen on the bus, the MAC ignores the current frame and goes to the next frame and a retry error is reported in the Transmit status. When reset, the MAC attempts 16 transmissions before signaling a retry error.	R/W	0b
9	**RESERVED**	RO	-
8	**Automatic Pad Stripping (PADSTR)** When set, the MAC strips the pad field on all incoming frames if the length field is less than 46 bytes. The FCS field is also stripped, since it is computed at the transmitting station based on the data and pad field characters, and is invalid for a received frame that has had the pad characters stripped. Receive frames with a 46-byte or greater length field are passed to the application unmodified (FCS is not stripped). When reset, the MAC passes all incoming frames to system memory unmodified.	R/W	0b

<table>
<tr><th>Bits</th><th>Description</th><th>Type</th><th>Default</th></tr>
<tr><td>7:6</td><td>BackOff Limit (BOLMT)
The BOLMT bits allow the user to set its back-off limit in a relaxed or aggressive mode. According to IEEE 802.3, the MAC has to wait for a random number [r] of slot-times (Note 6-13) after it detects a collision, where:
(eq.1)0 < r < 2^K
The exponent K is dependent on how many times the current frame to be transmitted has been retried, as follows:
(eq.2)K = min (n, 10) where n is the current number of retries.
If a frame has been retried three times, then K = 3 and r = 8 slot-times maximum. If it has been retried 12 times, then K = 10 and r = 1024 slot-times maximum.
An LFSR (linear feedback shift register) 20-bit counter emulates a 20-bit random number generator, from which r is obtained. Once a collision is detected, the number of the current retry of the current frame is used to obtain K (eq.2). This value of K translates into the number of bits to use from the LFSR counter. If the value of K is 3, the MAC takes the value in the first three bits of the LFSR counter and uses it to count down to zero on every slot-time. This effectively causes the MAC to wait eight slot-times. To give the user more flexibility, the BOLMT value forces the number of bits to be used from the LFSR counter to a predetermined value as in the table below.

<table>
<tr><th>BOLMT Value</th><th># Bits Used from LFSR Counter</th></tr>
<tr><td>00</td><td>10</td></tr>
<tr><td>01</td><td>8</td></tr>
<tr><td>10</td><td>4</td></tr>
<tr><td>11</td><td>1</td></tr>
</table>
Thus, if the value of K = 10, the MAC will look at the BOLMT if it is 00, then use the lower ten bits of the LFSR counter for the wait countdown. If the BOLMT is 10, then it will only use the value in the first four bits for the wait countdown, etc.

Note 6-13 Slot-time = 512 bit times. (See IEEE 802.3 Spec., Sections 4.2.3.25 and 4.4.2.1)</td><td>R/W</td><td>00b</td></tr>
<tr><td>5</td><td>Deferral Check (DFCHK)
When set, enables the deferral check in the MAC. The MAC will abort the transmission attempt if it has deferred for more than 24,288 bit times. Deferral starts when the transmitter is ready to transmit, but is prevented from doing so because the CRS is active. Defer time is not cumulative. If the transmitter defers for 10,000 bit times, then transmits, collides, backs off, and then has to defer again after completion of back-off, the deferral timer resets to 0 and restarts. When reset, the deferral check is disabled in the MAC and the MAC defers indefinitely.</td><td>R/W</td><td>0b</td></tr>
<tr><td>4</td><td>RESERVED</td><td>RO</td><td>-</td></tr>
<tr><td>3</td><td>Transmitter Enable (TXEN)
When set, the MAC’s transmitter is enabled and it will transmit frames from the buffer onto the cable. When reset, the MAC’s transmitter is disabled and will not transmit any frames.</td><td>R/W</td><td>0b</td></tr>
<tr><td>2</td><td>Receiver Enable (RXEN)
When set (‘1’), the MAC’s receiver is enabled and will receive frames from the internal PHY. When reset, the MAC’s receiver is disabled and will not receive any frames from the internal PHY.</td><td>R/W</td><td>0b</td></tr>
</table>

Bits	Description	Type	Default
1:0	**RESERVED**	RO	-

6.4.2 MAC ADDRESS HIGH REGISTER (ADDRH)

Address: 104h Size: 32 bits

This register contains the upper 16 bits of the physical address of the MAC, where ADDRH[15:8] is the 6th octet of the RX frame.

Note: This register is protected by Reset Protection (RST_PROTECT).

Bits	Description	Type	Default
31:16	**RESERVED**	RO	-
15:0	**Physical Address [47:32]** This field contains the upper 16 bits (47:32) of the physical address of the device.	R/W	FFFFh

6.4.3 MAC ADDRESS LOW REGISTER (ADDRL)

Address: 108h Size: 32 bits

This register contains the lower 32 bits of the physical address of the MAC, where ADDRL[7:0] is the first octet of the Ethernet frame.

Note: This register is protected by Reset Protection (RST_PROTECT).

Bits	Description	Type	Default
31:0	**Physical Address [31:0]** This field contains the lower 32 bits (31:0) of the Physical Address of this MAC device.	R/W	FFFF_FFFFh

Table 6-5 illustrates the byte ordering of the ADDRL and ADDRH registers with respect to the reception of the Ethernet physical address.

TABLE 6-5: ADDRL, ADDRH BYTE ORDERING

ADDRL, ADDRH	Order of Reception on Ethernet
ADDRL[7:0]	1st
ADDRL[15:8]	2nd
ADDRL[23:16]	3rd
ADDRL[31:24]	4th
ADDRH[7:0]	5th
ADDRH[15:8]	6th

As an example, if the desired Ethernet physical address is 12-34-56-78-9A-BC, the ADDRL and ADDRH registers would be programmed as shown in Figure 6-1.

FIGURE 6-1: EXAMPLE ADDRL, ADDRH ADDRESS ORDERING

6.4.4 MULTICAST HASH TABLE HIGH REGISTER (HASHH)

Address: 10Ch Size: 32 bits

The 64-bit Multicast table is used for group address filtering. For hash filtering, the contents of the destination address in the incoming frame is used to index the contents of the hash table. The most significant bit determines the register to be used (high/low), while the other five bits determine the bit within the register. A value of 00000 selects bit 0 of the Multicast Hash Table Low register and a value of 11111 selects the bit 31 of the Multicast Hash Table High register.

If the corresponding bit is '1', then the multicast frame is accepted. Otherwise, it is rejected. If the "Pass All Multicast" (MCPAS) bit is set ('1'), then all multicast frames are accepted regardless of the multicast hash values.

The Multicast Hash Table High register contains the higher 32 bits of the hash table and the Multicast Hash Table Low register contains the lower 32 bits of the hash table.

Bits	Description	Type	Default
31:0	**Upper 32 bits of the 64-bit hash table**	R/W	0000_0000h

6.4.5 MULTICAST HASH TABLE LOW REGISTER (HASHL)

Address: 110h Size: 32 bits

This register defines the lower 32-bits of the Multicast Hash Table. Refer to Section 6.4.4, "Multicast Hash Table High Register (HASHH)" for further details.

Bits	Description	Type	Default
31:0	**Lower 32 bits of the 64-bit hash table**	R/W	0000_0000h

6.4.6 MII ACCESS REGISTER (MII_ACCESS)

Address: 114h Size: 32 bits

This register is used to control the management cycles to the internal PHY.

Bits	Description	Type	Default
31:16	**RESERVED**	RO	-
15:11	**PHY Address** For every access to this register, this field must be set to 00001b.	R/W	00000b
10:6	**MII Register Index (MIIRINDA)** These bits select the desired MII register in the PHY.	R/W	00000b
5:2	**RESERVED**	RO	-
1	**MII Write (MIIWnR)** Setting this bit tells the PHY that this will be a write operation using the MII data register. If this bit is not set, this will be a read operation, packing the data in the MII data register.	R/W	0b
0	**MII Busy (MIIBZY)** This bit must be polled to determine when the MII register access is complete. This bit must read a logical '0' before writing to this register or to the MII data register. The LAN driver software must set ('1') this bit in order for the host to read or write any of the MII PHY registers. During an MII register access, this bit will be set, signifying a read or write access is in progress. The MII data register must be kept valid until the MAC clears this bit during a PHY write operation. The MII data register is invalid until the MAC has cleared this bit during a PHY read operation.	SC	0b

6.4.7 MII DATA REGISTER (MII_DATA)

Address: 118h Size: 32 bits

This register contains either the data to be written to the PHY register specified in the MII Access register, or the read data from the PHY register whose index is specified in the MII Access register. Refer to Section 6.4.6, "MII Access Register (MII_ACCESS)" for further details.

Note: The MIIBZY bit in the MII_ACCESS register must be cleared when writing to this register.

Bits	Description	Type	Default
31:16	**RESERVED**	RO	-
15:0	**MII Data** This contains the 16-bit value read from the PHY read operation or the 16-bit data value to be written to the PHY before an MII write operation.	R/W	0000h

6.4.8 FLOW CONTROL REGISTER (FLOW)

Address: 11Ch Size: 32 bits

This register is used to control the generation and reception of the Control frames by the MAC's flow control block. Before writing to this register, the application has to make sure that the busy bit is not set.

Bits	Description	Type	Default
31:16	**Pause Time (FCPT)** This field indicates the value to be used in the PAUSE TIME field in the control frame.	R/W	0000h
15:3	**RESERVED**	RO	-
2	**Pass Control Frames (FCPASS)** When set, the MAC sets the packet filter bit in the receive packet status to indicate to the application that a valid pause frame has been received. The application must accept or discard a received frame based on the packet filter control bit. The MAC receives, decodes and performs the pause function when a valid pause frame is received in full-duplex mode and when flow control is enabled (FCEN bit set). When reset, the MAC resets the packet filter bit in the receive packet status. The MAC always passes the data of all frames it receives (including flow control frames) to the application. Frames that do not pass address filtering, as well as frames with errors, are passed to the application. The application must discard or retain the received frame's data based on the received frame's STATUS field. Filtering modes (promiscuous mode, for example) take precedence over the FCPASS bit.	R/W	0b
1	**Flow Control Enable (FCEN)** When set, enables the MAC flow control function. The MAC decodes all incoming frames for control frames; if it receives a valid control frame (PAUSE command), it disables the transmitter for a specified time (decoded pause time x slot time). When reset, the MAC flow control function is disabled; the MAC does not decode frames for control frames. **Note:** Flow Control is applicable when the MAC is set in full-duplex mode. In half-duplex mode, this bit enables the back pressure function to control the flow of received frames to the MAC.	R/W	0b
0	**Flow Control Busy (FCBSY)** This bit is set high whenever a pause frame or back pressure is being transmitted. This bit should read logical 0 before writing to the Flow Control (FLOW) register. During a transfer of Control Frame, this bit continues to be set, signifying that a frame transmission is in progress. After the PAUSE control frame's transmission is complete, the MAC resets to 0. **Note:** When writing this register the FCBSY bit must always be zero. **Note:** Applications must always write a zero to this bit.	R/W	0b

6.4.9 VLAN1 TAG REGISTER (VLAN1)

Address: 120h Size: 32 bits

This register contains the VLAN tag field to identify VLAN1 frames. For VLAN frames, the legal frame length is increased from 1518 bytes to 1522 bytes.

The RXCOE also uses this register to determine the protocol value to use to indicate the existence of a VLAN tag. When using the RXCOE, this value may only be changed if the RX path is disabled. If it is desired to change this value during run time, it is safe to do so only after the MAC is disabled and the TLI is empty.

Bits	Description	Type	Default
31:16	**RESERVED**	RO	-
15:0	**VLAN1 Tag Identifier (VTI1)** This contains the VLAN Tag field to identify the VLAN1 frames. This field is compared with the 13th and 14th bytes of the incoming frames for VLAN1 frame detection.	R/W	FFFFh

6.4.10 VLAN2 TAG REGISTER (VLAN2)

Address: 124h Size: 32 bits

This register contains the VLAN tag field to identify VLAN2 frames. For VLAN frames the legal frame length is increased from 1518 bytes to 1522 bytes.

Bits	Description	Type	Default
31:16	**RESERVED**	RO	-
15:0	**VLAN2 Tag Identifier (VTI2)** This contains the VLAN Tag field to identify the VLAN2 frames. This field is compared with the 13th and 14th bytes of the incoming frames for VLAN2 frame detection.	R/W	FFFFh

6.4.11 WAKEUP FRAME FILTER (WUFF)

Address: 128h Size: 32 bits

This register is used to configure the Wakeup Frame Filter.

Bits	Description	Type	Default
31:0	**Wakeup Frame Filter (WFF)** The Wakeup Frame Filter is configured through this register using an indexing mechanism. Following a reset, the MAC loads the first value written to this location to the first DWORD in the Wakeup Frame Filter (Filter 0 Byte Mask 0). The second value written to this location is loaded to the second DWORD in the Wakeup Frame Filter (Filter 0 Byte Mask 1) and so on. Once 40 DWORDs (8 filters) have been written, the internal pointer will once again point to the first entry and the filter entries can be modified in the same manner. Similarly, 40 DWORDS (8 filters) can be read sequentially to obtain the values stored in the WFF. Refer to Section 4.5.5, "Wakeup Frame Detection" for further information concerning the Wakeup Frame Filter. **Note:** This register should be read and written using 40 (8 filters) consecutive DWORD operations. Failure to read or write the entire contents of the WFF may cause the internal read/write pointers to be left in a position other than pointing to the first entry. A mechanism for resetting the internal pointers to the beginning of the WFF is available via the WFF Pointer Reset (WFF_PTR_RST) bit of the Wakeup Control and Status Register (WUCSR). This mechanism enables the application program to re-synchronize with the internal WFF pointers if it has not previously read/written the complete contents of the WFF.	R/W	0000_0000h

6.4.12 WAKEUP CONTROL AND STATUS REGISTER (WUCSR)

Address: 12Ch Size: 32 bits

This register contains data pertaining to the MAC's remote wakeup status and capabilities.

Bits	Description	Type	Default
31	**WFF Pointer Reset (WFF_PTR_RST)** This self-clearing bit resets the Wakeup Frame Filter (WFF) internal read and write pointers to the beginning of the WFF.	SC	0b
30:10	**RESERVED**	RO	-
9	**Global Unicast Wakeup Enable (GUEN)** When set, the MAC wakes up from power-saving mode on receipt of a global unicast frame. A global unicast frame has the MAC Address [0] bit set to 0. **Note:** The Wakeup Frame Enable (WUEN) bit of this register must also be set to enable wakeup.	R/W	0b
8	**RESERVED**	RO	-
7	**Perfect DA Frame Received (PFDA_FR)** The MAC sets this bit upon receiving a valid frame with a destination address that matches the physical address. This bit is automatically cleared at the completion of a resume sequence if Resume Clears Remote Wakeup Status (RES_CLR_WKP_STS) is set.	R/WC	0b
6	**Remote Wakeup Frame Received (WUFR)** The MAC sets this bit upon receiving a valid remote Wakeup Frame. This bit is automatically cleared at the completion of a resume sequence if Resume Clears Remote Wakeup Status (RES_CLR_WKP_STS) is set.	R/WC	0b
5	**Magic Packet Received (MPR)** The MAC sets this bit upon receiving a valid Magic Packet. This bit is automatically cleared at the completion of a resume sequence if Resume Clears Remote Wakeup Status (RES_CLR_WKP_STS) is set.	R/WC	0b
4	**Broadcast Frame Received (BCAST_FR)** The MAC sets this bit upon receiving a valid broadcast frame. This bit is automatically cleared at the completion of a resume sequence if Resume Clears Remote Wakeup Status (RES_CLR_WKP_STS) is set.	R/WC	0b
3	**Perfect DA Wakeup Enable (PFDA_EN)** **When set, remote wakeup mode is enabled and the MAC is capable of waking up on receipt of a frame with a destination address that matches the physical address of the device. The physical address is stored in the MAC Address High Register (ADDRH) and MAC Address Low Register (ADDRL).**	R/W	0b
2	**Wakeup Frame Enable (WUEN)** When set, remote wakeup mode is enabled and the MAC is capable of detecting Wakeup Frames as programmed in the Wakeup Frame Filter.	R/W	0b
1	**Magic Packet Enable (MPEN)** When set, Magic Packet wakeup mode is enabled.	R/W	0b
0	**Broadcast Wakeup Enable (BCAST_EN)** When set, remote wakeup mode is enabled and the MAC is capable of waking up from a broadcast frame.	R/W	0b

6.4.13 CHECKSUM OFFLOAD ENGINE CONTROL REGISTER (COE_CR)

Address: 130h Size: 32 bits

This register controls the RX and TX checksum offload engines.

Bits	Description	Type	Default
31:17	**RESERVED**	RO	-
16	**TX Checksum Offload Engine Enable (TX_COE_EN)** TX_COE_EN may only be changed if the TX path is disabled. If it is desired to change this value during run time, it is safe to do so only after the MAC is disabled and the TLI is empty. 0 = The TXCOE is bypassed. 1 = The TXCOE is enabled.	R/W	0b
15:2	**RESERVED**	RO	-
1	**RX Checksum Offload Engine Mode (RX_COE_MODE)** This register indicates whether the COE will check for VLAN tags or a SNAP header prior to beginning its checksum calculation. In its default mode, the calculation will always begin 14 bytes into the frame. RX_COE_MODE may only be changed if the RX path is disabled. If it is desired to change this value during run time, it is safe to do so only after the MAC is disabled and the TLI is empty. 0 = Begin checksum calculation after first 14 bytes of Ethernet Frame. 1 = Begin checksum calculation at start of L3 packet by adjusting for VLAN tags and/or SNAP header.	R/W	0b
0	**RX Checksum Offload Engine Enable (RX_COE_EN)** RX_COE_EN may only be changed if the RX path is disabled. If it is desired to change this value during run time, it is safe to do so only after the MAC is disabled and the TLI is empty. 0 = The RXCOE is bypassed. 1 = The RXCOE is enabled.	R/W	0b

6.5 PHY Registers

The PHY registers are not memory mapped. These registers are accessed indirectly through the MAC via the MII Access Register (MII_ACCESS) and MII Access Register (MII_ACCESS). An index is used to access individual PHY registers. PHY Register Indexes are shown in Table 6-6, "PHY Control and Status Register" below.

Note:	The NASR (**Not Affected by Software Reset)** designation is only applicable when bit 15 of the PHY Basic Control Register (Reset) is set.

TABLE 6-6: PHY CONTROL AND STATUS REGISTER

Index (In Decimal)	Register Name
0	Basic Control Register
1	Basic Status Register
2	PHY Identifier 1 Register
3	PHY Identifier 2 Register
4	Auto Negotiation Advertisement Register
5	Auto Negotiation Link Partner Ability Register
6	Auto Negotiation Expansion Register
16	EDPD NLP/Crossover TimeRegister
17	Mode Control/Status Register
18	Special Modes Register
26	Symbol Error Counter Register
27	Special Control/Status Indications Register
29	Interrupt Source Flag Register
30	Interrupt Mask Register
31	PHY Special Control/Status Register

6.5.1 BASIC CONTROL REGISTER

Index (In Decimal): 0 Size: 16 bits

Bits	Description	Type	Default
15	**PHY Soft Reset** 1 = PHY software reset. Bit is self-clearing. When setting this bit do not set other bits in this register. **Note:** The PHY will be in the normal mode after a PHY software reset.	R/W SC	0b
14	**Loopback** 0 = Normal operation 1 = Loopback mode	R/W	0b
13	**Speed Select** 0 = 10 Mbps 1 = 100 Mbps **Note:** Ignored if Auto Negotiation is enabled (0.12 = 1).	R/W	1b
12	**Auto-Negotiation Enable** 0 = Disable auto-negotiate process 1 = Enable auto-negotiate process (overrides 0.13 and 0.8)	R/W	1b
11	**Power Down** 0 = Normal operation 1 = General power down mode **Note:** The Auto-Negotiation Enable must be cleared before setting the Power Down.	R/W	0b
10	**RESERVED**	RO	-
9	**Restart Auto-Negotiate** 0 = Normal operation 1 = Restart auto-negotiate process **Note:** Bit is self-clearing.	R/W SC	0b
8	**Duplex Mode** 0 = Half duplex 1 = Full duplex **Note:** Ignored if Auto Negotiation is enabled (0.12 = 1).	R/W	0b
7	**Collision Test** 0 = Disable COL test 1 = Enable COL test	R/W	0b
6:0	**RESERVED**	RO	-

6.5.2 BASIC STATUS REGISTER

Index (In Decimal): 1 Size: 16 bits

Bits	Description	Type	Default
15	**100BASE-T4** 0 = No T4 ability 1 = T4 able	RO	0b
14	**100BASE-TX Full Duplex** 0 = No TX full-duplex ability 1 = TX with full-duplex ability	RO	1b
13	**100BASE-TX Half Duplex** 0 = No TX half-duplex ability 1 = TX with half-duplex ability	RO	1b
12	**10BASE-T Full Duplex** 0 = No 10 Mbps with full-duplex ability 1 = 10 Mbps with full-duplex ability	RO	1b
11	**10BASE-T Half Duplex** 0 = No 10 Mbps with half-duplex ability 1 = 10 Mbps with half-duplex ability	RO	1b
10:6	**RESERVED**	RO	-
5	**Auto-Negotiate Complete** 0 = Auto-negotiate process not completed 1 = Auto-negotiate process completed	RO	0b
4	**Remote Fault** 1 = Remote fault condition detected 0 = No remote fault	RO/LH	0b
3	**Auto-Negotiate Ability** 0 = Unable to perform auto-negotiation function 1 = Able to perform auto-negotiation function	RO	1b
2	**Link Status** 0 = Link is down 1 = Link is up	RO/LL	0b
1	**Jabber Detect** 0 = No jabber condition detected 1 = Jabber condition detected	RO/LH	0b
0	**Extended Capabilities** 0 = Does not support extended capabilities registers 1 = Supports extended capabilities registers	RO	1b

6.5.3 PHY IDENTIFIER 1 REGISTER

Index (In Decimal): 2 Size: 16 bits

Bits	Description	Type	Default
15:0	**PHY ID Number** Assigned to the 3rd through 18th bits of the Organizationally Unique Identifier (OUI), respectively.	R/W	0007h

6.5.4 PHY IDENTIFIER 2 REGISTER

Index (In Decimal): 3 Size: 16 bits

Bits	Description	Type	Default
15:10	**PHY ID Number** Assigned to the 19th through 24th bits of the OUI.	R/W	C101h
9:4	**Model Number** Six-bit manufacturer's model number.	R/W	
3:0	**Revision Number** Four-bit manufacturer's revision number.	R/W	

Note: The default value of the Revision Number field may vary dependent on the silicon revision number.

6.5.5 AUTO NEGOTIATION ADVERTISEMENT REGISTER

Index (In Decimal): 4 Size: 16 bits

Bits	Description	Type	Default
15:14	**RESERVED**	RO	-
13	**Remote Fault** 0 = No remote fault 1 = Remote fault detected	R/W	0b
12	**RESERVED**	RO	-
11:10	**Pause Operation** 00 = No PAUSE 01 = Symmetric PAUSE 10 = Asymmetric PAUSE toward link partner 11 = Advertise support for both Symmetric PAUSE and Asymmetric PAUSE toward local device **Note:** When both Symmetric PAUSE and Asymmetric PAUSE are set, the device will only be configured to, at most, one of the two settings upon autonegotiation completion.	R/W	00b
9	**RESERVED**	RO	-
8	**100BASE-TX Full Duplex** 0 = No TX full-duplex ability 1 = TX with full-duplex ability	R/W	1b
7	**100BASE-TX** 0 = No TX ability 1 = TX able	R/W	1b
6	**10BASE-T Full Duplex** 0 = No 10 Mbps with full-duplex ability 1 = 10 Mbps with full-duplex ability	R/W	1b
5	**10BASE-T** 0 = No 10 Mbps ability 1 = 10 Mbps able	R/W	1b
4:0	**Selector Field** 00001 = IEEE 802.3	R/W	00001b

6.5.6 AUTO NEGOTIATION LINK PARTNER ABILITY REGISTER

Index (In Decimal): 5 Size: 16 bits

Bits	Description	Type	Default
15	**Next Page** 0 = No next page ability 1 = Next page capable **Note:** Next page ability is not supported.	RO	0b
14	**Acknowledge** 0 = Link code word not yet received 1 = Link code word received from partner	RO	0b
13	**Remote Fault** 0 = No remote fault 1 = Remote fault detected	RO	0b
12	**RESERVED**	RO	-
11:10	**Pause Operation** 00 = No PAUSE supported by partner station 01 = Symmetric PAUSE supported by partner station 10 = Asymmetric PAUSE supported by partner station 11 = Both Symmetric PAUSE and Asymmetric PAUSE supported by partner station	RO	00b
9	**100BASE-T4** 0 = No T4 ability 1 = T4 able **Note:** This device does not support T4 ability.	RO	0b
8	**100BASE-TX Full Duplex** 0 = No TX full-duplex ability 1 = TX with full-duplex ability	RO	0b
7	**100BASE-TX** 0 = No TX ability 1 = TX able	RO	0b
6	**10BASE-T Full Duplex** 0 = No 10 Mbps with full-duplex ability 1 = 10 Mbps with full-duplex ability	RO	0b
5	**10BASE-T** 0 = No 10 Mbps ability 1 = 10 Mbps able	RO	0b
4:0	**Selector Field** 00001 = IEEE 802.3	RO	00001b

6.5.7 AUTO NEGOTIATION EXPANSION REGISTER

Index (In Decimal): 6 Size: 16 bits

Bits	Description	Type	Default
15:5	**RESERVED**	RO	-
4	**Parallel Detection Fault** 0 = No fault detected by parallel detection logic. 1 = Fault detected by parallel detection logic.	RO/LH	0b
3	**Link Partner Next Page Able** 0 = Link partner does not have next page ability. 1 = Link partner has next page ability.	RO	0b
2	**Next Page Able** 0 = Local device does not have next page ability. 1 = Local device has next page ability. **Note:** Next page ability is not supported.	RO	0b
1	**Page Received** 0 = New page not yet received 1 = New page received	RO/LH	0b
0	**Link Partner Auto-Negotiation Able** 0 = Link partner does not have auto-negotiation ability. 1 = Link partner has auto-negotiation ability.	RO	0b

6.5.8 EDPD NLP/CROSSOVER TIMEREGISTER

Index (In Decimal): 16 Size: 16 bits

Bits	Description	Type	Default
15	**EDPD TX NLP Enable** When in Energy Detect Power-Down (EDPD) mode (EDPWRDOWN=1), this bit enables the transmission of single TX NLPs at the interval defined by the EDPD TX NLP Interval Timer Select field. 0 = TX NLP disabled 1 = TX NLP enabled when in EDPD mode	R/W NASR	0b
14:13	**EDPD TX NLP Interval Timer Select** When in Energy Detect Power-Down (EDPD) mode (EDPWRDOWN=1) and EDPD TX NLP Enable is 1, this field defines the interval used to send single TX NLPs. 00 = 1 second (default) 01 = 768 ms 10 = 512 ms 11 = 256 ms	R/W NASR	00b
12	**EDPD RX Single NLP Wake Enable** When in Energy Detect Power-Down (EDPD) mode (EDPWRDOWN=1), this bit enables waking the PHY on reception of a single RX NLP. 0 = RX NLP wake disabled 1 = TX NLP wake enabled when in EDPD mode	R/W NASR	0b
11:10	**EDPD RX NLP Max Interval Detect Select** When in Energy Detect Power-Down (EDPD) mode (EDPWRDOWN=1) and EDPD RX Single NLP Wake Enable is 0, this field defines the maximum interval for detecting two RX NLPs to wake from EDPD mode. 00 = 64 ms (default) 01 = 256 ms 10 = 512 ms 11 = 1 second	R/W NASR	00b
9:2	**RESERVED**	RO	-
1	**EDPD Extend Crossover** When in Energy Detect Power-Down (EDPD) mode (EDPWRDOWN = 1), setting this bit to 1 extends the crossover time by 2976 ms. 0 = Crossover time extension disabled 1 = Crossover time extension enabled (2976 ms)	R/W NASR	0b
0	**Extend Manual 10/100 Auto-MDIX Crossover Time** When Auto-MIDX is enabled and the PHY is in manual 10BASE-T or 100BASE-TX mode, setting this bit to 1 extends the crossover time by 1984 ms to allow linking to an auto-negotiation link partner PHY. 0 = Crossover time extension disabled 1 = Crossover time extension enabled (1984 ms)	R/W NASR	0b

6.5.9 MODE CONTROL/STATUS REGISTER

Index (In Decimal): 17 Size: 16 bits

Bits	Description	Type	Default
15:14	**RESERVED**	RO	-
13	**EDPWRDOWN** Enable the Energy Detect Power-Down mode: 0 = Energy Detect Power-Down is disabled 1 = Energy Detect Power-Down is enabled **Note:** It is recommended to enable auto-negotiation before enabling EDPD mode. **Note:** When in EDPD mode, the device's NLP characteristics can be modified via the EDPD NLP/Crossover TimeRegister.	R/W	0b
12:2	**RESERVED**	RO	-
1	**ENERGYON** Indicates whether energy is detected. This bit goes to a "0" if no valid energy is detected within 256 ms. It is reset to "1" by a hardware reset and unaffected by a software reset.	RO	1b
0	**RESERVED**	R/W	0b

6.5.10 SPECIAL MODES REGISTER

Index (In Decimal): 18 Size: 16 bits

Bits	Description	Type	Default
15:8	**RESERVED**	RO	-
7:5	**MODE** PHY Mode of operation. Refer to Table 6-7 for more details.	R/W NASR	111b
4:0	**PHYADD** PHY Address. The PHY Address is used for the SMI address.	R/W NASR	00001b

TABLE 6-7: MODE CONTROL

MODE	Mode Definitions	Default Register Bit Values	
		Register 0	Register 4
		[13,12,8]	[8,7,6,5]
000b	10BASE-T half duplex. Auto-negotiation disabled.	000	N/A
001b	10BASE-T full duplex. Auto-negotiation disabled.	001	N/A
010b	100BASE-TX half duplex. Auto-negotiation disabled. CRS is active during Transmit & Receive.	100	N/A
011b	100BASE-TX full duplex. Auto-negotiation disabled. CRS is active during Receive.	101	N/A
100b	100BASE-TX half duplex is advertised. Auto-negotiation enabled. CRS is active during Transmit & Receive.	110	0100
101b	Repeater mode. Auto-negotiation enabled. 100BASE-TX half duplex is advertised. CRS is active during Receive.	110	0100
110b	RESERVED - Do not set the device in this mode.	N/A	N/A
111b	All capable. Auto-negotiation enabled.	X1X	1111

6.5.11 SYMBOL ERROR COUNTER REGISTER

Index (In Decimal): 26 Size: 16 bits

Bits	Description	Type	Default
15:0	**Symbol Error Counter (SYM_ERR_CNT)** This 100BASE-TX receiver-based error counter increments when an invalid code symbol is received, including IDLE symbols. The counter is incremented only once per packet, even when the received packet contains more than one symbol error. This field counts up to 65,536 and rolls over to 0 if incremented beyond it's maximum value. **Note:** This register is cleared on reset, but is not cleared by reading the register. It does not increment in 10BASE-T mode.	RO	0000h

6.5.12 SPECIAL CONTROL/STATUS INDICATIONS REGISTER

Index (In Decimal): 27 Size: 16 bits

Bits	Description	Type	Default
15	**Override AUTOMDIX_EN Strap** 0 = AUTOMDIX_EN configuration strap enables or disables HP Auto MDIX 1 = Override AUTOMDIX_EN configuration strap. PHY Register 27.14 and 27.13 determine MDIX function	R/W NASR	0b
14	**Auto-MDIX Enable** Only effective when 27.15 = 1, otherwise ignored. 0 = Disable Auto-MDIX. 27.13 determines normal or reversed connection. 1 = Enable Auto-MDIX. 27.13 must be set to 0.	R/W NASR	0b
13	**Auto-MDIX State** Only effective when 27.15 = 1, otherwise ignored. When 27.14 = 0 (manually set MDIX state): 0 = No crossover (TPO = output, TPI = input) 1 = Crossover (TPO = input, TPI = output) When 27.14 = 1 (automatic MDIX) this bit must be set to 0. Do not use the combination 27.15=1, 27.14=1, 27.13=1.	R/W NASR	0b
12:5	**RESERVED**	RO	-
4	**XPOL** Polarity state of the 10BASE-T: 0 = Normal polarity 1 = Reversed polarity	RO	0b
3:0	**RESERVED**	RO	-

6.5.13 INTERRUPT SOURCE FLAG REGISTER

Index (In Decimal): 29 Size: 16 bits

Bits	Description	Type	Default
15:8	**RESERVED**	RO	-
7	**INT7** 0 = Not source of interrupt 1 = ENERGYON generated	RO/LH	0b
6	**INT6** 0 = Not source of interrupt 1 = Auto-Negotiation complete	RO/LH	0b
5	**INT5** 0 = Not source of interrupt 1 = Remote Fault Detected	RO/LH	0b
4	**INT4** 0 = Not source of interrupt 1 = Link Down (link status negated)	RO/LH	0b
3	**INT3** 0 = Not source of interrupt 1 = Auto-Negotiation LP Acknowledge	RO/LH	0b
2	**INT2** 0 = Not source of interrupt 1 = Parallel Detection Fault	RO/LH	0b
1	**INT1** 0 = Not source of interrupt 1 = Auto-Negotiation Page Received	RO/LH	0b
0	**RESERVED**	RO	0b

6.5.14 INTERRUPT MASK REGISTER

Index (In Decimal): 30 Size: 16 bits

Bits	Description	Type	Default
15:8	**RESERVED**	RO	-
7:1	**Mask Bits** These bits mask the corresponding interrupts in the Interrupt Source Flag Register. 0 = Interrupt source is masked. 1 = Interrupt source is enabled.	R/W	0000000b
0	**RESERVED**	RO	-

6.5.15 PHY SPECIAL CONTROL/STATUS REGISTER

Index (In Decimal): 31 Size: 16 bits

Bits	Description	Type	Default
15:13	**RESERVED**	RO	-
12	**Autodone** Auto-negotiation done indication: 0 = Auto-negotiation is not done or disabled (or not active). 1 = Auto-negotiation is done.	RO	0
11:5	**RESERVED - Write as 0000010b, ignore on read.**	R/W	0000010b
4:2	**Speed Indication** HCDSPEED value: 001 = 10 Mbps half-duplex 101 = 10 Mbps full-duplex 010 = 100BASE-TX half-duplex 110 = 100BASE-TX full-duplex	RO	XXXb
1:0	**RESERVED**	RO	-

7.0 OPERATIONAL CHARACTERISTICS

7.1 Absolute Maximum Ratings*

+3.3 V Supply Voltage (VDD33IO, VDD33A) Note 7-1 ..... 0 V to +3.6 V

+1.2 V Supply Voltage (VDD12CORE, VDD12PLL, VDD12USBPLL, VDD12A) Note 7-1 ..... 0 V to +1.5 V

Positive voltage on input signal pins, with respect to ground Note 7-2 ..... VDD33IO + 2.0 V

Negative voltage on input signal pins, with respect to ground Note 7-3 ..... -0.5 V

Positive voltage on XI, with respect to ground VDD12CORE

Storage Temperature ..... -55°C to +150°C

Lead Temperature Range ..... Refer to JEDEC Spec. J-STD-020

HBM ESD Performance ..... JEDEC Class 3A

Note 7-1 When powering this device from laboratory or system power supplies, it is important that the absolute maximum ratings not be exceeded or device failure can result. Some power supplies exhibit voltage spikes on their outputs when AC power is switched on or off. In addition, voltage transients on the AC power line may appear on the DC output. If this possibility exists, it is suggested to use a clamp circuit.

Note 7-2 This rating does not apply to the following pins: XI, XO, EXRES, USBRBIAS, HSIC_STROBE, HSIC_DATA.

Note 7-3 This rating does not apply to the following pins: EXRES, USBRBIAS, HSIC_STROBE, HSIC_DATA.

* Stresses exceeding those listed in this section could cause permanent damage to the device. This is a stress rating only. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. Functional operation of the device at any condition exceeding those indicated in Section 7.2, "Operating Conditions**", Section 7.4, "DC Specifications", or any other applicable section of this specification is not implied. Note, device signals are *NOT* 5 Volt tolerant unless specified otherwise.

7.2 Operating Conditions**

+3.3 V Supply Voltage (VDD33IO, VDD33A) ..... +3.3 V +/-10%

+1.2 V Supply Voltage (VDD12CORE, VDD12PLL, VDD12USBPLL, VDD12A) ..... +1.2 V +10%/-5%

Ambient Operating Temperature in Still Air (T_A) ..... Note 7-4

Note 7-4 0°C to +70°C for commercial version, -40°C to +85°C for industrial version.

** Proper operation of the device is guaranteed only within the ranges specified in this section.

7.3 Power Consumption

This section details the power consumption of the device as measured during various modes of operation. Power consumption values are provided for both the device-only, and for the device plus Ethernet components. Power dissipation is determined by temperature, supply voltage and external source/sink requirements.

Note:	All current consumption and power dissipation values were measured with VDD33IO and VDD33A equal to 3.3 V.

7.3.1 POWER CONSUMPTION - INTERNAL REGULATOR DISABLED

7.3.1.1 SUSPEND0 - Internal Regulator Disabled

TABLE 7-1: POWER CONSUMPTION/DISSIPATION - SUSPEND0 - INT. REG. DISABLED

Parameter	Min	Typ.	Max.	Unit
Supply current (VDD33IO, VDD33A) (Device Only)		30		mA
Supply current (VDD12CORE, VDD12USBPLL, VDD12A, VDD12PLL) (Device Only)		19		mA
Power Dissipation (Device Only)		121		mW
Power Dissipation (Device and Ethernet components)		255		mW

7.3.1.2 SUSPEND1 - Internal Regulator Disabled

TABLE 7-2: POWER CONSUMPTION/DISSIPATION - SUSPEND1 - INT. REG. DISABLED

Parameter	Min	Typ.	Max.	Unit
Supply current (VDD33IO, VDD33A) (Device Only)		5		mA
Supply current (VDD12CORE, VDD12USBPLL, VDD12A, VDD12PLL) (Device Only)		3		mA
Power Dissipation (Device Only)		19		mW
Power Dissipation (Device and Ethernet components)		19		mW

7.3.1.3 SUSPEND2 - Internal Regulator Disabled

TABLE 7-3: POWER CONSUMPTION/DISSIPATION - SUSPEND2 - INT. REG. DISABLED

Parameter	Min	Typ.	Max.	Unit
Supply current (VDD33IO, VDD33A) (Device Only)		1		mA
Supply current (VDD12CORE, VDD12USBPLL, VDD12A, VDD12PLL) (Device Only)		1		mA
Power Dissipation (Device Only)		5		mW
Power Dissipation (Device and Ethernet components)		6		mW

7.3.1.4 SUSPEND3 - Internal Regulator Disabled

TABLE 7-4: POWER CONSUMPTION/DISSIPATION - SUSPEND3 - INT. REG. DISABLED

Parameter	Min	Typ.	Max.	Unit
Supply current (VDD33IO, VDD33A) (Device Only)		30		mA
Supply current (VDD12CORE, VDD12USBPLL, VDD12A, VDD12PLL) (Device Only)		37		mA
Power Dissipation (Device Only)		145		mW
Power Dissipation (Device and Ethernet components)		279		mW

7.3.1.5 Operational - Internal Regulator Disabled

TABLE 7-5: OPERATIONAL POWER CONSUMPTION/DISSIPATION - INT. REG. DISABLED

Parameter	Min	Typ.	Max.	Unit
100BASE-TX Full Duplex (HSIC)				
Supply current (VDD33IO, VDD33A) (Device Only)		32		mA
Supply current (VDD12CORE, VDD12USBPLL, VDD12A, VDD12PLL) (Device Only)		42		mA
Power Dissipation (Device Only)		156		mW
Power Dissipation (Device and Ethernet components)		292		mW
10BASE-T Full Duplex (HSIC)				
Supply current (VDD33IO, VDD33A) (Device Only)		12		mA
Supply current (VDD12CORE, VDD12USBPLL, VDD12A, VDD12PLL) (Device Only)		32		mA
Power Dissipation (Device Only)		77		mW
Power Dissipation (Device and Ethernet components)		407		mW

7.3.2 POWER CONSUMPTION - INTERNAL REGULATOR ENABLED

7.3.2.1 SUSPEND0 - Internal Regulator Enabled

TABLE 7-6: POWER CONSUMPTION/DISSIPATION - SUSPEND0 - INT. REG. ENABLED

Parameter	Min	Typ.	Max.	Unit
Supply current (VDD33IO, VDD33A) (Device Only)		50		mA
Power Dissipation (Device Only)		164		mW
Power Dissipation (Device and Ethernet components)		296		mW

7.3.2.2 SUSPEND1 - Internal Regulator Enabled

TABLE 7-7: POWER CONSUMPTION/DISSIPATION - SUSPEND1 - INT. REG. ENABLED

Parameter	Min	Typ.	Max.	Unit
Supply current (VDD33IO, VDD33A) (Device Only)		8		mA
Power Dissipation (Device Only)		26		mW
Power Dissipation (Device and Ethernet components)		26		mW

7.3.2.3 SUSPEND2 - Internal Regulator Enabled

TABLE 7-8: POWER CONSUMPTION/DISSIPATION - SUSPEND2 - INT. REG. ENABLED

Parameter	Min	Typ.	Max.	Unit
Supply current (VDD33IO, VDD33A) (Device Only)		3		mA
Power Dissipation (Device Only)		9		mW
Power Dissipation (Device and Ethernet components)		10		mW

7.3.2.4 SUSPEND3 - Internal Regulator Enabled

TABLE 7-9: POWER CONSUMPTION/DISSIPATION - SUSPEND3 - INT. REG. ENABLED

Parameter	Min	Typ.	Max.	Unit
Supply current (VDD33IO, VDD33A) (Device Only)		69		mA
Power Dissipation (Device Only)		228		mW
Power Dissipation (Device and Ethernet components)		361		mW

7.3.2.5 Operational - Internal Regulator Enabled

TABLE 7-10: OPERATIONAL POWER CONSUMPTION/DISSIPATION - INT. REG. ENABLED

Parameter	Min	Typ.	Max.	Unit
100BASE-TX Full Duplex (HSIC)				
Supply current (VDD33IO, VDD33A) (Device Only)		71		mA
Power Dissipation (Device Only)		235		mW
Power Dissipation (Device and Ethernet components)		370		mW
10BASE-T Full Duplex (HSIC)				
Supply current (VDD33IO, VDD33A) (Device Only)		44		mA
Power Dissipation (Device Only)		146		mW
Power Dissipation (Device and Ethernet components)		478		mW

7.4 DC Specifications

TABLE 7-11: I/O BUFFER CHARACTERISTICS

Parameter	Symbol	Min.	Typ.	Max.	Unit	Note
IS Type Input Buffer						
Low Input Level	V_{ILI}	-0.3			V	
High Input Level	V_{IHI}			3.6	V	
Negative-Going Threshold	V_{ILT}	1.01	1.19	1.39	V	Schmitt trigger
Positive-Going Threshold	V_{IHT}	1.39	1.59	1.8	V	Schmitt trigger
SchmittTrigger Hysteresis (V_{IHT} - V_{ILT})	V_{HYS}	336	399	485	mV	
Input Leakage (V_{IN} = VSS or VDD33IO)	I_{IH}	-10		10	µA	Note 7-5
Input Capacitance	C_{IN}			3	pF	
O8 Type Buffers						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 8 mA
High Output Level	V_{OH}	VDD33IO - 0.4			V	I_{OH} = -8 mA
OD8 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 8 mA
O12 Type Buffers						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 12 mA
High Output Level	V_{OH}	VDD33IO - 0.4			V	I_{OH} = -12 mA
OD12 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 12 mA
HSIC Type Buffers						
Low Input Level	V_{IL}	-0.3		0.35*VDD12A	V	
High Input Level	V_{IH}	0.65*VDD12A		VDD12A + 0.3	V	
Low Output Level	V_{OL}			0.25*VDD12A	V	
High Output Level	V_{OH}	0.75*VDD12A			V	
I/O Pad Drive Strength (50DRIVER_EN = VSS)	O_D	38	40	42	Ohm	
I/O Pad Drive Strength (50DRIVER_EN = +3.3V)	O_D	47.5	50	52.5	Ohm	
Input Leakage (V_{IN} = VSS or VDD33IO)	I_{IH}	-10		10	µA	
Input Capacitance	C_{IN}			4	pF	

TABLE 7-11: I/O BUFFER CHARACTERISTICS (CONTINUED)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Note
ICLK Type Buffer (XI Input)						Note 7-6
Low Input Level	V_{ILI}			0.35	V	
High Input Level	V_{IHI}	0.8			V	

Note 7-5 This specification applies to all inputs and tri-stated bi-directional pins. Internal pull-down and pull-up resistors add +/- 50 µA per-pin (typical).

Note 7-6 XI can optionally be driven from a 25 MHz single-ended clock oscillator.

TABLE 7-12: 100BASE-TX TRANSCEIVER CHARACTERISTICS

Parameter	Symbol	Min.	Typ.	Max.	Unit	Note
Peak Differential Output Voltage High	V_{PPH}	950	-	1050	mVpk	Note 7-7
Peak Differential Output Voltage Low	V_{PPL}	-950	-	-1050	mVpk	Note 7-7
Signal Amplitude Symmetry	V_{SS}	98	-	102	%	Note 7-7
Signal Rise and Fall Time	T_{RF}	3.0	-	5.0	ns	Note 7-7
Rise and Fall Symmetry	T_{RFS}	-	-	0.5	ns	Note 7-7
Duty Cycle Distortion	D_{CD}	35	50	65	%	Note 7-8
Overshoot and Undershoot	V_{OS}	-	-	5	%	
Jitter				1.4	ns	Note 7-9

Note 7-7 Measured at line side of transformer, line replaced by 100 Ω (+/-1%) resistor.

Note 7-8 Offset from 16 ns pulse width at 50% of pulse peak.

Note 7-9 Measured differentially.

TABLE 7-13: 10BASE-T TRANSCEIVER CHARACTERISTICS

Parameter	Symbol	Min.	Typ.	Max.	Unit	Note
Transmitter Peak Differential Output Voltage	V_{OUT}	2.2	2.5	2.8	V	Note 7-10
Receiver Differential Squelch Threshold	V_{DS}	300	420	585	mV	

Note 7-10 Min/max. voltages guaranteed as measured with 100 Ω resistive load.

7.5 AC Specifications

This section details the various AC timing specifications of the device.

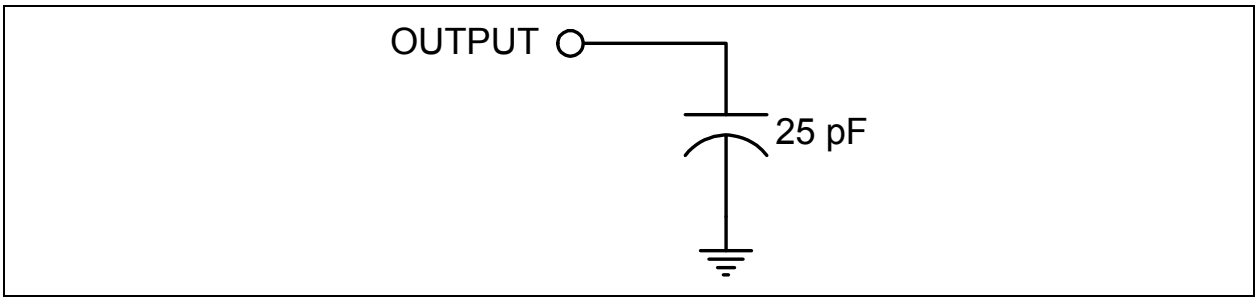
Note:	The HSIC_DATA and HSIC_STROBE pin timing adheres to the HSIC 1.0 specification. Refer to the High-Speed Interchip USB Electrical Specification Revision 1.0 (09-23-07) and USB HSIC ECN for detailed USB timing information.

Note:	The Ethernet TX/RX pin timing adheres to the IEEE 802.3 specification. Refer to the IEEE 802.3 specification for detailed Ethernet timing information.

7.5.1 EQUIVALENT TEST LOAD

Output timing specifications assume the 25 pF equivalent test load illustrated in Figure 7-1 below, unless otherwise specified.

FIGURE 7-1: OUTPUT EQUIVALENT TEST LOAD



7.5.2 POWER SEQUENCE TIMING

Power supplies must adhere to the following rules:

- All power supplies of the same voltage must be powered up/down together.
- There is no power-up sequencing requirement, however all power supplies must reach operational levels within the time periods specified in Table 7-14.
- There is no power-down sequencing or timing requirement, however the device must not be powered for an extended period of time without all supplies at operational levels.
- Do not drive input signals without power supplied to the device.
- It is acceptable for the 3.3 V supplies to remain powered up while the 1.2 V supplies are at zero volts for a period not to exceed 750 ms. In this case, nRESET must be asserted while 1.2 V is off, and must remain asserted for a minimum of 50 ms after the 1.2 V supplies reach operational level. Configuration straps must meet the requirements specified in Section 7.5.4, "Reset and Configuration Strap Timing," on page 203.

Note:	When operating with an external 1.2 V power source, the 1.2 V input must not exceed the 3.3 V source by more than 0.4 V.

Note:	Violation of these specifications may damage the device.

Note:	A Power-On Reset (POR) occurs whenever power is initially applied to the device, or if power is removed and reapplied to the device. A timer within the device will assert the internal reset for approximately 22 ms.

FIGURE 7-2: POWER SEQUENCE TIMING

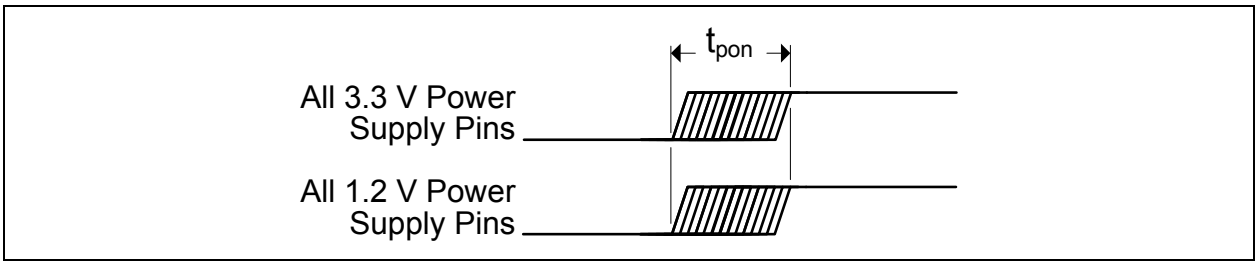


TABLE 7-14: POWER SEQUENCE TIMING VALUES

Symbol	Description	Min.	Typ.	Max.	Unit
t_{pon}	Power supply turn on time	0		750	ms

7.5.3 POWER-ON CONFIGURATION STRAP VALID TIMING

Figure 7-3 illustrates the configuration strap valid timing requirement in relation to power-on. In order for valid configuration strap values to be read at power-on, the following timing requirements must be met.

FIGURE 7-3: POWER-ON CONFIGURATION STRAP VALID TIMING

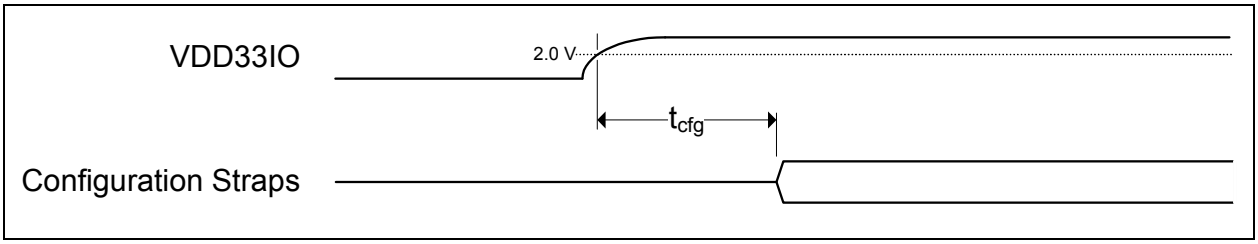


TABLE 7-15: POWER-ON CONFIGURATION STRAP VALID TIMING

Symbol	Description	Min.	Typ.	Max.	Unit
t_{cfg}	Configuration strap valid time			15	ms

7.5.4 RESET AND CONFIGURATION STRAP TIMING

Figure 7-4 illustrates the nRESET pin timing requirements and its relation to the configuration strap pins and output drive. Assertion of nRESET is not a requirement. However, if used, it must be asserted for the minimum period specified.

FIGURE 7-4: nRESET RESET PIN TIMING

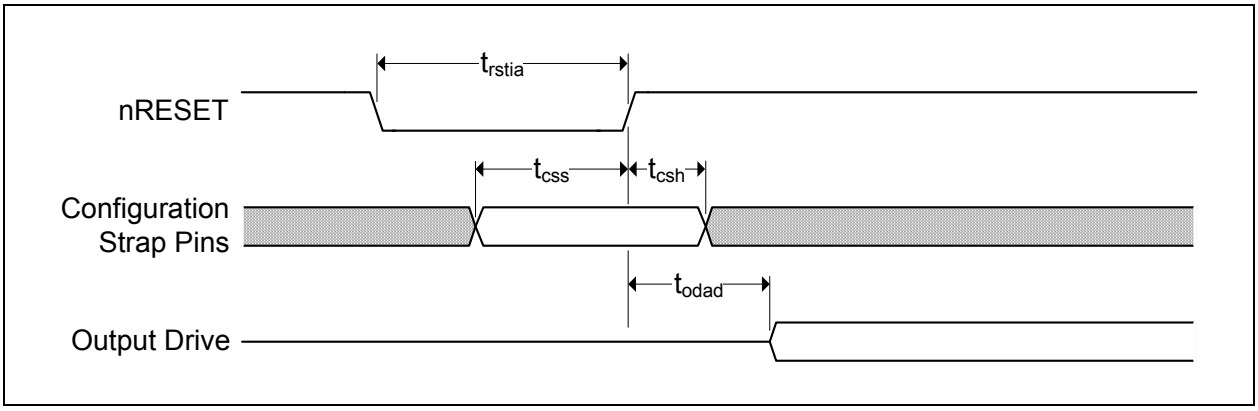


TABLE 7-16: nRESET RESET PIN TIMING VALUES

Symbol	Description	Min.	Typ.	Max.	Unit
t_{rstia}	nRESET input assertion time	1			µs
t_{css}	Configuration strap pins setup to nRESET deassertion	200			ns
t_{csh}	Configuration strap pins hold after nRESET deassertion	10			ns
t_{odad}	Output drive after nRESET deassertion	30			ns

7.5.5 EEPROM TIMING

The following specifies the EEPROM timing requirements for the device:

FIGURE 7-5: EEPROM TIMING

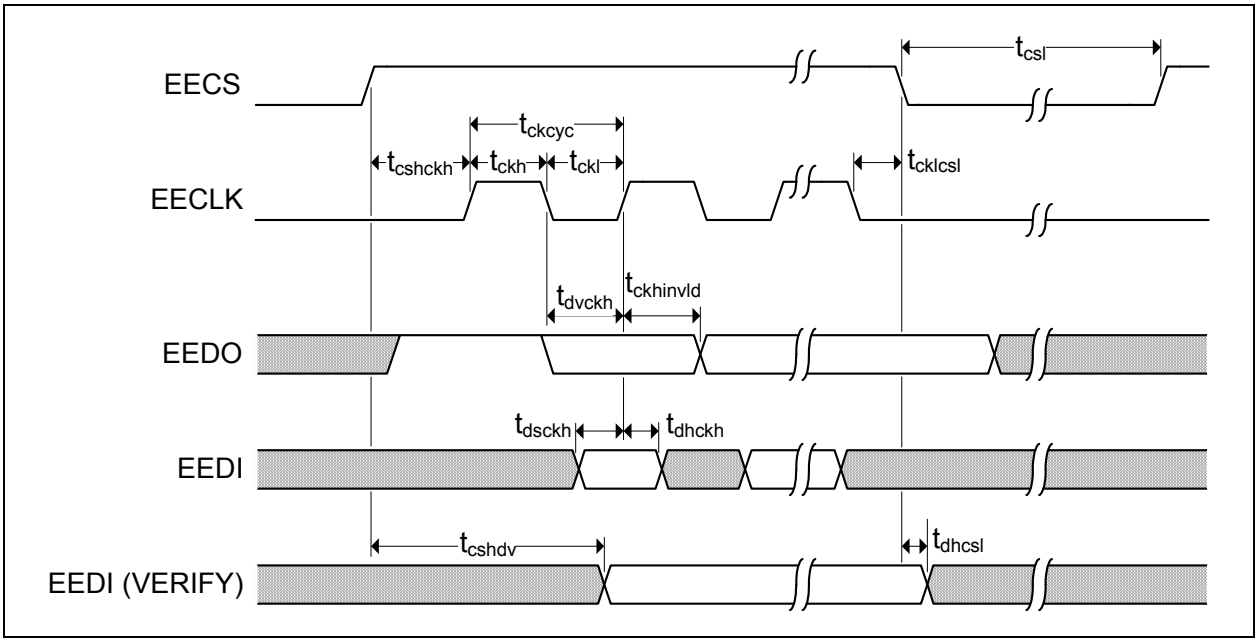


TABLE 7-17: EEPROM TIMING VALUES

Symbol	Description	Min	TYP	Max	Unit
t_{ckcyc}	EECLK cycle time	1110		1130	ns
t_{ckh}	EECLK high time	550		570	ns
t_{ckl}	EECLK low time	550		570	ns
t_{cshckh}	EECS high before rising edge of EECLK	1070			ns
t_{cklcsl}	EECLK falling edge to EECS low	30			ns
t_{dvckh}	EEDO valid before rising edge of EECLK	550			ns
$t_{ckhinvld}$	EEDO invalid after rising edge EECLK	550			ns
t_{dsckh}	EEDI setup to rising edge of EECLK	90			ns
t_{dhckh}	EEDI hold after rising edge of EECLK	0			ns
t_{cshdv}	EEDIO valid after EECS high (VERIFY)			600	ns
t_{dhcsl}	EEDIO hold after EECS low (VERIFY)	0			ns
t_{csl}	EECS low	1070			ns

7.5.6 MII INTERFACE TIMING

This section specifies the MII interface transmit and receive timing.

Note:	The MII timing adheres to the IEEE 802.3 specification. Refer to the IEEE 802.3 specification for additional MII timing information.

FIGURE 7-6: MII TRANSMIT TIMING

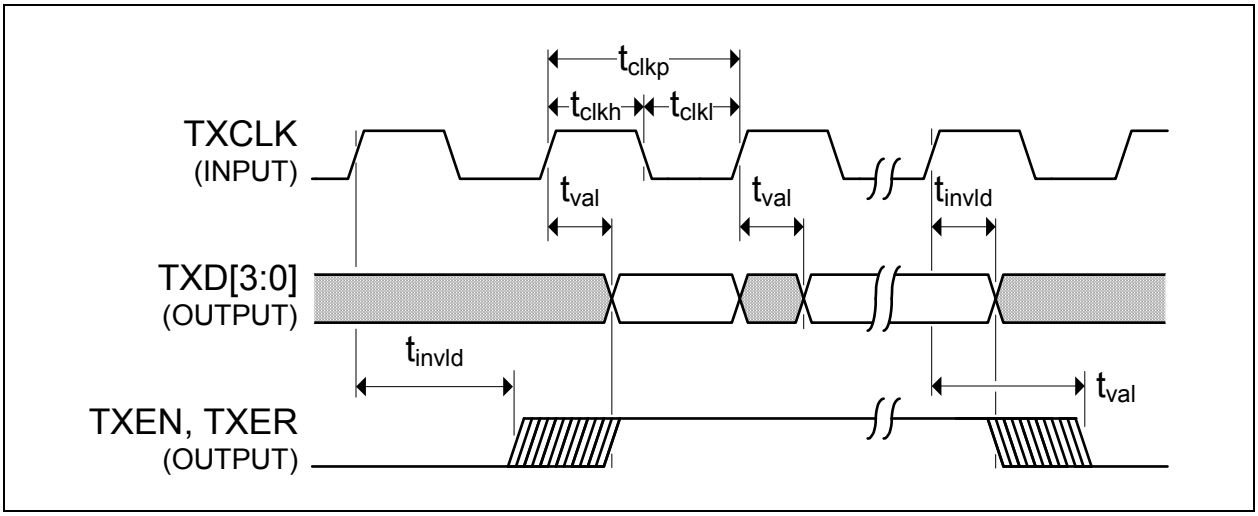


TABLE 7-18: MII TRANSMIT TIMING VALUES

Symbol	Description	Min	Max	Units	Notes
t_{clkp}	TXCLK period	40		ns	
t_{clkh}	TXCLK high time	t_{clkp}*0.4	t_{clkp}*0.6	ns	
t_{clkl}	TXCLK low time	t_{clkp}*0.4	t_{clkp}*0.6	ns	
t_{val}	TXD[3:0], TXEN, TXER output valid from rising edge of TXCLK		22.0	ns	Note 7-11
t_{invld}	TXD[3:0], TXEN, TXER output invalid from rising edge of TXCLK	0		ns	Note 7-11

Note 7-11 Timing was designed for a system load between 10 pf and 25 pf.

FIGURE 7-7: MII RECEIVE TIMING

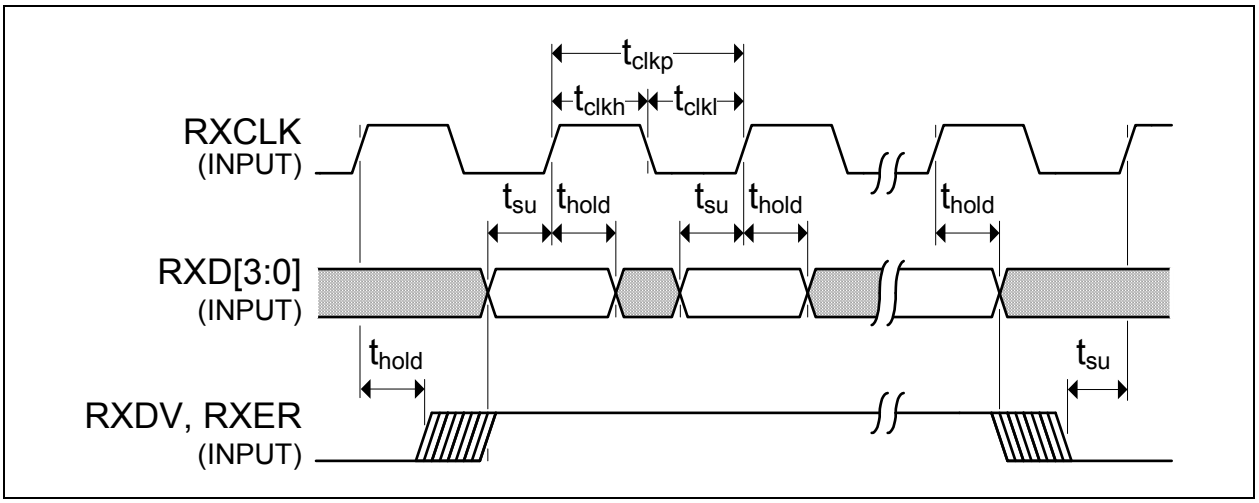


TABLE 7-19: MII RECEIVE TIMING VALUES

Symbol	Description	Min	Max	Units	Notes
t_{clkp}	RXCLK period	40		ns	
t_{clkh}	RXCLK high time	t_{clkp}*0.4	t_{clkp}*0.6	ns	
t_{clkl}	RXCLK low time	t_{clkp}*0.4	t_{clkp}*0.6	ns	
t_{su}	RXD[3:0], RXDV setup time to rising edge of RXCLK	8.0		ns	Note 7-12
t_{hold}	RXD[3:0], RXDV hold time after rising edge of RXCLK	9.0		ns	Note 7-12

Note 7-12 Timing was designed for a system load between 10 pf and 25 pf.

7.5.7 TURBO MII INTERFACE TIMING

This section specifies the Turbo MII interface transmit and receive timing.

FIGURE 7-8: TURBO MII TRANSMIT TIMING

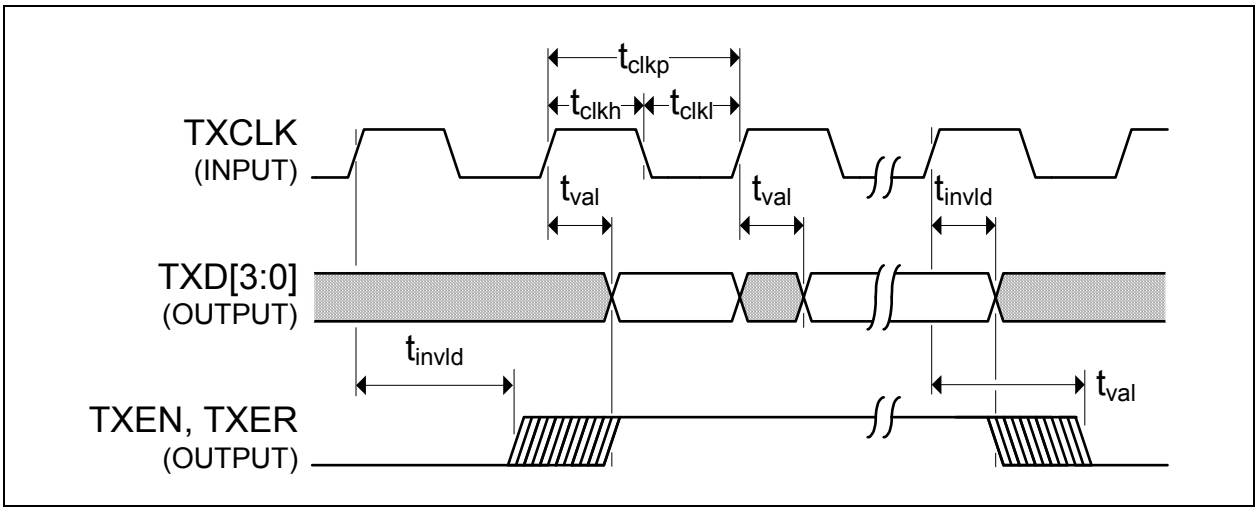


TABLE 7-20: TURBO MII TRANSMIT TIMING VALUES

Symbol	Description	Min	Max	Units	Notes
t_{clkp}	TXCLK period	20		ns	
t_{clkh}	TXCLK high time	t_{clkp}*0.4	t_{clkp}*0.6	ns	
t_{clkl}	TXCLK low time	t_{clkp}*0.4	t_{clkp}*0.6	ns	
t_{val}	TXD[3:0], TXEN, TXER output valid from rising edge of TXCLK		12.5	ns	Note 7-13
t_{invld}	TXD[3:0], TXEN, TXER output invalid from rising edge of TXCLK	1.5		ns	Note 7-13

Note 7-13 Timing was designed for a system load between 10 pf and 15 pf.

FIGURE 7-9: TURBO MII RECEIVE TIMING

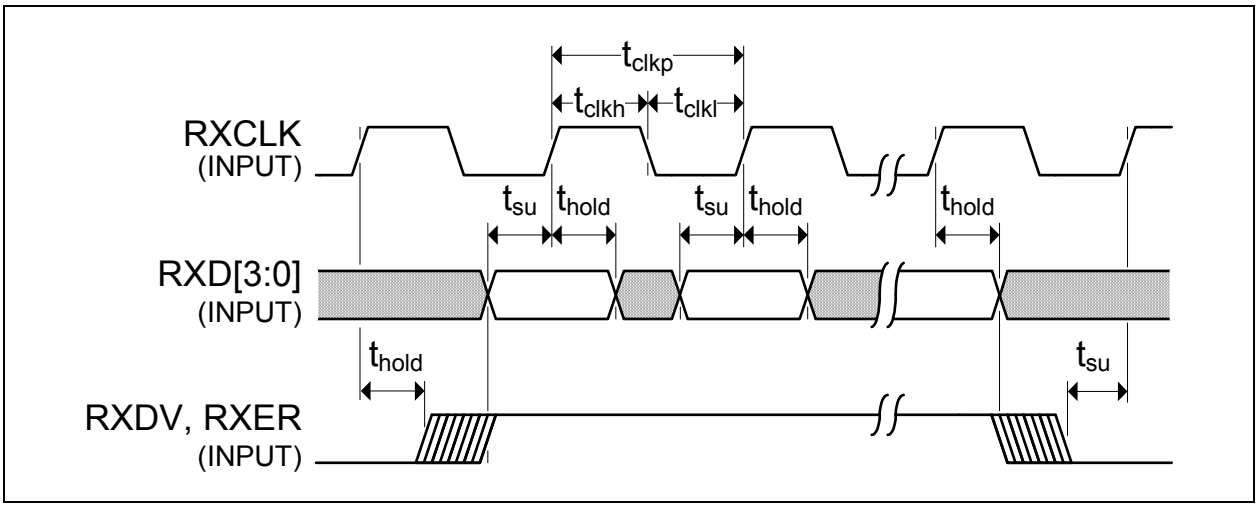


TABLE 7-21: TURBO MII RECEIVE TIMING VALUES

Symbol	Description	Min	Max	Units	Notes
t_{clkp}	RXCLK period	20		ns	
t_{clkh}	RXCLK high time	t_{clkp}*0.4	t_{clkp}*0.6	ns	
t_{clkl}	RXCLK low time	t_{clkp}*0.4	t_{clkp}*0.6	ns	
t_{su}	RXD[3:0], RXDV setup time to rising edge of RXCLK	5.5		ns	Note 7-14
t_{hold}	RXD[3:0], RXDV hold time after rising edge of RXCLK	0		ns	Note 7-14

Note 7-14 Timing was designed for a system load between 10 pf and 15 pf.

7.5.8 JTAG TIMING

This section specifies the JTAG timing of the device.

FIGURE 7-10: JTAG TIMING

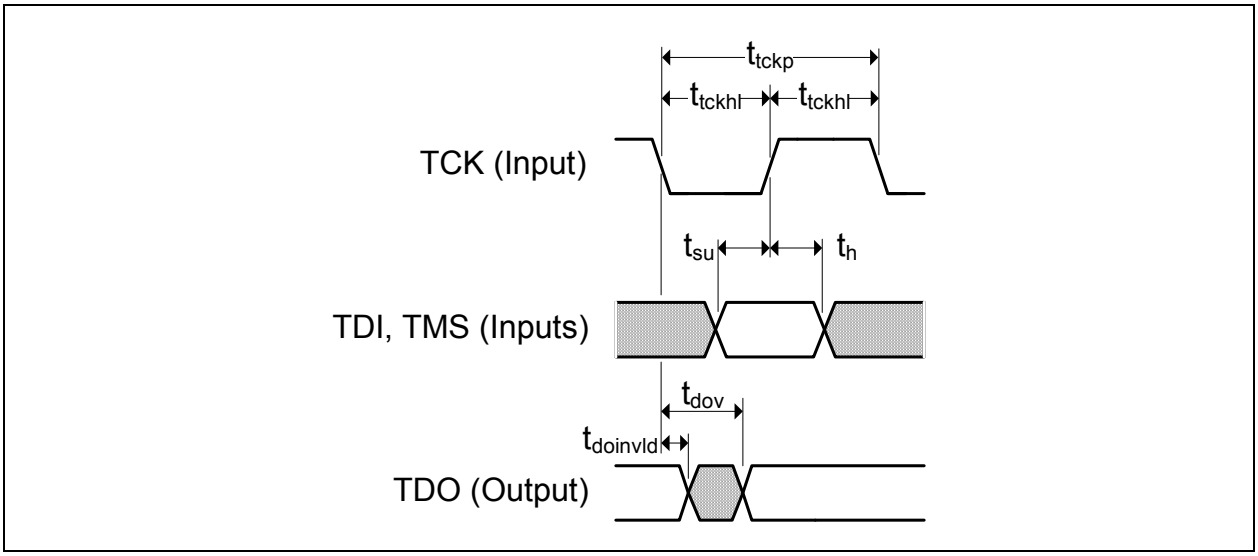


TABLE 7-22: JTAG TIMING VALUES

Symbol	Description	Min	Max	Units	Notes
t_{tckp}	TCK clock period	66.67		ns	
t_{tckhl}	TCK clock high/low time	t_{tckp}*0.4	t_{tckp}*0.6	ns	
t_{su}	TDI, TMS setup to TCK rising edge	10		ns	
t_h	TDI, TMS hold from TCK rising edge	10		ns	
t_{dov}	TDO output valid from TCK falling edge		16	ns	
$t_{doinvld}$	TDO output invalid from TCK falling edge	0		ns	

7.6 Clock Circuit

The device can accept either a 25 MHz crystal (preferred) or a 25 MHz single-ended clock oscillator (+/-50ppm) input. If the single-ended clock oscillator method is implemented, XO should be left unconnected and XI should be driven with a nominal 0-3.3 V clock signal. The input clock duty cycle is 40% minimum, 50% typical and 60% maximum.

It is recommended that a crystal utilizing matching parallel load capacitors be used for the crystal input/output signals (XI/XO). Either a 300 µW or 100 µW 25 MHz crystal may be utilized. The 300 µW 25 MHz crystal specifications are detailed in Section 7.6.1. The 100 µW 25 MHz crystal specifications are detailed in Section 7.6.2.

7.6.1 300 µW 25 MHZ CRYSTAL SPECIFICATIONS

When utilizing a 300 µW 25 MHz crystal, the following circuit design (Figure 7-11) and specifications (Table 7-23) are required to ensure proper operation.

FIGURE 7-11: 300 µW 25 MHZ CRYSTAL CIRCUIT

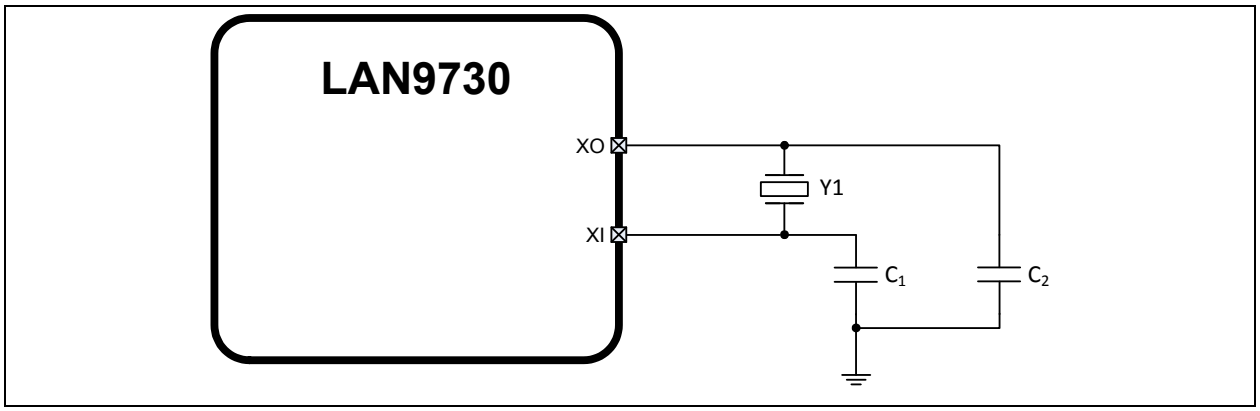


TABLE 7-23: 300 µW CRYSTAL SPECIFICATIONS

Parameter	Symbol	Min.	Nom.	Max.	Unit	Note
Crystal Cut	AT, typ					
Crystal Oscillation Mode	Fundamental Mode					
Crystal Calibration Mode	Parallel Resonant Mode					
Frequency	F_{fund}	-	25.000	-	MHz	
Frequency Tolerance @ 25°C	F_{tol}	-	-	+/-50	PPM	Note 7-15
Frequency Stability Over Temp	F_{temp}	-	-	+/-50	PPM	Note 7-15
Frequency Deviation Over Time	F_{age}	-	+/-3 to 5	-	PPM	Note 7-16
Total Allowable PPM Budget		-	-	+/-50	PPM	Note 7-17
Shunt Capacitance	C_O	-	7 typ	-	pF	
Load Capacitance	C_L	-	20 typ	-	pF	
Drive Level	P_W	300	-	-	µW	
Equivalent Series Resistance	R_1	-	-	50	Ohm	
Operating Temperature Range		Note 7-18	-	Note 7-19	°C	
XI Pin Capacitance		-	3 typ	-	pF	Note 7-20
XO Pin Capacitance		-	3 typ	-	pF	Note 7-20

7.6.2 100 µW 25 MHZ CRYSTAL SPECIFICATIONS

When utilizing a 100 µW 25 MHz crystal, the following circuit design (Figure 7-12) and specifications (Table 7-24) are required to ensure proper operation.

FIGURE 7-12: 100 µW 25 MHZ CRYSTAL CIRCUIT

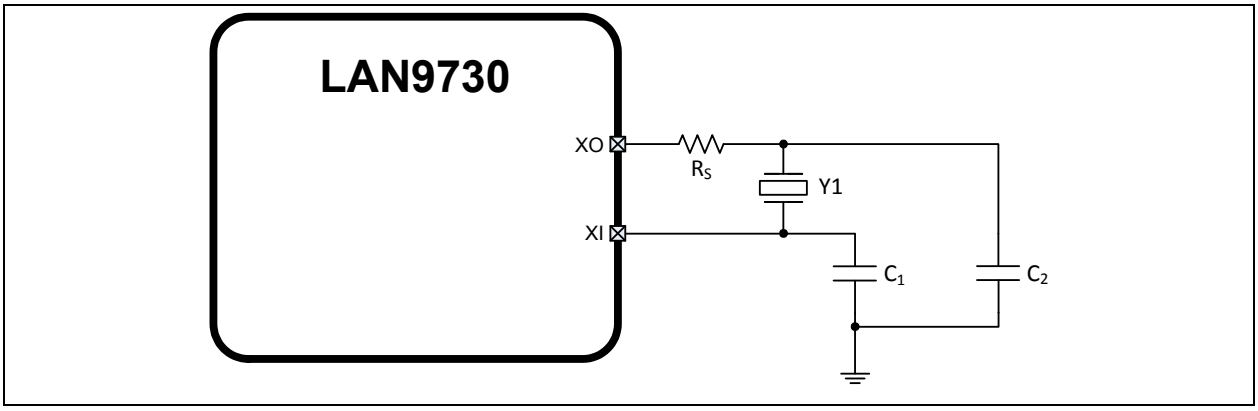


TABLE 7-24: 100 µW CRYSTAL SPECIFICATIONS

Parameter	Symbol	Min.	Nom.	Max.	Unit	Note
Crystal Cut	AT, typ					
Crystal Oscillation Mode	Fundamental Mode					
Crystal Calibration Mode	Parallel Resonant Mode					
Frequency	F_{fund}	-	25.000	-	MHz	
Frequency Tolerance @ 25°C	F_{tol}	-	-	+/-50	PPM	Note 7-15
Frequency Stability Over Temp	F_{temp}	-	-	+/-50	PPM	Note 7-15
Frequency Deviation Over Time	F_{age}	-	+/-3 to 5	-	PPM	Note 7-16
Total Allowable PPM Budget		-	-	+/-50	PPM	Note 7-17
Shunt Capacitance	C_O	-	-	5	pF	
Load Capacitance	C_L	8	-	12	pF	
Drive Level	P_W	-	100		µW	
Equivalent Series Resistance	R_1	-	-	80	Ohm	
XO Series Resistor	R_S	495	500	505	Ohm	
Operating Temperature Range		Note 7-18	-	Note 7-19	°C	
XI Pin Capacitance		-	3 typ	-	pF	Note 7-20
XO Pin Capacitance		-	3 typ	-	pF	Note 7-20

Note 7-15 The maximum allowable values for frequency tolerance and frequency stability are application dependent. Since any particular application must meet the IEEE +/-50 PPM Total PPM Budget, the combination of these two values must be approximately +/-45 PPM (allowing for aging).

Note 7-16 Frequency Deviation Over Time is also referred to as Aging.

Note 7-17 The total deviation for the Transmitter Clock Frequency is specified by IEEE 802.3u as +/-50 PPM.

Note 7-18 0°C for commercial version, -40°C for industrial version.

Note 7-19 +70°C for commercial version, +85°C for industrial version.

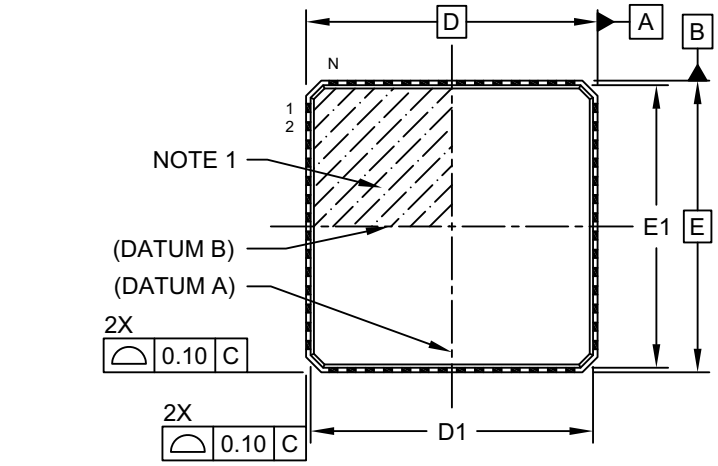
Note 7-20 This number includes the pad, the bond wire and the lead frame. PCB capacitance is not included in this value. The XO/XI pin and PCB capacitance values are required to accurately calculate the value of the two external load capacitors. These two external load capacitors determine the accuracy of the 25.000 MHz frequency.

NOTES:

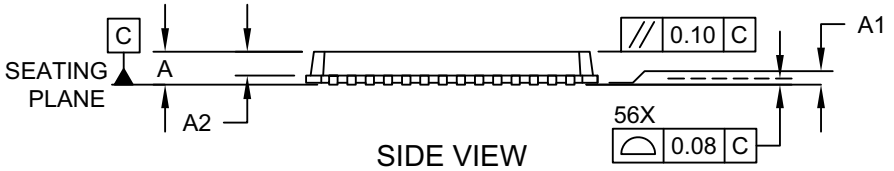
8.0 PACKAGING INFORMATION

56L Very Thin Quad Flat, No Lead Package (RT) - 8x8 mm Body [VQFN]
With 5.9x5.9 mm Exposed Pad; Punch Singulated

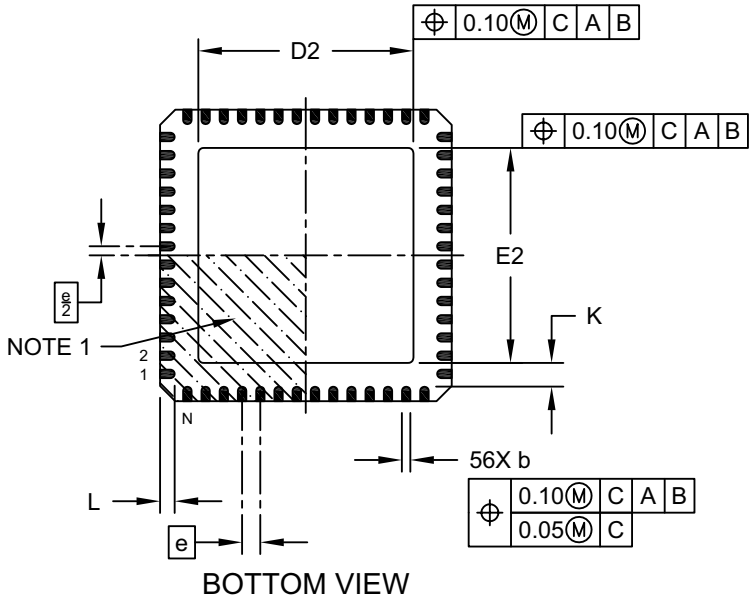
Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



TOP VIEW



SIDE VIEW

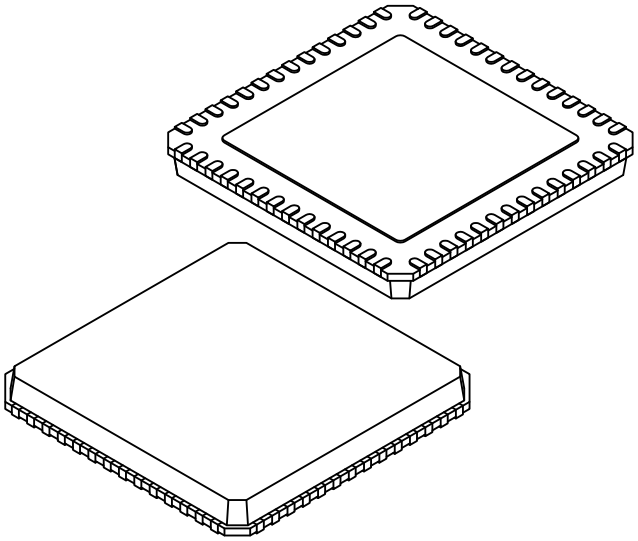


BOTTOM VIEW

Microchip Technology Drawing C04-375A Sheet 1 of 2

56L Very Thin Quad Flat, No Lead Package (RT) - 8x8 mm Body [VQFN]
With 5.9x5.9 mm Exposed Pad; Punch Singulated

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Terminals	N		56	
Pitch	e		0.50 BSC	
Overall Height	A	0.70	0.85	1.00
Standoff	A1	0.00	0.02	0.05
Mold Cap Thickness	A2	-	-	0.90
Overall Length	D		8.00 BSC	
Molded Top Length	D1	7.65	7.75	7.85
Exposed Pad Length	D2	5.80	5.90	6.00
Overall Width	E		8.00 BSC	
Molded Top Width	E1	7.65	7.75	7.85
Exposed Pad Width	E2	5.80	5.90	6.00
Terminal Width	b	0.18	0.23	0.30
Terminal Length	L	0.30	0.40	0.50
Terminal-to-Exposed-Pad	K	0.20	-	-

Notes:

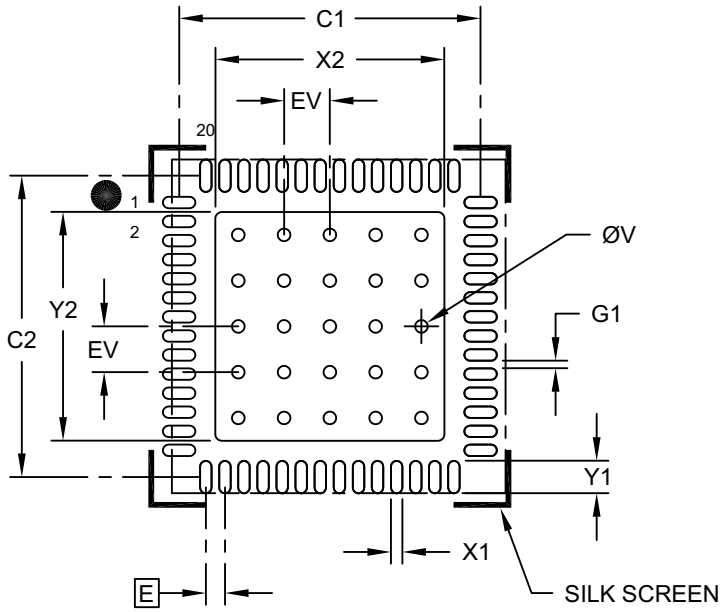
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is punch singulated
3. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-375A Sheet 2 of 2

56L Very Thin Quad Flat, No Lead Package (RT) - 8x8 mm Body [VQFN]
With 5.9x5.9 mm Exposed Pad; Punch Singulated

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.50 BSC	
Optional Center Pad Width	X2			5.90
Optional Center Pad Length	Y2			5.90
Contact Pad Spacing	C1		7.90	
Contact Pad Spacing	C2		7.90	
Contact Pad Width (X56)	X1			0.28
Contact Pad Length (X56)	Y1			0.69
Contact Pad to Center Pad (X52)	G1	0.20		
Thermal Via Diameter	V		0.33	
Thermal Via Pitch	EV		1.20	

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-2375A

NOTES:

APPENDIX A: REVISION HISTORY

Revision Level and Date	Section/Figure/ Entry	Correction
Rev. A (07-24-15)	All	• Microchip branding. • Changed ‘QFN’ to ‘VQFN’ Rev. A replaces previous numbered datasheet releases.
	Section 7.2	Updated 1.2 V supply voltage operating conditions to 1.2 V +10%/-5%
	Section 7.4, "DC Specifications", Table 7-11	Changed: • V_{ILI} min from ‘-0.3’ to ‘ ‘ (ICLK buffer) • V_{IHI} min from ‘0.33 - 0.35’ to ‘0.8’ (ICLK buffer) • V_{IHI} max from ‘3.3’ to ‘ ‘ (ICLK buffer) • I_{IH} min from ‘-4.75’ to ‘-10’ (HSIC Buffer) • I_{IH} max from ‘4.75’ to ‘10’ (HSIC Buffer)
	Section 7.6, "Clock Circuit"	Added new 100 µW crystal specifications and circuit diagram. The section is now split into two subsections, one for 300 µW crystals and the other for 100 µW crystals.
	Chapter 8, "Packaging Information"	Updated package information
	Product Identification System	Added
Rev. 1.0 (06-18-12)	All	Initial release

NOTES:

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://microchip.com/support

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO.	[X]	XXXX	[XX](1)
Device	Temperature Range	Package	Tape and Reel Option

Device:	LAN9730, LAN9730i	
Temperature Range:	Blank i	= 0°C to +70°C (commercial) = -40°C to +85°C (automotive, industrial)
Package:	ABZJ	= VQFN (56-pin)
Tape and Reel Option:	Blank TR	= Standard packaging (tray) = Tape and Reel(1)

Examples:

a) LAN9730-ABZJ
0°C to +70°C,
56-pin VQFN,
Tray

b) LAN9730i-ABZJ-TR
-40°C to +85°C,
56-pin VQFN,
Tape & Reel

Note 1: Tape and Reel identifier only appears in the catalog part number description. This identifier is used for ordering purposes and is not printed on the device package. Check with your Microchip Sales Office for package availability with the Tape and Reel option. Reel size is 3,000.

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

ISBN: 978-1-63277-410-1

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110

Canada - Toronto
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon

Hong Kong
Tel: 852-2943-5100
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Dongguan
Tel: 86-769-8702-9880

China - Hangzhou
Tel: 86-571-8792-8115
Fax: 86-571-8792-8116

China - Hong Kong SAR
Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

ASIA/PACIFIC

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-3019-1500

Japan - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo
Tel: 81-3-6880- 3770
Fax: 81-3-6880-3771

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7828

Taiwan - Taipei
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Dusseldorf
Tel: 49-2129-3766400

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Venice
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Poland - Warsaw
Tel: 48-22-3325737

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820

07/14/15